NASA CONTRACTOR
REPORT

NASA CR-170959

(NASA-CR-170959) THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED BOUNDARY CONDITIONS Final Report, 11 May 1982 - 12 May 1983 (Science Applications, Inc.) 141 p HC A07/MF A01 CSCL 20B G3/76

N84-17017

Unclas
18178

THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED BOUNDARY CONDITIONS

By L. M. Foster and J. McIntosh
Science Applications, Inc.
2109 W. Clinton Avenue, Suite 800
Huntsville, Alabama 35805

Final Report

December 1983

Prepared for

NASA-MARSHALL SPACE FLIGHT CENTER
Marshall Space Flight Center, Alabama 35812

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
NASA CR-170959		

4. TITLE AND SUBTITLE

The Control of Float Zone Interfaces By The Use of Selected Boundary Conditions

5. REPORT DATE
December 1983

6. PERFORMING ORGANIZATION CODE
JA64

7. AUTHOR(S)
L. M. Foster/J. McIntosh

8. PERFORMING ORGANIZATION REPORT #

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Science Applications, Inc.
2109 W. Clinton Avenue, Suite 800
Huntsville, AL 35805

10. WORK UNIT NO.
RTOP 179-80-70

11. CONTRACT OR GRANT NO.
NAS8-35108

13. TYPE OF REPORT & PERIOD COVERED

Contractor Report
11/5/82 - 12/5/83

12. SPONSORING AGENCY NAME AND ADDRESS

National Aeronautics and Space Administration
Washington, D. C. 20546

14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Technical Manager: I. C. Yates, Marshall Space Flight Center, AL
Final Report

16. ABSTRACT

The main goal of the float zone crystal growth project of NASA's Materials Processing in Space Program is to thoroughly understand the molten zone/freezing crystal system and all the mechanisms that govern this system.

The purpose of this effort was to study and compute the surface boundary conditions required to give flat float zone solid-melt interfaces. The results of this study provide float zone furnace designers with better methods for controlling solid-melt interface shapes and for computing thermal profiles and gradients. Documentation and a user's guide were provided for the computer software required during this study.

17. KEY WORDS

Float zone; crystal growth; solid-melt interface model; Materials Processing in Space

18. DISTRIBUTION STATEMENT

Unclassified - Unlimited

James A. Downey III

James A. Downey III, Manager
Spacelab Payload Project Office

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	136	NTIS

MSFC - Form 3292 (Rev. December 1972)

For sale by National Technical Information Service, Springfield, Virginia 22151

FOREWORD

One of the main goals of the Float Zone (FZ) growth project of NASA's Materials Processing in Space Program is to thoroughly understand the molten zone/freezing crystal system and all the mechanisms that govern this system. To accomplish this, the melt and interface properties, the heat and mass flows, and the dependencies of these on each other and on growth rate and g levels must be studied.

Since the float zone process involves two solid-melt interfaces, possible gas interfaces, heat and mass transfers, various driving forces and complex heating sources, an analysis of the entire process would be very complex. For an initial investigation, a more feasible approach is to examine each component of the process separately, particularly if mathematical models are to be manageable. The three principal components are: (1) the shapes of the melt and solid-melt interfaces, (2) the heat and mass transfers, and (3) the heating and cooling sources. This study combined facets of all three components.

The purpose of this 12-month effort was to study and compute the surface boundary conditions required to give flat FZ solid-melt interfaces. The successful completion of this study should provide FZ furnace designers with better methods for controlling solid-melt interface shapes and for computing thermal profiles and gradients.

This study was undertaken in two phases. The first phase was to investigate the solid zones surface boundary conditions required for flat solid-melt interfaces when given the melt zone surface boundary conditons. The second phase complemented the first and was to investigate the melt zone surface boundary conditions required for flat solid-melt interfaces if given the solid zones surface boundary conditions. Dual integral transform methods were used in both phases; in addition, the use of various numerical methods for differential equations and linear systems of equations were required.

Using NASA supplied data, the surface boundary conditions required for flat solid-melt interfaces were studied. In addition, complete documentation and a simple user's guide are provided for all the computer software required during this study.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1.0 INTRODUCTION

1.1 OVERVIEW AND STUDY DEFINITION

Silicon (Si) is used in a wide variety of electronic devices including high power rectifiers, space solar cells, infrared detector arrays and high density integrated circuits. The three principal industrial methods for growing silicon crystal ingots or boules are the float zone (FZ), Czochralski (Cz), and cold crucible methods. Because molten silicon acts as a universal solvent, Cz grown Si is plagued with crucible contamination which is intolerable for high performance optical and infrared devices. However, because the FZ process is containerless, crucible contaminants are avoided. Other advantages of FZ growth include uniformity of axial resistivity (on a macroscale), visibility of the growth region, low consumable material costs, and high growth rates. Although the cold crucible method combines many of the best features of the FZ and Cz techniques, the molten Si must be superheated and volatile dopants such as In, Ga, and Tl are unfortunately evaporated.

Because most industrial advances in the FZ growth technologies have come about empirically, detailed analysis of the growth process has not kept pace with presently used FZ methods. Theoretical modeling of the melt dynamics has led to some understanding of the growth process, but it is very incomplete. The characteristics of the FZ melt must be more accurately modeled if an understanding of the heat balance and flow, isotherm shapes, density (including inversion) and surface tension variations is to lead to better methods of controlling the growth conditions. Moreover, such studies should contribute to the design and execution of FZ experiments in low-gravity (g) environments. In addition, knowledge gained by studying silicon FZ methods should be applicable to other FZ processed materials.

As noted by E. Kern [10], the main goal of the FZ growth project of NASA's Material Processing in Space program is to thoroughly understand the molten zone/freezing crystal system and all the mechanisms that govern this system. In addition, more optimal crystal growth conditions at g=1 and possible improvements made by processing in near zero-g environments need to be investigated. To accomplish this, the melt and interface properties, the heat and mass flows, and the dependencies of these on each other and on the growth rate and g levels must be studied.

To transform a polycrystalline material into a single crystal, it is not always necessary to melt the entire sample or charge before growing the desired monocrystal. In some cases, it is possible to melt a small portion of the original charge, translate this molten zone through the charge, and hopefully leave a monocrystal behind the translating molten zone. The actual heating sources for this type zone melting process are varied and include induction, resistance, electron beam, and laser beam. The molten zone itself can be moved through the charge by either moving the heating source over the charge or by moving the charge through the heating source. The actual charge may, but need not be, contained in some type of crucible or ampoule. If no container is involved, the technique is called a float zone method and is used for reactive or high melting point materials. For most float zone applications, the molten zone is held intact by surface tension with the occasional aid of a magnetic field [6]. A simple illustration of the float zone technique is given in Figure 1-1.

In order to reduce nonuniformities in such things as resistivities and defect distributions, for example, the entire solid-melt system must be characterized. Since the float zone process involves two solid-melt interfaces, possible gas interfaces, heat and mass transfers, various driving forces and complex heating sources, an analysis of the entire float zone process would be very complex. A more feasible approach (at least for an initial investigation) is to examine each component of the system separately, particularly if the mathematical models are to be manageable. Three principal system components are: (1) the shapes of the melt and solid-melt interfaces, (2) the heat and mass transfers, and (3) the heating and cooling sources. This study combines facets of all three components.

While many investigators, e.g., R. Brown [2], R. Naumann [14], and W. Wilcox [20], are making significant progress studying the solid-melt interface shapes and the thermal gradients at the solid-melt interfaces for float zone and analogous systems subject to specified surface boundary conditions, the principal thrust of this effort was to study and compute the surface boundary conditions required to give flat FZ solid-melt interfaces.

The completion of this study hopefully results in a better understanding of the FZ diffusion and growth mechanisms and should provide FZ furnace designers with better methods for controlling solid-melt interface shapes and for computing thermal profiles and gradients. Moreover, the methods developed in this study should aid in the design of FZ heaters that achieve the required melt fluxes with minimal energy expenditures and, hence, perhaps reduce the system power requirements (a natural concern for any long-term, low-g FZ experiment). In particular, if radio frequency heating is used, the methodology developed in this study should be useful for computing the performance requirements and position of auxiliary heating and insulation required for the proper thermal profiles. In addition, the methodology developed in this effort might provide, for future studies, a starting point for the more complex and realistic case of a slightly concave solid-melt interface.

This study was performed in two phases. The first phase analyzed the solid zones' surface boundary conditions required for flat solid-melt interfaces when given (a priori) the melt zone surface boundary conditions. The second phase complemented the first and analyzed the melt zone surface boundary conditions required for flat solid-melt interfaces when given (a priori) the surface boundary conditions for the solid zones. Dual integral transform methods were used in both phases; in addition, both phases required the use of various numerical methods for boundary value problems. lthough such a study has apparently never before been undertaken, analogous studies for the Bridgman-Stockbarger method have been completed by L. Foster [7], [8].

Mathematical descriptions of the problems posed above are stated in Section 1-2. In Chapter 2, various mathematical tools are developed followed by some rather interesting examples. The methodologies used to compute the surface

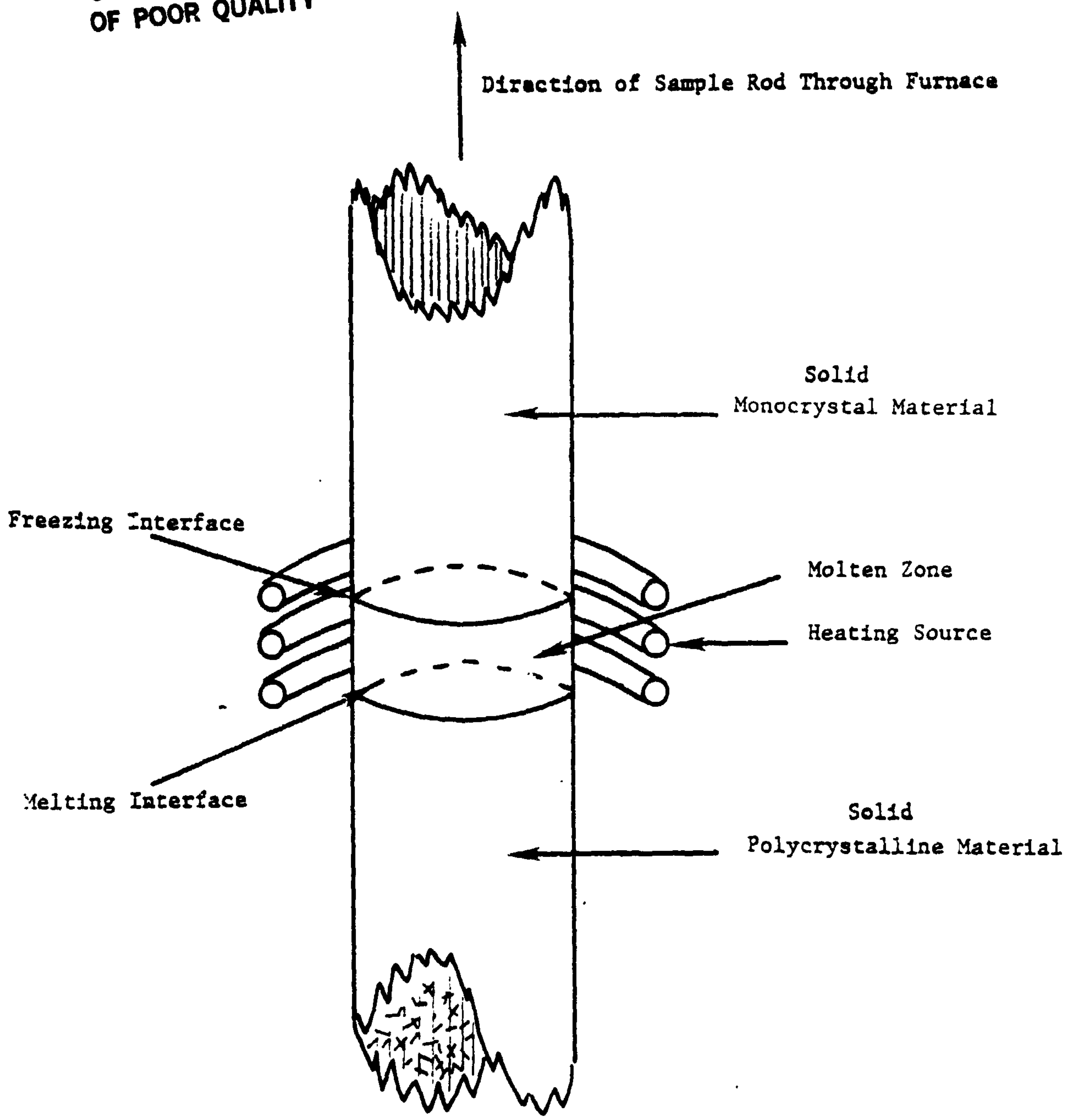


Figure 1-1 Illustration of the Float Zone Process

boundary conditions for the solid regions and melt zone surface boundary conditions required for flat interface shapes are developed in Chapters 3 and 4. The results of various test cases using NASA supplied data are presented in Chapter 5 and recommendations for future efforts are given in Chapter 6. A simple user's guide to various computer codes (listed in Appendix C) implementing the methods described in Chapters 2, 3, and 4 is presented in of Appendix A. Appendix B contains the proof of a claim made in Section 2.3.

1.2 MATHEMATICAL STATEMENT OF THE CONTROL PROBLEMS

Concise mathematical statements of the problems described in the previous section are given next using the numbered equations in the FZ model shown in Figure 1-2. First suppose that the melt zone surface temperature is some a priori known (by design or happenstance) distribution $h(x)$ (Figure 1-2, Equation (FZ8)). Since the temperature at both of the assumed flat solid-melt interfaces is the material melting point (Equations (FZ1) and (FZ3)), the temperature distribution in the melt zone is known and may be computed by the method described in Section 2.2. Hence, the axial thermal gradients in the melt zone at both of the solid-melt interfaces are known (see Section 2.2). Invoking Equations (FZ2) and (FZ4), the solid regions' axial thermal gradients at the interfaces are also known.†

For the moment, consider the lower solid region ($x \leq 0$ in Figure 1-2) and let $B(r)$ denote the known required thermal gradient* in the solid region at the interface ($x=0$), i.e.,

$$T_x(0,r) = B(r) \ , \ 0 < r < 1 \tag{1.2.1}$$

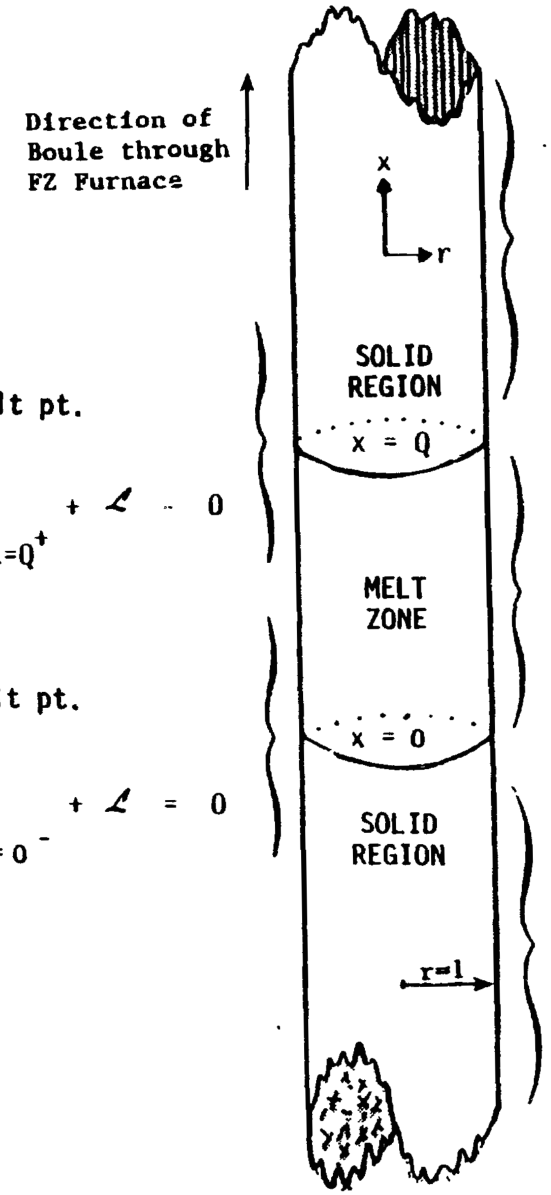
The basic idea is to compute a temperature distribution $f(x)$, $x \leq 0$, (henceforth called a surface control function), to be maintained on the surface of the lower solid region such that the resulting temperature distribution, $T(x,r)$, for the lower solid region satisfies Equation (1.2.1). This is concisely stated in Problem P1-1.

† Equations (FZ2) and (FZ4) of Figure 1-2 guarantee the conservation of energy at the solid-melt interfaces ($x=0$ and $x=Q$). k_ℓ and k_s are the solid and liquid thermal conductivities while $\mathcal{L}$ is product of the growth rate, solid density and latent heat of fusion [15].

* Standard mathematical nomenclature is used in this report. Both the operator and subscript notation are used for partial derivatives, e.g., $\frac{\partial T}{\partial x}$ and T_x both denote the partial derivative of $T(x,r)$ with respect to x. For functions of one variable, the "prime" convention for derivatives is observed, e.g., $h''(x)$ denotes the second derivative of $h(x)$. The Laplacian operator is denoted by Δ and is, in cylindrical coordinates,

$$\Delta T = T_{xx} + T_{rr} + \frac{1}{r} T_r .$$

ORIGINAL PAGE IS
OF POOR QUALITY



(FZ1) $\left. T \right)_{x=Q^-} = \left. T \right)_{x=Q^+} = \text{melt pt.}$

(FZ2) $\left. -k_\ell \frac{\partial T}{\partial x} \right)_{x=Q^-} \left. + k_s \frac{\partial T}{\partial x} \right)_{x=Q^+} + \mathcal{L} = 0$

(FZ3) $\left. T \right)_{x=0^+} = \left. T \right)_{x=0^-} = \text{melt pt.}$

(FZ4) $\left. -k_\ell \frac{\partial T}{\partial x} \right)_{x=0^+} \left. + k_s \frac{\partial T}{\partial x} \right)_{x=0^-} + \mathcal{L} = 0$

$\Delta T = P_s \frac{\partial T}{\partial x}$ (FZ5)

$T(x,1) = g(x)$ (FZ6)

$\Delta T = P_\ell \frac{\partial T}{\partial x}$ (FZ7)

$T(x,1) = h(x)$ (FZ8)

$\Delta T = P_s \frac{\partial T}{\partial x}$ (FZ9)

$T(x,1) = f(x)$ (FZ10)

Figure 1-2 FZ Model

ORIGINAL PAGE IS
OF POOR QUALITY

Problem P1-1 Compute f(x) such that the solution T(x,r) of

$$\left.\begin{array}{l} \Delta T = P_s \dfrac{\partial T}{\partial x}, \ x < 0, \ 0 < r < 1 \\ T(x,1) = f(x), \ x < 0, \\ T(0,r) = A(r), \ 0 < r < 1 \end{array}\right\} \qquad (1.2.$$

also satisfies the boundary condition (1.2.1)

The constant P_s is the solid Peclet number [20] and from a practical viewpoint, the function A(r) in Equations (1.2.1) is the material melting point. Moreover, since numerical methods will be employed, Condition (1.2.1) will only be satisfied approximately in practice.

Having stated the question for the lower solid region, the corresponding question for the upper solid region is analogous. Namely, let $B(r)^{\dagger}$ be the required thermal gradient in the upper solid region at the upper solid-melt interface (x=Q), i.e.,

$$\frac{\partial T}{\partial x}(x,r) = B(r), \ x = Q, \ 0 < r < 1 \ . \qquad (1.2.3)$$

Then find a surface temperature distribution g(x), $x \geq Q$ (henceforth also called a surface control function), to be maintained such that the resulting temperature distribution, T(x,r), for the upper solid region satisfies (1.2.3). This is concisely stated in Problem P1-2.

Problem P1-2. Determine g(x) such that the solution T(x,r) of

$$\left.\begin{array}{l} \Delta T = P_s \dfrac{\partial T}{\partial x}, \ x > Q, \ 0 < r < 1 \\ T(x,1) = g(x), \ x > Q, \\ T(Q,r) = A(r), \ 0 < r < 1 \end{array}\right\} \qquad (1.2.4)$$

also satisfies the boundary condition (1.2.3).

† To help make various computer codes listed in Appendix C easier to follow, the thermal gradients in Problem P1-1 and P1-2 are both represented by the same symbol, B(r); however, these gradients are not necessarily the same. For generality, a similar remark holds for the symbol A(r).

As before, in practice $A(r)$ is set to the melting temperature and Equation (1.2.3) will only be satisfied approximately due to the numerical solution of the problem.

Problems P1-1 and P1-2 stated above belong to the class of so called ill-posed or over-under posed problems. Unlike most classical second order boundary value problems where each portion of the boundary surface is assigned a boundary condition, Problems P1-1 and P1-2 have two boundary conditions (over-posed) assigned to each of their respective solid-melt interfaces (for example, in Problem P1-1, $T(0,r)= (r)$ and $T_x(0,r)=B(r)$) and no boundary condition (under-posed) assigned to the lateral surfaces of either of the solid regions. Indeed, part of the problem is to determine the proper missing boundary condition (for example, $T(x,1)=f(x)$ for Problem P1-1) so as to relax the overposing of boundary conditions at the solid-melt interfaces. The solutions of Problem P1-1 and P1-2 are the subject of Chapter 3.

Next suppose that the solid regions' surface temperature distributions $f(x)$ and $g(x)$ (see Figure 1-2, Equations (FZ6) and (FZ10)) are fixed (by design or happenstance). Since the temperature at both of the solid-melt interfaces is assumed to be the melting temperature for FZ applications, the temperature distributions in both of the solid regions are computable (see Section 2.3). Hence the axial thermal gradients in the solid regions at the solid-melt interfaces are computable. Thus, the axial thermal gradients in the melt zone at the solid-melt interfaces ($x=0$ and $x=Q$) are known after invoking Equations (FZ2) and (FZ4) of Figure 1-2 and are denoted by

$$\left.\begin{aligned} T_x(0,r) &= A(r),\ 0 < r < 1 \\ T_x(Q,r) &= B(r),\ 0 < r < 1 \end{aligned}\right\} \qquad (1.2.5)$$

The problem is to determine a surface temperature $h(x)$, $0 \le x \le Q$ (henceforth called the melt zone surface control function), to be maintained on the melt zone surface such that the resulting temperature distribution, $T(x,r)$, for the melt zone satisfies (1.2.5). This is concisely stated in Problem P1-3.

<u>Problem P1-3</u> Determine $h(x)$ such that the solution $T(x,r)$ of

$$\left.\begin{aligned} \Delta T &= P_\ell \frac{\partial T}{\partial x},\ 0 < x < Q,\ 0 < r < 1 \\ T(x,1) &= h(x),\ 0 < x < Q \\ T(0,r) &= C(r),\ 0 < r < 1 \\ T(Q,r) &= D(r),\ 0 < r < 1 \end{aligned}\right\} \qquad (1.2.6)$$

also satisfies the boundary conditions (1.2.5)

The constant P_ℓ is the liquid Peclet number and from a FZ point of view, C(r)and D(r) equal the material melting point. As with Problems P1-1 and P1-2, the numerical nature of the proposed solution method (the subject of Section 4.0) means that Conditions (1.2.5) will only be approximately satisfied.

ORIGINAL PAGE IS OF POOR QUALITY

2.0 TWO CLASSICAL PROBLEMS

Before turning to those moral and mental aspects of the matter which present the greatest difficulties, let the inquirer begin by mastering more elementary problems.

--Sherlock Holmes, "A Study in Scarlet"

2.1 DESCRIPTION OF THE CLASSICAL PROBLEMS AND CHAPTER OUTLINE

Before developing methods to compute the melt zone and solid regions' surface control functions which will yield the desired flat solid-melt interfaces†, two more elementary problems must be dispatched. These are:

Problem P2-1: Given a surface temperature distribution for the melt zone, compute the resulting interior temperature distribution of the melt zone.

Problem P2-2: Given a surface temperature distribution for one of the semi-infinite solid regions, compute the resulting interior temperature distribution for that region.

In addition to solving Problems P2-1 and P2-2, methods for approximating the interface gradients are presented in this chapter. The techniques developed to solve Problems P2-1 and P2-2 will have three important functions in this study. First, they will be used to generate the solid and melt zone gradients required at the interfaces. Second, and probably most important, the solution techniques for Problems P2-1 and P2-2 will introduce the essential definitions and dual integral transforms which will be used later to compute the desired surface control functions (Chapters 3 and 4). Third, these techniques will be used to study how well (or poorly) the computed melt zone (or solid region) surface control function performs.

Problems P2-1 and P2-2 are resolved in Sections 2.2 and 2.3 respectively. Some numerical test cases are discussed in Section 2.3 along with two examples with correspondingly important remarks.

2.2 SOLUTION OF PROBLEM P2-1

Suppose the melt zone of Figure 1-2 is isolated (and perhaps translated) as displayed in Figure 2-1.

† To reduce the terminology, the solid-melt interfaces will henceforth be referred to merely as the interfaces. The axial thermal gradient in a solid region (or melt zone) at an interface will be referred to as a solid region (a melt zone) interface gradient.

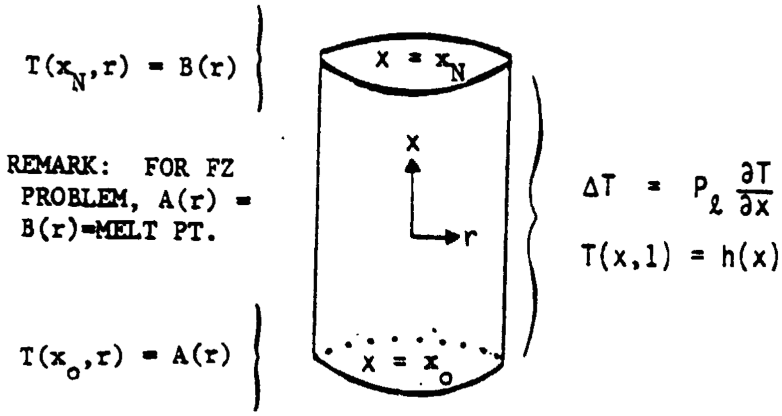


Figure 2-1 Generalized Melt Zone

Realistically, the melt zone end temperatures A(r) and B(r) are both the material melting temperature; however, for sake of illustration, we require only that A(r) and B(r) be sufficiently smooth. Problem P2-1 can then be mathematically stated as:

<u>Problem P2-3:</u> Determine T(x,r) such that

$$\Delta T = PT_x \ , \ 0 < r < 1 \text{ and } x_o < x < x_N \tag{2.2.1}$$

$$T(x_o,r) = A(r) \ , \ 0 < r < 1 \tag{2.2.2}$$

$$T(x_N,r) = B(r) \ , \ 0 < r < 1 \tag{2.2.3}$$

$$T(x,1) = h(x) \ , \ x_o < x < x_N \tag{2.2.4}$$

and

$$T_r(x,0) = 0 \ , \ x_o < x < x_N \tag{2.2.5}$$

where A(r), B(r) and h(x) are sufficiently smooth, $A(1)=h(x_o)$ and $B(1)=h(x_N)$, and P (the Peclet number with the subscript "ℓ" suppressed for convenience) is a positive constant.

Before solving Problem P2-3, some notation is in order:

<u>Notation N2-1:</u>

(i) $\mathcal{A}(r) = A(r)-A(1)$

ORIGINAL PAGE IS OF POOR QUALITY

(ii) $\mathscr{B}(r) = B(r) - B(1)$

(iii) $\psi_n(r) = J_0(\lambda_n r)$ where $\lambda_1 < \lambda_2 < \lambda_3 < \ldots$ is the increasing sequence of real roots of the Bessel function J_0.

(iv) $G(x) = Ph'(x) - h''(x)$

Solution Technique: The basic idea is to assume the solution $T(x,r)$ is the sum of the lateral surface temperature $h(x)$ plus some unknown function $\theta(x,r)$, i.e.,

$$T(x,r) = \theta(x,r) + h(x)$$

Problem P2-3 can then be recast as:

$$\Delta\theta = P\theta_x + G \ , \quad 0 < r < 1 \text{ and } x_o < x < x_N \tag{2.2.6}$$

$$\theta(x_o,r) = \mathscr{A}(r) \quad , \ 0 < r < 1 \tag{2.2.7}$$

$$\theta(x_N,r) = \mathscr{B}(r) \quad , \ 0 < r < 1 \tag{2.2.8}$$

$$\theta(x,1) = 0 \quad , \ x_o < x < x_N \tag{2.2.9}$$

and

$$\theta_r(x,0) = 0 \quad , \ x_o < x < x_N \tag{2.2.10}$$

Although Equation (2.2.6) is more complex than Equation (2.2.1), the corresponding boundary conditions are greatly simplified. First, the Dirichlet condition (2.2.4) is replaced by a simple homogenous boundary condition (2.2.9). In addition, because $\mathscr{A}(1) = \mathscr{B}(1) = 0$, the boundary conditions (2.2.7) and (2.2.8) can be further simplified by various Bessel series expansions. For the moment, assume $\theta(x,r)$ is expanded as

$$\theta(x,r) = \sum_{n=1}^{\infty} C_n(x)\psi_n(r) \tag{2.2.11}$$

Then using the following well known property of Bessel functions [18],

$$\int_0^1 \psi_n(r)\psi_m(r)\,r\,dr = \begin{cases} 0 \text{ if } n \neq m \\ \frac{1}{2} J_1^2(\lambda_n) \text{ if } n = m \end{cases} \tag{2.2.12}$$

the functions $C_n(x)$ of Equation (2.2.11) are computed to be

$$C_n(x) = \frac{2}{J_1^2(\lambda_n)} \int_0^1 \theta(x,r)\psi_n(r)\,r\,dr \tag{2.2.13}$$

If the integral portion of Equation (2.2.13) is denoted by $\overline{\theta}_n(x)$, then Equations (2.2.11) and (2.2.13) may be combined to form a dual integral transform pair:

$$\theta(x,r) = \sum_{n=1}^{\infty} \frac{2\psi_n(r)\overline{\theta}_n(x)}{J_1^{\,2}(\lambda_n)} \quad \text{and} \quad \overline{\theta}_n(x) = \int_0^1 \theta(x,r)\psi_n(r)\,r\,dr \;. \tag{2.2.14}$$

Unfortunately, the desired $\theta(x,r)$ of Equation (2.2.14) involves $\overline{\theta}_n(x)$ which in turn requires knowing $\theta(x,r)$; fortunately, this rather circular problem may be resolved by invoking Green's theorem. If both sides of the partial differential equation (2.2.6) are multiplied by $\psi_n(r)rdr$ and the resulting terms integrated from $r=0$ to $r=1$, a application of Green's theorem combined with the fact that

$$\psi_n\theta_r - \theta\frac{\partial}{\partial r}\psi_n \Bigg|_{r=0}^{r=1} = 0$$

implies

$$\overline{\theta}_n''(x) - P\overline{\theta}_n'(x) - \lambda_n^2\,\overline{\theta}_n(x) = \overline{G}_n(x) \quad , \; x_o < x < x_N \tag{2.2.15}$$

where

$$\overline{G}_n(x) = \int_0^1 G(x)\psi_n(r)\,r\,dr = G(x)\,\frac{J_1(\lambda_n)}{\lambda_n} \tag{2.2.16}$$

Since $\mathcal{A}(1) = \mathcal{B}(1) = 0$, the smooth functions $\mathcal{A}(r)$ and $\mathcal{B}(r)$ may be represented by the following Bessel expansions:

$$\mathcal{A}(r) = \sum_{n=1}^{\infty} \mathcal{A}_n J_o(\lambda_n r) \tag{2.2.17}$$

$$\mathcal{B}(r) = \sum_{n=1}^{\infty} \mathcal{B}_n J_o(\lambda_n r) \tag{2.2.18}$$

The above coefficients $\mathcal{A}_n$ and $\mathcal{B}_n$ could be computed using integral

ORIGINAL PAGE IS
OF POOR QUALITY

representations [18], for example,

$$\mathcal{A}_n = \frac{2}{J_1^2(\lambda_n)} \int_0^1 \mathcal{A}(r) J_o(\lambda_n r) r dr$$

However, to avoid the eventually required numerical integration of such representations, the coefficients $\mathcal{A}_n$ and $\mathcal{B}_n$ may be approximated using a least squares method as described at the end of this section. Combining Equations (2.2.7), (2.2.8), (2.2.12), and (2.2.14)-(2.2.18), $\bar{\theta}_n(x)$ may be uncoupled from $\theta(x,r)$ as the solution of the following two point boundary value problem:

$$\left.\begin{aligned} &\bar{\theta}_n'' - P\bar{\theta}_n' - \lambda_n^2 \bar{\theta}_n = \bar{G}_n \quad , x_o < x < x_N \\ &\bar{\theta}_n(x_o) = \mathcal{A}_n \frac{J_1^2(\lambda_n)}{\lambda_n} \\ &\bar{\theta}_n(x_N) = \mathcal{B}_n \frac{J_1^2(\lambda_n)}{\lambda_n} \end{aligned}\right\} \qquad (2.2.19)$$

Since $\lambda_n^2 > 0$, it is well known [5] that Problem (2.2.19) has a unique solution. Although the solution of (2.2.19) could be determined by a variation of parameters method [1], such a technique inevitably requires numerical integration. A more straightforward method is to discretize (2.2.19) in the following fashion. First, the interval from x_o to x_N is partitioned by the grid points:

$$t_j = x_o + j\Delta x \quad , j = 0, \ldots, M$$

where $M \cdot \Delta x = x_N - x_o$. Then solve the following finite difference analog of the boundary value problem (2.2.19):

$$\left.\begin{aligned} &\frac{\mu_{j+1} - 2\mu_j + \mu_{j-1}}{(\Delta x)^2} - P\frac{\mu_{j+1} - \mu_{j-1}}{2\Delta x} - \lambda_n^2 \mu_j = \bar{G}_n(t_j), \ j = 1, \cdots, M-1 \\ &\mu_o = \mathcal{A}_n \frac{J_1^2(\lambda_n)}{2} \\ &\mu_N = \mathcal{B}_n \frac{J_1^2(\lambda_n)}{2} \end{aligned}\right\} \qquad (2.2.20)$$

ORIGINAL PAGE IS
OF POOR QUALITY

The linear system (2.2.20) is tridiagonal and guaranteed to have a solution [9] if $P \cdot \Delta x \leq 2$. Moreover, the solution vector $\{\mu_o, \cdots, \mu_M\}$ provides a second order approximation of $\bar{\theta}_n(x)$, i.e.,

$$\left|\mu_j - \bar{\theta}_n(t_j)\right| = O\left((\Delta x)^2\right)$$

In addition, the boundary derivatives of $\bar{\theta}_n$ may be accurately approximated [3] by the following unbalanced finite differences:

$$\left.\begin{aligned} \bar{\theta}_n'(x_o) &\approx (-3\mu_4 + 16\mu_3 - 36\mu_2 + 48\mu_1 - 25\mu_o)/12\Delta x \\ \bar{\theta}_n'(x_N) &\approx (3\mu_{M-4} - 16\mu_{M-3} + 36\mu_{M-2} - 48\mu_{M-1} + 25\mu_M)/12\Delta x \end{aligned}\right\} \tag{2.2.21}$$

Since

$$T_x(x,r) = h'(x) + 2\sum_{n=1}^{\infty} \frac{\psi_n(r)}{J_1^2(\lambda_n)} \bar{\theta}_n'(x) \tag{2.2.22}$$

Equations (2.2.21) and (2.2.22) may be combined to approximate the axial gradients at $x=x_o$ and x_N (a very important requirement in Chapters 3 and 4).

To finish this section, a short description is given of how the coefficients $\mathcal{A}_n$ of Equation (2.2.17) are approximated (the same technique applies to Equation (2.2.18)). First, denote $r_i = (i-1)/M$, $i = 1, \ldots, M+1$ and select $N \ll M$ (typically $N = 20$ and $M = 100$). Define an $(M+1)$ by N array L and $(M+1)$ dimension vector $\bar{b}$ by the respective elements:

$$L_{ij} = J_o(\lambda_j r_i) \text{ and } b_i = \mathcal{A}(r_i)$$

Let $\bar{a}$ be the solution of the linear least squares problem [17, Chapter 5]:

$$L\bar{a} = \bar{b} \tag{2.2.23}$$

Then the first N coefficients, $\mathcal{A}_n$, of (2.2.17) are approximated by

$$\mathcal{A}_n \approx a_n, \; n = 1, \cdots, N$$

2.3 SOLUTION OF PROBLEM P2-2

Analogous to the solution technique of Problem P2-1 in Section 2.2, suppose the lower solid region of Figure 1-2 is isolated as shown in Figure 2-2.

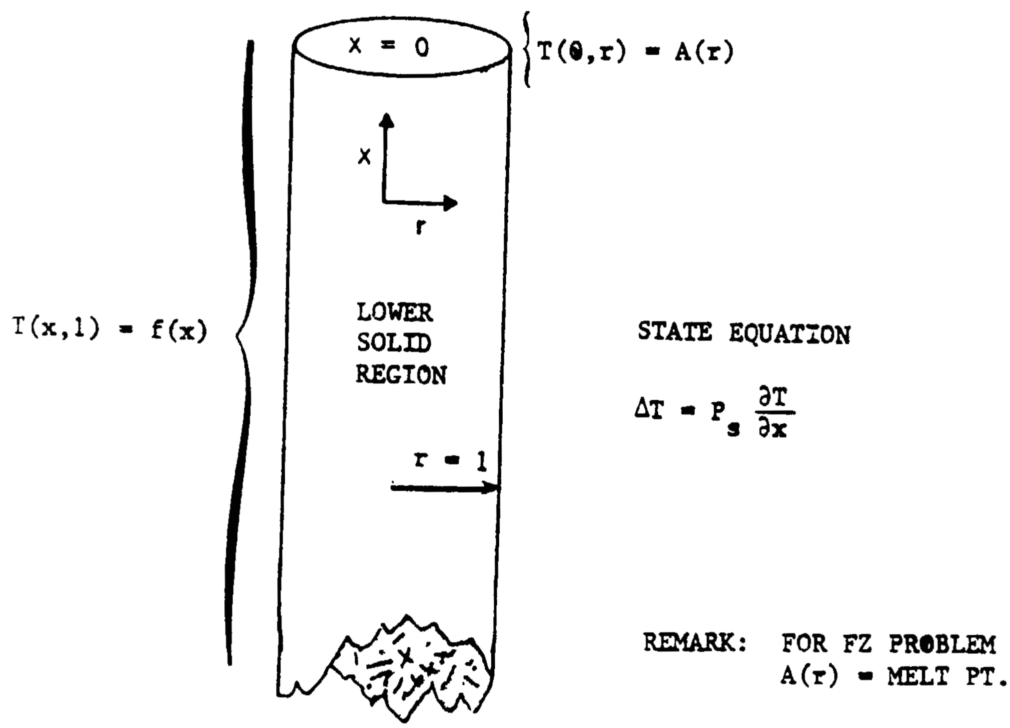


Figure 2-2 Generalized Lower Solid Region

Realistically, the upper end temperature $A(r)$ of the lower solid region is the material melting temperature; however, for the sake of illustration, it is only required that $A(r)$ be sufficiently smooth. In addition, it is assumed that the lateral surface temperature $f(x)$ is smooth, asymptotically constant as $x \to -\infty$ and is such that f' and f'' approach zero as $x \to -\infty$ (loosely, this means $f(x)$ resembles a horizontal line as x approaches $-\infty$). Mathematically, the lower solid region case of Problem P2-2 may be stated as:

<u>Problem P2-4:</u> Determine $T(x,r)$ such that

$$\Delta T = PT_x, \quad 0 < r < 1 \text{ and } x < 0 \tag{2.3.1}$$

$$T(0,r) = A(r), \quad 0 < r < 1 \tag{2.3.2}$$

$$T(x,1) = f(x), \quad x < 0 \tag{2.3.3}$$

$$T_r(x,0) = 0, \quad x < 0 \tag{2.3.4}$$

and

$$\lim_{x \to -\infty} \max_{0<r<1} \left| f(x) - T(x,r) \right| = 0 \tag{2.3.5}$$

The functions $A(r)$ and $f(x)$ are assumed sufficiently smooth, and for compatibility, $A(1) = f(0)$. In addition, $\lim_{x \to -\infty} f(x)$ exists and is finite and both f' and f'' approach zero as $x \to -\infty$. The constant P is assumed to be positive (the subscript "s" is suppressed for convenience).

The notation established in Section 2.2 will be retained with the exception of $G(x)$ which now represents $G(x) = Pf'(x) - f''(x)$. The solution technique is very similar to that used in Section 2.2 First, $T(x,r)$ is expressed as

$$T(x,r) = \theta(x,r) + f(x)$$

and Equations (2.3.1) - (2.3.5) are recast as:

$$\Delta\theta = P\theta_x + G, \quad 0 < r < 1 \text{ and } x < 0 \tag{2.3.6}$$

$$\theta(0,r) = \mathcal{A}(r), \quad 0 < r < 1 \tag{2.3.7}$$

$$\theta(x,1) = 0, \quad x < 0 \tag{2.3.8}$$

$$\theta_r(x,0) = 0, \quad x < 0 \tag{2.3.9}$$

and

$$\lim_{x \to -\infty} \max_{0<r<1} \left| \theta(x,r) \right| = 0 \tag{2.3.10}$$

ORIGINAL PAGE IS
OF POOR QUALITY

The boundary value problem (BVP) given by the Equations (2.3.6) - (2.3.10) is solved in a manner similar to the solution of the BVP (2.2.6) - (2.2.10). If $\mathcal{A}(r)$ is represented as in Equation (2.2.17), then the BVP (2.3.6) - (2.3.10) may be transformed by the dual integral transform pair (2.2.14) into the following boundary value problem:

$$\overline{\theta}_n'' - P\overline{\theta}_n' - \lambda_n^2\overline{\theta}_n = \overline{G}_n \ , x < 0$$

$$\overline{\theta}_n(0) = \mathcal{A}_n \frac{J_1^2(\lambda_n)}{2}$$

and

$$\lim_{x \to -\infty} \overline{\theta}_n(x) = 0$$

where $\overline{G}_n$ is still defined as in (2.2.16). Using a variation of parameters technique, the solution of this BVP is given by

$$\left.\begin{aligned} \overline{\theta}_n(x) = &\left[A_n + \frac{1}{S_n}\int_0^x \overline{G}_n e^{-\alpha_n t}\, dt\right] e^{\alpha_n x} \\ &+ \left[B_n - \frac{1}{S_n}\int_0^x \overline{G}_n e^{-\beta_n t}\, dt\right] e^{\beta_n x} \end{aligned}\right\} \tag{2.3.11}$$

where

$$S_n = \sqrt{P^2 + 4\lambda_n^2}$$

$$\alpha_n = (P + S_n)/2$$

$$\beta_n = (P - S_n)/2$$

$$B_n = \frac{-1}{S_n}\int_{-\infty}^0 \overline{G}_n e^{-\beta_n t}\, dt$$

and

$$A_n = - B_n + \mathcal{A}_n \frac{J_1^2(\lambda_n)}{2}$$

Since each summand in (2.3.11) is the product of an exponentially exploding and exponentially decaying term, the proof that $\lim_{x \to -\infty} \overline{\theta}_n(x) = 0$ is rather delicate and is reserved for Appendix B. The solution of Problem P2-4 is

ORIGINAL PAGE IS
OF POOR QUALITY

$$T(x,r) = f(x) + 2\sum_{n=1}^{\infty} \frac{\psi_n(r)}{J_1^2(\lambda_n)} \bar{\theta}_n(x) \tag{2.3.12}$$

Since the float zone process also involves the upper solid region of Figure 1-2, an upper region analog of Problem P2-4 must be solved. After translating the upper interface to $x=0$ for convenience, the mathematical statement of such a problem is:

Problem P2-5: Determine $T(x,r)$ such that

$$\Delta T = PT_x \ , \ 0 < r < 1 \ , \ x > 0 \tag{2.3.13}$$

$$T(0,r) = A(r) \ , \ 0 < r < 1 \tag{2.3.14}$$

$$T(x,1) = g(x) \ , \ x > 0 \tag{2.3.15}$$

$$T_r(x,0) = 0 \ , \ x > 0 \tag{2.3.16}$$

and

$$\lim_{x\to\infty} \max_{0<r<1} |g(x) - T(x,r)| = 0 \tag{2.3.17}$$

As before, $A(r)$ and $g(x)$ are assumed sufficiently smooth and, for compatibility, $A(1) = g(0)$. In addition, $\lim_{x\to\infty} g(x)$ exists and is finite, both g' and g'' approach zero as $x\to\infty$, and $P>0$.

Without belaboring the details, the solution of Problem P2-5 is given by Equation (2.3.12) ($g(x)$ obviously replaces $f(x)$) where $\bar{\theta}_n(x)$ is still represented by Equation (2.3.11) with the $\bar{G}_n, S_n, \alpha_n$ and β_n unchanged but with new A_n and B_n, namely

$$A_n = \frac{-1}{S_n} \int_0^{\infty} \bar{G}_n e^{-\alpha_n t} \, dt$$

and

$$B_n = -A_n + \mathcal{A}_n \frac{J_1^2(\lambda_n)}{2}$$

As a computational aside, the numerical method described in Section 2.2 can be used to approximate the analytically defined solutions of this section. For example, in the upper solid region case, if the surface temperature $g(x)$ is rather constant for x, say, greater than some L, then the solution of Problem P2-5 may be approximated for $0 < x < x_N$ by the solution of Problem P2-3 with x_N set to, say 3L, and $B(r) = g(x_N)$ and $h(x) = g(x)$. Moreover, the gradient at the translated bottom, $x = 0$, of the upper solid region may be accurately estimated by the approximate gradient generated by Equations (2.2.21) and (2.2.22).

Numerous test cases to numerically verify the above remarks were generated with $A(r) = 0$ (to simulate a solid-melt interface), $0.1 \leq P \leq 1.0$, and $x_N = 2L$ and $3L$ (L typically on the order of 10). The approximate temperatures (and gradients) so obtained were quite accurate.

2.4 AN UNSETTLING FACT WITH AN ADDED NICE SURPRISE

Consider for the moment Problem P2-4. If $A(r) = 0$ (to simulate a solid-melt interface) and two temperature distributions T_1 and T_2 are generated corresponding to two surface temperature conditions $f_1(x)$ and $f_2(x)$, then if $f_1 \approx f_2$, it is reasonable to expect $T_1 \approx T_2$. In addition, if $f_1' \approx f_2'$ near $x = 0$, then it is also reasonable to expect that the corresponding thermal gradients of T_1 and T_2 will be close at $x = 0$. These intuitive observations are indeed true and may be rigorously proven after such concepts as "close" are precisely defined. All of this, however, might lead to the assumption that if f_1 and f_2 are not close, then the corresponding thermal gradients and temperature distributions near the simulated solid-melt interface ($x = 0$) are probably not close. This, of course, is not always true, and will be illustrated in this section by two examples. In fact, the second example will demonstrate the somewhat unsettling fact that it is quite possible for $f_1(x)$ to exponentially explode while $f_2(x)$ remains nicely bounded with the corresponding thermal gradients at $x = 0$ virtually indistinguishable. In light of the development presented in Section 1.2, this implies there might exist many varied surface control functions, all of which provide the required (or nearly so) thermal gradient at the desired interface. If this is the case, then the float zone furnace designer may have at his disposal many different prospective surface control functions to choose from (a nice surprise). For example, the designer might select a surface control function that requires a minimum of power.

The two examples in this section clearly demonstrate that small changes in the thermal gradient at the end boundary ($x = 0$) can result in a rather large change in the resulting surface control function. This, as noted before, can provide an entire family of useful surface control functions if the FZ designer is willing to permit a slight "misfit" (albeit small) between the desired and obtained thermal gradients at the end boundary ($x = 0$ for the following examples). Unfortunately, this also means that an attempt to measure the sensitivity of the required surface control functions to changes in the material or system parameters (which obviously produce changes in the desired interface thermal gradient) can be quite misleading and should probably not be attempted.

<u>Example E2-1:</u> In this example, $P = 0.1$, $A(r) = 0$ and the lower solid region case ($x \leq 0$) is selected. The nominal surface temperature f_1 is illustrated in Figure 2-3; the surface temperatures $f_2, \ldots, f_5$ (also illustrated in Figure 2-3) are perturbations of the nominal f_1.

Letting T_i denote the thermal distributions corresponding to the surface temperatures f_i, the relative difference (measured in both L^2 and L^∞ norms†) between the nominal gradient of T_1 and each of the gradients of T_i, $i = 2, \ldots, 5$, at the end boundary, $x = 0$, is illustrated in Figure 2-4.

† For a function $h(r)$, $0 \leq r \leq 1$, the L^2 and L^∞ norms are (respectively)

$$\|h\|_2 = \left[\int_0^1 h^2(r)\,dr\right]^{\frac{1}{2}} \text{ and } \|h\|_\infty = \max_{0 \leq r \leq 1} |h(r)|.$$

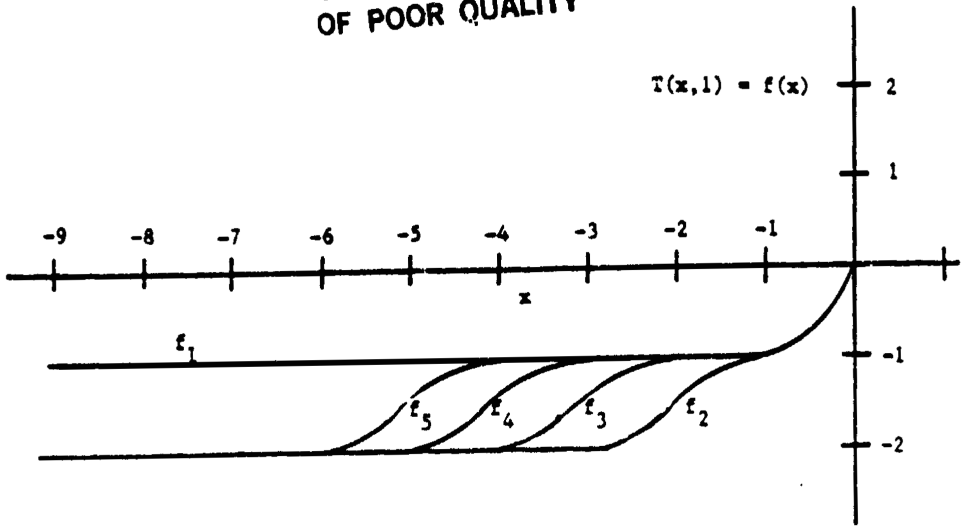


Figure 2-3 Nominal and Perturbed Surface Temperatures

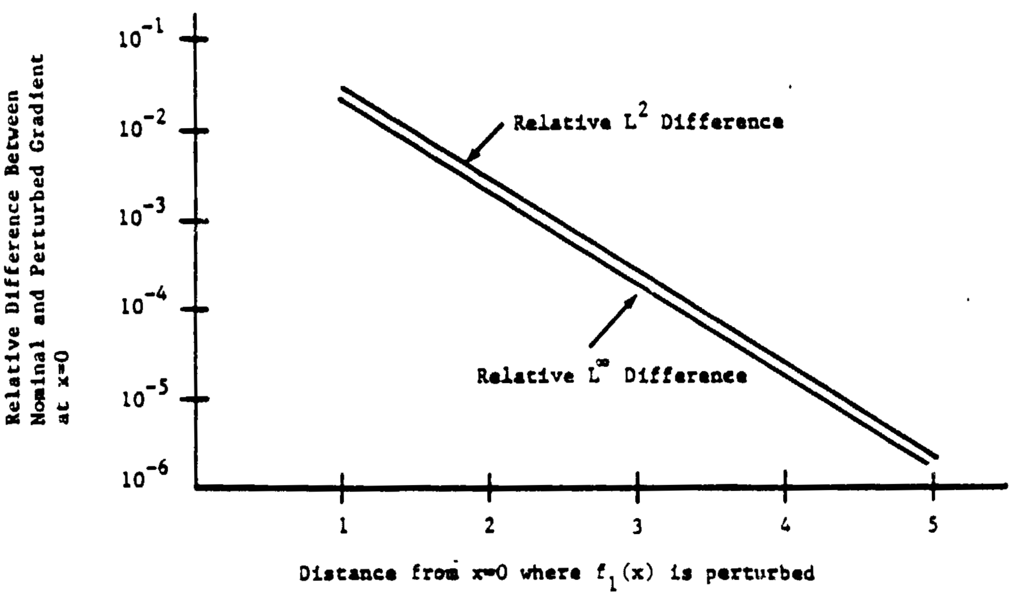


Figure 2-4 Influence of Perturbations of the Surface Temperature on the Thermal Gradient

Note that even for the cases where f_1, the nominal surface temperature, is perturbed relatively close to the end boundary ($x = 0$), the corresponding perturbations of the thermal gradients at the $x = 0$ boundary are still remarkably close to that of the nominal gradient.

<u>Example E2-2:</u> In this example, $P = 0.1$, $A(r) = 0$, and the upper solid region (translated to $x \geq 0$) is selected. Suppose it is required that the thermal gradient at $x = 0$ be identically 1, i.e., $T_x(0,r) = 1$. A particular surface control function $g_1(x)$ which will give the desired result is the exponentially growing surface temperature:

$$g_1(x) = -10 + 10\mathrm{EXP}(x/10),\ x>0$$

In fact, the corresponding thermal distribution T_1 is identicall[illegible]ual to g_1. Suppose $g_2(x)$, ..., $g_5(x)$ are surface control functions that eq[illegible](x) on an interval $[0,z_i]$, $i = 2, \ldots, 5$ but are asymptotically constant as [illegible] grows (see Figure 2-5.) The relative L^2 and L^∞ differences between the thermal gradients at $x = 0$ of the corresponding temperature distributions T_2, ..., T_5 and the thermal gradient of T_1 is illustrated in Figure 2-6. Note that even when g_2 (a bounded surface temperature) separates from g_1 (an unbounded surface temperature) rather close to the end boundary ($x=0$), the two corresponding thermal gradients at $x = 0$ are remarkably close (see Figure [illegible]-6, $z = 0.5$).

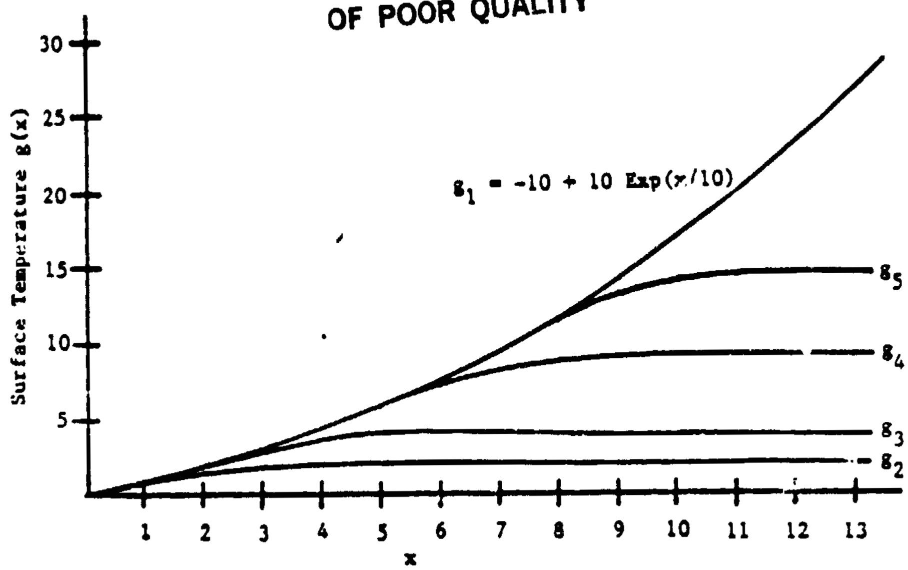


Figure 2-5 Nominal and Perturbed Surface Temperatures

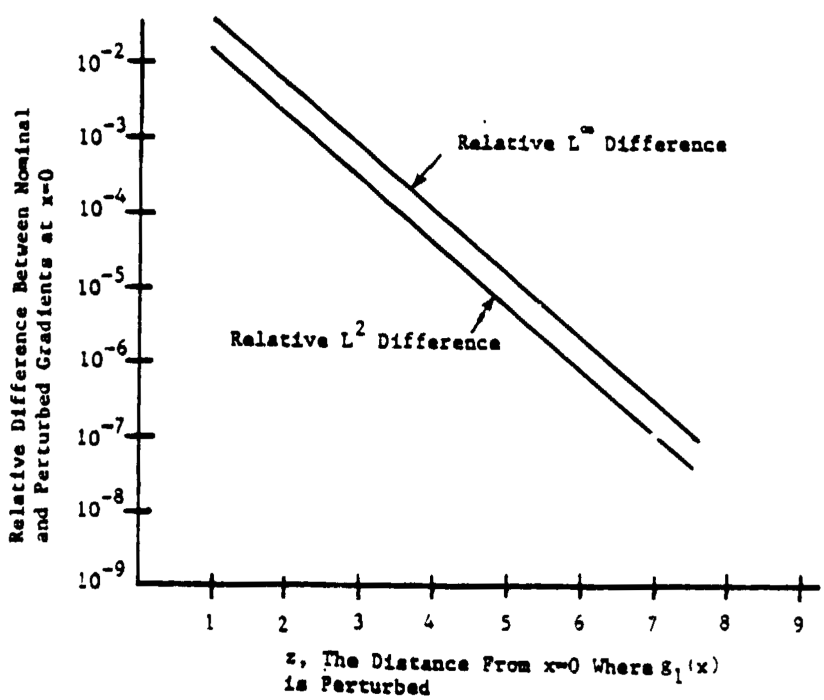


Figure 2-6 Influence of Perturbations of the Surface Temperature on the Thermal Gradients

3.0 THE COOLING CONTROL FUNCTIONS

It is an old maxim of mine that when
you have excluded the impossible,
whatever remains, however improbable,
must be the truth.

--Sherlock Holmes, "The Adventure of the Beryl Coronet"

The main thrust of this chapter is to provide solution methods for Problems P1-1 and P1-2. Beginning with Problem P1-1, suppose a temperature distribution $T(x,r)$ is required to satisfy two known boundary conditions

$$T(0,r) = A(r) \tag{3.0.1}$$

$$T_x(0,r) = B(r) \tag{3.0.2}$$

at the lower solid region's end boundary (the melt-solid interface in practice) as depicted in Figure 3-1.

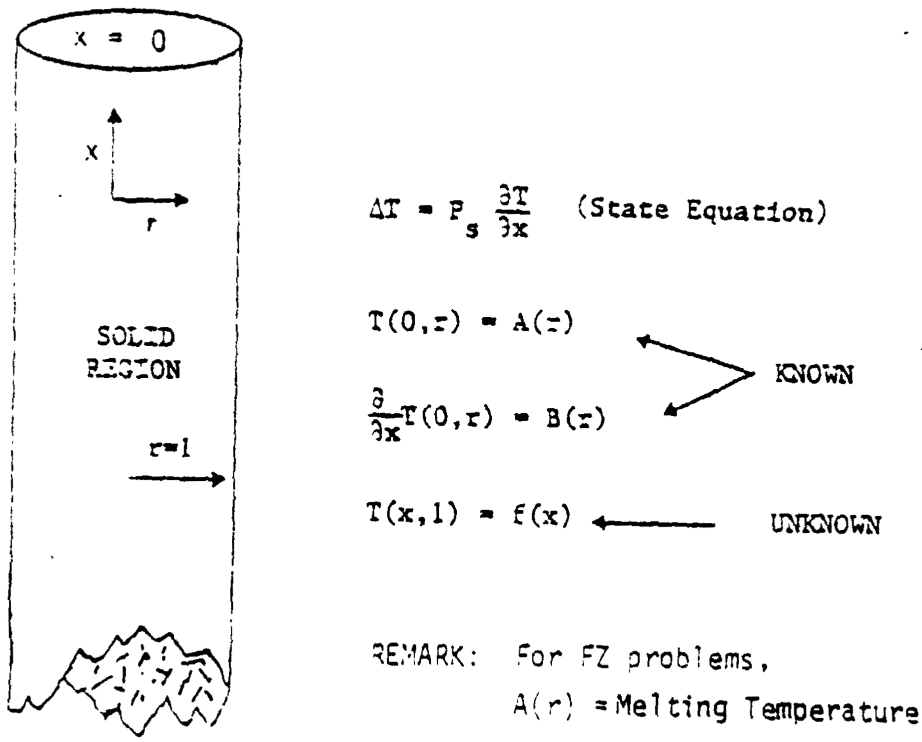


Figure 3-1 FZ Lower Solid Region Problem

ORIGINAL PAGE IS
OF POOR QUALITY

Recall from Section 1.2 that the problem is to find some (at this point unknown) cooling control function $f(x)$, $x\leq 0$, such that the solution of the well-posed boundary value problem:

$$\Delta T = P_s T_x \quad , x< 0 \quad , \quad 0 < r < 1 \tag{3.0.3}$$

$$T(0,r)=A(r) \quad , 0 < r < 1 \tag{3.0.4}$$

and

$$T(x,1)=f(x) \quad , x < 0 \tag{3.0.5}$$

also satisfies the addition boundary condition (3.0.2). The basic idea of the proposed method is to solve the boundary value problem (3.0.3)-(3.0.5) by the method described in Section 2.3 and, in the process find a sufficient number of conditions to allow the calculation of the desired, but unknown, $f(x)$. First, from a practical point of view, any viable control function $f(x)$ should become rather constant as the distance from the lower interface increases. Thus it is expected that:

$$\lim_{x\to-\infty} f(x) \text{ exists and is finite} \tag{3.0.6}$$

and

$$\lim_{x\to-\infty} f'(x) = \lim_{x\to-\infty} f''(x) = 0 \tag{3.0.7}$$

Proceeding as in Section 2.3, denote[†]

$$T(x,r) = \theta(x,r) + f(x)$$

$$G(x) = Pf' - f''$$

$$\mathcal{A}(r) = A(r) - f(0)$$

$$\mathcal{B}(r) = B(r) - f'(0)$$

For $f(x)$ to be compatible with $A(r)$ and $B(r)$,

$$A(1) = f(0) \tag{3.0.8}$$

and

$$B(1) = f'(0) \tag{3.0.9}$$

Then Equations (3.0.2)-(3.0.5) reduce to

[†]For the moment, suppress the solid subscript "s", i.e., $P_s=P$.

$$\left.\begin{aligned} \Delta\theta &= P\theta_x + G(x) \\ \theta(0,r) &= \mathcal{A}(r) \\ \theta_x(0,r) &= \mathcal{B}(r) \\ \theta(x,1) &= 0 \end{aligned}\right\} \qquad (3.0.10)$$

Denote $\psi_n(r) = J_o(\lambda_n r)$ where $\lambda_1 < \lambda_2 < \lambda_3 < \ldots$ are the real roots of the zero order Bessel function. As in Chapter 2, let

$$\theta(x,r) = \sum_{M=1}^{\infty} \frac{2\,\psi_M(r)}{J_1^2(\lambda_M)} \bar{\theta}_M(x) \qquad (3.0.11)$$

and

$$\bar{\theta}_M(x) = \int_0^1 \theta(x,r)\psi_M(r)rdr \qquad (3.0.12)$$

form a dual integral transform pair. If $\mathcal{A}(r)$ and $\mathcal{B}(r)$ are expanded in the Bessel series:

$$\mathcal{A}(r) = \sum_{n=1}^{\infty} \mathcal{A}_n J_o(\lambda_n r) \qquad (3.0.13)$$

$$\mathcal{B}(r) = \sum_{n=1}^{\infty} \mathcal{B}_n J_o(\lambda_n r) \qquad (3.0.14)$$

and denoting

$$\bar{G}_M(x) = \int_0^1 G(x)\,\psi_M(r)rdr = G(x)\,\frac{J_1(\lambda_M)}{\lambda_M} \qquad (3.0.15)$$

then operating on (3.0.10) by the integral transform (3.0.12) yields

$$-\lambda_M^2\,\bar{\theta}_M + \bar{\theta}_M'' = P\bar{\theta}_M' + \bar{G}_M\ ,\ x < 0 \qquad (3.0.16)$$

$$\bar{\theta}_M(0) = \mathcal{A}_M\ \frac{J_1^2(\lambda_M)}{2} \qquad (3.0.17)$$

and

$$\bar{\theta}'_M(0) = \mathcal{B}_M \frac{J_1^2(\lambda_M)}{2} \qquad (3.0.18)$$

In practice, the Bessel coefficients, $\mathcal{A}_n$ and $\mathcal{B}_n$ in (3.0.13) and (3.0.14) are approximated by least square methods as described in Section 2.2. In addition, for FZ applications $\mathcal{A}_M = 0$ because A(r) is constant (the material melting temperature). Denoting

$$S_M = \sqrt{P^2 + 4\lambda_M^2}$$

$$\alpha_M = (P + S_M)/2$$

and

$$\beta_M = (P - S_M)/2,$$

the solution of (3.0.16)-(3.0.18) is then:

$$\begin{aligned}\bar{\theta}_M(x) = {} & \frac{J_1^2(\lambda_M)}{2\,S_M}\left(\mathcal{B}_M - \beta_M \mathcal{A}_M\right) e^{\alpha_M x} \qquad (3.0.19)\\ & + e^{\alpha_M x} \int_0^x \frac{\bar{G}_M(t)}{S_M} e^{-\alpha_M t}\,dt \\ & + \frac{J_1^2(\lambda_M)}{2\,S_M}\left(\alpha_M \mathcal{A}_M - \mathcal{B}_M\right) e^{\beta_M x} \\ & - e^{\beta_M x} \int_0^x \frac{\bar{G}_M(t)}{S_M} e^{-\beta_M t}\,dt\end{aligned}$$

Since $\bar{G}_M(x) = (Pf'(x) - f''(x))\,J_1^2(\lambda_M)/2$ approaches zero as x proceeds toward negative infinity (see (3.0.7)), an argument similar to that found in Appendix B will show that the second summand in (3.0.19) approaches zero as x approaches negative infinity; since α_M is positive, the first summand of (3.0.19) shares a similar fate. In light of (3.0.6), it is reasonable to assume (or require depending on the point of view) that

$$\lim_{x \to -\infty} \max_{0<r<1} \left| T(x,r) - f(x) \right| = 0$$

and hence $\lim_{x \to -\infty} \bar{\theta}_n(x) = 0$. Combining these observations with (3.0.19), the remaining conditions to be used in determining $f(x)$ are easily discerned, namely:

$$\lim_{x \to -\infty} \left[\frac{J_1^2(\lambda_M)}{2\, S_M} \left(\alpha_M \mathcal{A}_M - \mathcal{B}_M \right) - \int_0^x \frac{\bar{G}_M(t)}{S_M} e^{-\beta_M t} dt \right] e^{\beta_M x} = 0 \quad (3.0.20)$$

Since $\beta_M < 0$, an analysis similar to that of Appendix B will show that (3.0.20) will be satisfied if

$$\frac{J_1^2(\lambda_M)}{2} (\alpha_M \mathcal{A}_M - \mathcal{B}_M) = -\int_{-\infty}^{0} \bar{G}_M(t) e^{-\beta_M t} dt \qquad (3.0.21)$$

Since

$$\bar{G}_M(t) = G(t) J_1(\lambda_M)/\lambda_M = (Pf'(t) - f''(t)) J_1(\lambda_M)/\lambda_M,$$

combining (3.0.6)-(3.0.9), and (3.0.21) with two applications of integration by parts yields:

$$\lambda_M J_1(\lambda_m) (\alpha_M \mathcal{A}_M - \mathcal{B}_M)/2$$

$$= \beta_M(\beta_M - P) \int_{-\infty}^{0} f(t) e^{-\beta_m t} dt + (\beta_M - P) A(1) + B(1)$$

Denoting

$$R_M = \frac{\frac{1}{2}\lambda_m J_1(\lambda_m)(\alpha_m \mathcal{A}_m - \mathcal{B}_m) + (P - \beta_M) A(1) - B(1)}{\beta_m(\beta_m - P)}$$

the desired properties of the surface function $f(x)$ may be summarized as:

ORIGINAL PAGE IS
OF POOR QUALITY

$$\left.\begin{aligned}
&f(0) = A(1) \text{ and } f'(0) = B(1) \\
&\lim_{x\to-\infty} f(x) \text{ exists and is finite} \\
&\text{and } f'(x) \text{ and } f''(x) \to 0 \text{ as } x \to -\infty \\
&R_M = \int_{-\infty}^{0} f(t)e^{-\beta_M t}\,dt \;,\; M = 1,2,\cdots
\end{aligned}\right\} \tag{3.0.22}$$

To numerically approximate such a surface control function as f(x), let†

$$f(x) \approx \sum_{k=1}^{NSYS} c_k e^{(k-1)t} \tag{3.0.23}$$

Then, in light of (3.0.22), set

$$c_1 + \cdots + c_{NSYS} = A(1)$$

$$c_2 + 2c_3 + \cdots + (NSYS-1)c_{NSYS} = B(1)$$

and

$$\sum_{k=1}^{NSYS} c_k \int_{-\infty}^{0} \mathrm{Exp}((k-1-\beta_M)t)dt = R_M \quad , M=1,2,\ldots,MTERM$$

If the (MTERM+2) by NSYS matrix L and (MTERM+2) dimension vector $\overline{b}$ are defined, for j = 1,2,..., NSYS, by

$$\left.\begin{aligned}
&\ell_{1j} = 1 \text{ and } b_1 = A(1) \\
&\ell_{2j} = j-1 \text{ and } b_2 = B(1) \\
&\left.\begin{aligned}
&\ell_{ij} = \frac{1}{j-\beta_{i-2}-1} \\
&b_i = R_{i-2}
\end{aligned}\right\} \quad i = 3,4,\ \ldots,\ MTERM+2
\end{aligned}\right\} \tag{3.0.24}$$

† The index in the expansion of f(x) starts at k=1 instead of k=0 to make referencing this section from the accompanying FORTRAN documentation easier (Appendix A). The index limits NSYS and MTERM noted here will be used in the same role in the accompanying FORTRAN codes (Appendix C).

then, provided MTERM+2 > NSYS, the coefficients c_k of (3.0.23) may be set to the least squares solution of

$$L\bar{C} = \bar{b} \tag{3.0.25}$$

that is,

$$\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{NSYS} \end{pmatrix} = \bar{C}$$

The solution method for Problem P1-2 is similar to the above and hence most of the details are left to the reader. Using the notation established for Problem P1-2 denote

$$G(x) = Pg'(x) - g''(x)$$

$$\mathcal{A}(r) = A(r) - A(1)$$

and

$$\mathcal{B}(r) = B(r) - B(1)$$

As before, expand $\mathcal{A}(r)$ and $\mathcal{B}(r)$ as

$$\mathcal{A}(r) = \sum_{n=1}^{\infty} \mathcal{A}_n J_o(\lambda_n r)$$

and

$$\mathcal{B}(r) = \sum_{n=1}^{\infty} \mathcal{B}_n J_o(\lambda_n r)$$

Then using the above established notation for S_n, α_n, and β_n, the desirable properties of an upper solid region surface control function $g(x)$ are summarized as:

$$g(Q) = A(1) \tag{3.0.26}$$

$$g'(Q) = B(1) \tag{3.0.27}$$

$$\frac{\lambda_n J_1(\lambda_n)}{2} \left(\beta_n \mathcal{A}_n - \mathcal{B}_n\right) = \int_Q^{\infty} G(t) e^{\alpha_n (Q-t)} dt \tag{3.0.28}$$

$$\lim_{x \to \infty} g(x) \text{ exists and is finite} \tag{3.0.29}$$

and

$$\lim_{x \to \infty} g'(x) = \lim_{x \to \infty} g''(x) = 0 \tag{3.0.30}$$

Mimicking the previous analysis, (3.0.28) is simplified by two applications of integration by parts. Thereafter, an approximation of g(x) given by:

$$g(x) \approx \sum_{k=1}^{NSYS} c_k e^{(k-1)(Q-x)} \tag{3.0.31}$$

is substituted into Equations (3.0.26)-(3.0.28) and the desired coefficients determined by a least squares method.

4.0 THE HEATING CONTROL FUNCTION

In five minutes you will say that it
is all so absurdly simple.

—Sherlock Holmes, "The Adventure of the Dancing Man"

The ultimate goal of this chapter is the solution of Problem P1-3. Suppose that the temperature distribution $T(x,r)$ for the melt zone is required to not only satisfy the state equation

$$\Delta T = P_\ell T \quad , \quad 0 < x < Q \quad , \quad 0 < r < 1 \tag{4.0.1}$$

but also must satisfy for $0<r<1$ the four boundary conditions:

$$T(0,r) = C(r) \tag{4.0.2}$$

$$T(Q,r) = D(r) \tag{4.0.3}$$

$$T_x(0,r) = A(r) \tag{4.0.4}$$

and

$$T_x(Q,r) = B(r) \tag{4.0.5}$$

Unfortunately, not only is too much information supplied for the two end boundaries ($x=0$ and $x=Q$; see Figure 4-1), no information whatsoever is supplied for the remaining boundary, $r = 1$ (again see Figure 4-1).

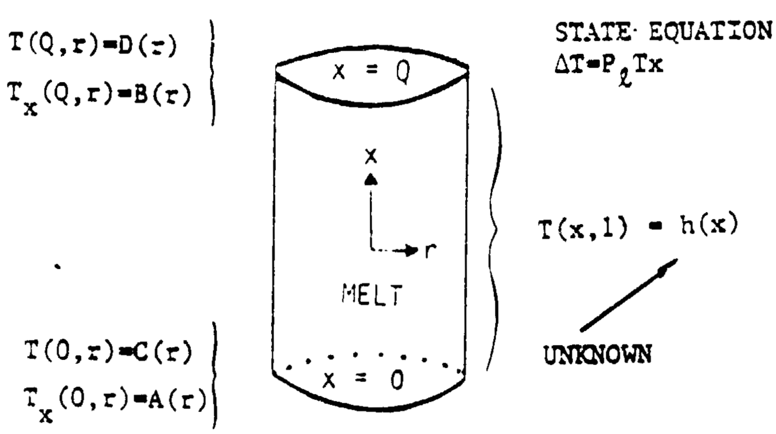


Figure 4-1 FZ Melt Zone Problem

The problem, therefore, is to find some heating control function $h(x)$, $0 \le x \le Q$, such that the solution $T(x,r)$ of the well-posed problem defined by the boundary condition $T(x,1) = h(x)$, the boundary conditions (4.0.2) and (4.0.3) and the state equation (4.0.1) also satisfies (or nearly so) the additional conditions (4.0.4) and (4.0.5).

For simplicity, the functions $C(r)$ and $D(r)$ are both assumed to be zero †. The generalization for nonconstant $C(r)$ or $D(r)$ is similar to the following analysis and is left to the interested reader. As in Chapters 2 and 3, define

$$G(x) = P_\ell h'(x) - h''(x) \qquad (4.0.6)$$

$$\mathcal{A}(r) = A(r) - A(1) = \sum_{n=1}^{\infty} \mathcal{A}_n J_o(\lambda_n r) \qquad (4.0.7)$$

and

$$\mathcal{B}(r) = B(r) - B(1) = \sum_{n=1}^{\infty} \mathcal{B}_n J_o(\lambda_n r) \qquad (4.0.8)$$

If $T(x,r)$ is decomposed into

$$T(x,r) = \theta(x,r) + h(x) \qquad (4.0.9)$$

then Equations (4.0.1)-(4.0.3) imply*

$$\left.\begin{aligned} &\Delta\theta = P\theta_x + G \ , \ 0 < x < Q \ , \ 0 < r < 1 \\ &\theta(0,r) = \theta(Q,r) = 0 \ , \ 0 < r < 1 \end{aligned}\right\} \qquad (4.0.10)$$

Denoting $\overline{G}_n(x) = G(x) \cdot J_1(\lambda_n)/\lambda_n$ and transforming (4.0.10) by the integral transform (3.0.12),

$$\left.\begin{aligned} &\overline{\theta}_n'' - P\overline{\theta}_n' - \lambda_n^2 \overline{\theta}_n = \overline{G}_n \ , \ 0 < x < Q \\ &\overline{\theta}_n(0) = \overline{\theta}_n(Q) = 0 \end{aligned}\right\} \qquad (4.0.11)$$

† For FZ work, both $C(r)$ and $D(r)$ are set to the material melting temperature which can itself always be assigned to be zero on some translated temperature scale.

*For convenience, suppress the "ℓ" (liquid) subscript, i.e., $P_\ell = P$

For convenience, let

$$S_n = \sqrt{P^2 + 4\lambda_n^2}$$

$$\alpha_n = (P + S_n)/2$$

and

$$\beta_n = (P - S_n)/2,$$

By a variation of parameters method, the solution of (4.0.11) is

$$\overline{\theta}_n(x) = \left[K_n - \frac{1}{S_n}\int_0^x \overline{G}_n(t)e^{-\beta_n t}dt\right]e^{\beta_n x} + \left[-K_n + \frac{1}{S_n}\int_0^x \overline{G}_n(t)e^{-\alpha_n t}dt\right]e^{\alpha_n x} \tag{4.0.12}$$

where

$$K_n = \frac{\int_0^Q \overline{G}_n(t)\left[e^{\alpha_n(Q-t)} - e^{\beta_n(Q-t)}\right]dt}{S_n\left[e^{\alpha_n Q} - e^{\beta_n Q}\right]}$$

For the desired h(x) to be compatible with Equations (4.0.2)-(4.0.5) (recall C(r) and D(r) are set to zero), $h(0) = 0 = h(Q)$, $h'(0) = A(1)$ and $h'(Q) = B(1)$. In addition, since $\theta_x(0,r) = \mathcal{A}(r)$ and $\theta_x(Q,r) = \mathcal{B}(r)$,

$$\left.\begin{aligned} \overline{\theta}_n'(0) &= \mathcal{A}_n J_1^2(\lambda_n)/2 \\ \overline{\theta}_n'(Q) &= \mathcal{B}_n J_1^2(\lambda_n)/2 \end{aligned}\right\} \tag{4.0.13}$$

ORIGINAL PAGE IS
OF POOR QUALITY

Combining Equations (4.0.13) with the derivative of $\overline{\theta}_n(x)$ (obtaining by differentiating† (4.0.12)) yields

$$-\frac{\mathcal{A}_n J_1(\lambda_n)\lambda_n}{2} - A(1) = \int_0^Q K_1(n,t)\, h(t)\, dt \qquad (4.0.14)$$

and

$$\frac{\lambda_n \mathcal{B}_n J_1(\lambda_n)}{2} + B(1) = \int_0^Q K_2(n,t)\, h(t)\, dt \qquad (4.0.15)$$

where, if C_n denotes,

$$C_n = \frac{1}{4}\,(P^2 - S_n^2)/\left[1 - \mathrm{Exp}(-S_n Q)\right]$$

then kernels K_1 and K_2 in Equations (4.0.14) and (4.0.15) are defined by

$$K_1(n,t) = C_n \left[e^{-\alpha_n t} - e^{-(\beta_n t + S_n Q)} \right] \qquad (4.0.16)$$

and

$$K_2(n,t) = C_n \left(1 - e^{-S_n t}\right) e^{\beta_n (t-Q)} \qquad (4.0.17)$$

† The actual process of differentiating (4.0.12) is routine but laborious and is left to the industrious reader. However, this is not to imply that great care should not be taken; several of the integrands are the difference of large functions (a numerically delicate situation). For the industrious reader willing to check these results, the removal of derivatives from integrands by integration by parts is necessary.

ORIGINAL PAGE IS
OF POOR QUALITY

The desired h(x) is numerically approximated by some expansion of the form:

$$h(x) \approx \sum_{k=1}^{NSYS} c_k h_k(x) \tag{4.0.18}$$

(for example, let $h_k(x)=x^{k-1}$). To finish this development, solve for the coefficients c_k in Equation (4.0.18) by solving (in a least squares sense†) the System (4.0.19)-(4.0.23) given below.

$$\sum_{k=1}^{NSYS} c_k h_k(0) = 0 \tag{4.0.19}$$

$$\sum_{k=1}^{NSYS} c_k h_k(Q) = 0 \tag{4.0.20}$$

$$\sum_{k=1}^{NSYS} c_k h_k'(Q) = B(1) \tag{4.0.21}$$

$$\sum_{k=1}^{NSYS} c_k h_k'(0) = A(1) \tag{4.0.22}$$

$$\sum_{k=1}^{NSYS} a_{jnk}\, c_k = b_{jn} \quad , n=1,2,\cdots, \text{MTERM, and } j = 1,2 \tag{4.0.23}$$

where

$$a_{jnk} = \int_0^Q K_j(n,t)\ h_k(t)\, dt$$

$$b_{1n} = -\frac{1}{2} \mathcal{A}_n J_1(\lambda_n)\lambda_n - A(1)$$

and

$$b_{2n} = \frac{1}{2} \mathcal{B}_n J_1(\lambda_n)\lambda_n + B(1)$$

† In order to use a least squares method, the index limits NSYS and MTERM should be selected such that NSYS/2 < MTERM + 2.

5.0 TEST CASES

And here--ah, now, this really is
something a little recherche

--Sherlock Holmes, "The Musgrave Ritual"

5.1 SOLID REGION SURFACE CONTROL FUNCTIONS

In this section, the methodology developed in Chapter 3 is illustrated using material and system data provided by NASA. Recall that the problem is to find, after being given the melt zone surface temperature distribution, the surface control functions for the solid regions which are compatible with flat interfaces. The material and system parameters used are listed in Table 5-1 and were provided (and in some cases appropriately modified) by E. Kern (NASA contractor, [11]) and E. Cothran (NASA, [4]). The material selected was silicon.

TABLE 5-1 MATERIAL AND SYSTEM PARAMETERS

PARAMETER	VALUE
Radius	0.2413 cm
Melt Length	1.1684 cm
Conductivity	
Solid	7.5 cal/^{o}K m sec
Melt	16 cal/^{o}K m sec
Density	
Solid	2.28 gm/cm^3
Melt	2.53 gm/cm^3
Heat Capacity	
Solid	0.241 cal/^{o}K gm
Melt	0.265 cal/^{o}K gm
Latent Heat	431 cal/gm
Growth Rate	2.5 mm/min
Peclet Number	
Solid	0.00734
Melt	0.00421
Melting Temperature	1693^{o} K

The melt zone surface temperature distribution used was suggested by E. Kern [11] and is illustrated in Figure 5-1.

The surface control functions for various combinations of MTERM and NSYS (see Equations (3.0.23) and (3.0.24)) obtained by the methods of Chapter 3 are illustrated in Figure 5-2.

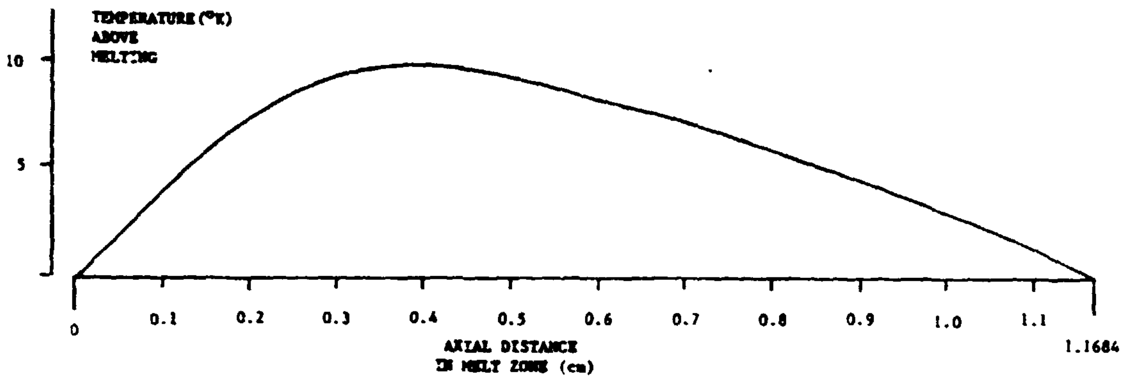
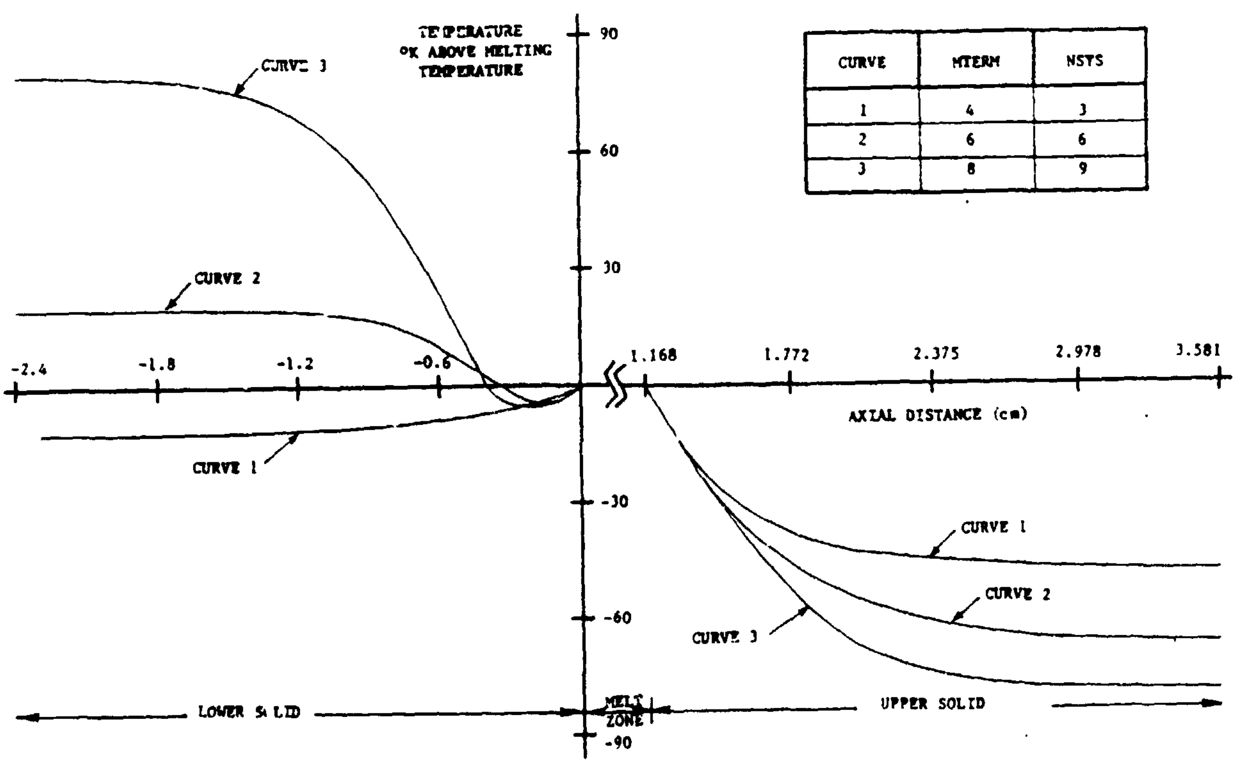


Figure 5-1 Melt Zone Surface Temperature Distribution



CURVE	MTERM	NSYS
1	4	3
2	6	6
3	8	9

Figure 5-2 Solid Regions' Surface Control Functions

ORIGINAL PAGE IS
OF POOR QUALITY

In light of Section 2.4, the variety of control functions depicted in Figure 5-2 is expected. In addition, the results of Section 2.4 suggest that the two lower solid surface control functions that eventually are above the melting temperature may be modified as illustrated in Figure 5-3 without substantially changing the thermal gradients at $x=0$.

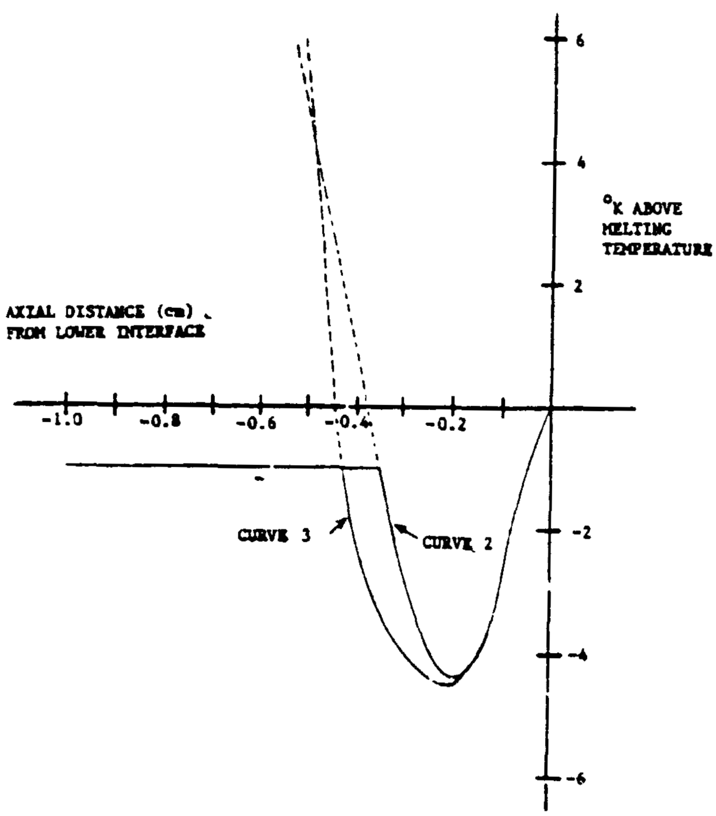
The relative differences between the thermal gradients (in the solid regions) required† at the interfaces ($x=0.0$ cm and 1.1684 cm) and those obtained using the surface control functions defined in Figures 5-2 and 5-3 are listed in Table 5-2

TABLE 5-2 RELATIVE DIFFERENCES BETWEEN THE REQUIRED INTERFACE GRADIENTS AND THOSE RESULTING FROM THE USE OF THE SOLID REGIONS' SURFACE CONTROL FUNCTIONS

MTERM	MSYS	Solid Region	Surface Control Function Definition	Relative Difference (in L^2 norm)
4	3	Upper Lower	Figure 5-2 Figure 5-2	0.0175 0.17
6	6	Upper Lower Lower	Figure 5-2 Figure 5-2 Figure 5-3	0.00012 0.001 0.056
8	9	Upper Lower Lower	Figure 5-2 Figure 5-2 Figure 5-3	0.000027 0.0013 0.061

As an aside, in a series of test cases over a range of values for MTERM and NSYS, the relative differences between the required interface gradients and those obtained using surface control functions first decreased and then increased as NSYS (or MTERM) was increased while holding fixed the value of MTERM (or NSYS). Naturally, this is to be expected since an approximate solution of an ill-posed problem is attempted by employing an overposed system. This, of course, reinforces the old maxim of always examining a computed solution for "reasonableness." In fact, the computer software developed (see Appendix A) to determine the solutions of Problems P1-1 and P1-2 automatically computes the relative errors between the required interface gradients and those resulting from the use of the surface control functions. It cannot be overstated: always examine these relative errors before accepting a computed solution as reasonable.

† See the Boundary Conditions (1.2.1) and (1.2.3).



CURVE	MTERM	NSYS
2	6	6
3	8	9

Figure 5-3 Modification of Lower Solid Region's Surface Control Functions

5.2 MELT ZONE SURFACE CONTROL FUNCTIONS

The techniques developed in Chapter 4 are illustrated in this section using material and system data as provided by NASA. The problem is to determine a melt zone surface control function compatible with some solid regions' surface temperature distributions (provided a priori) such that flat melt-solid interfaces are achieved. The material and system parameters used are listed in Table 5-3 and were provided by E. Kern (NASA contractor [11] and E. Cothran (NASA [4]). The material selected was silicon.

TABLE 5-3 MATERIAL AND SYSTEM PARAMETERS

PARAMETER	VALUE
Crystal Radius	0.69 cm
Melt Length	1.43 cm
Conductivity	
Solid	7.5 cal/^{o}K m sec
Melt	16 cal/^{o}K m sec
Density	
Solid	2.28 gm/cm^3
Melt	2.53 gm/cm^3
Heat Capacity	
Solid	0.241 cal/^{o}K gm
Melt	0.265 cal/^{o}K gm
Latent Heat	431 cal/gm
Growth Rate	2 mm/min
Peclet Number	
Solid	0.01685
Melt	0.009638
Melting Temperature	1693^{o} K

The lower and upper solid regions' surface temperature distributions used were suggested by E. Kern [11] and are illustrated in Figure 5-4.

The melt zone surface control functions for various combinations of MTERM and NSYS (see Equations (4.0.18)-(4.0.23)) obtained by the methods of Chapter 4 are illustrated in Figure 5-5.

Because of the ill-posed nature of the problem, a variety of surface control functions is expected. The relative differences between the required† melt zone interface gradients and those obtained using the surface control functions defined in Figure 5-5 are listed in Table 5-4.

† See Boundary Condition (1.2.5)

ORIGINAL PAGE IS
OF POOR QUALITY

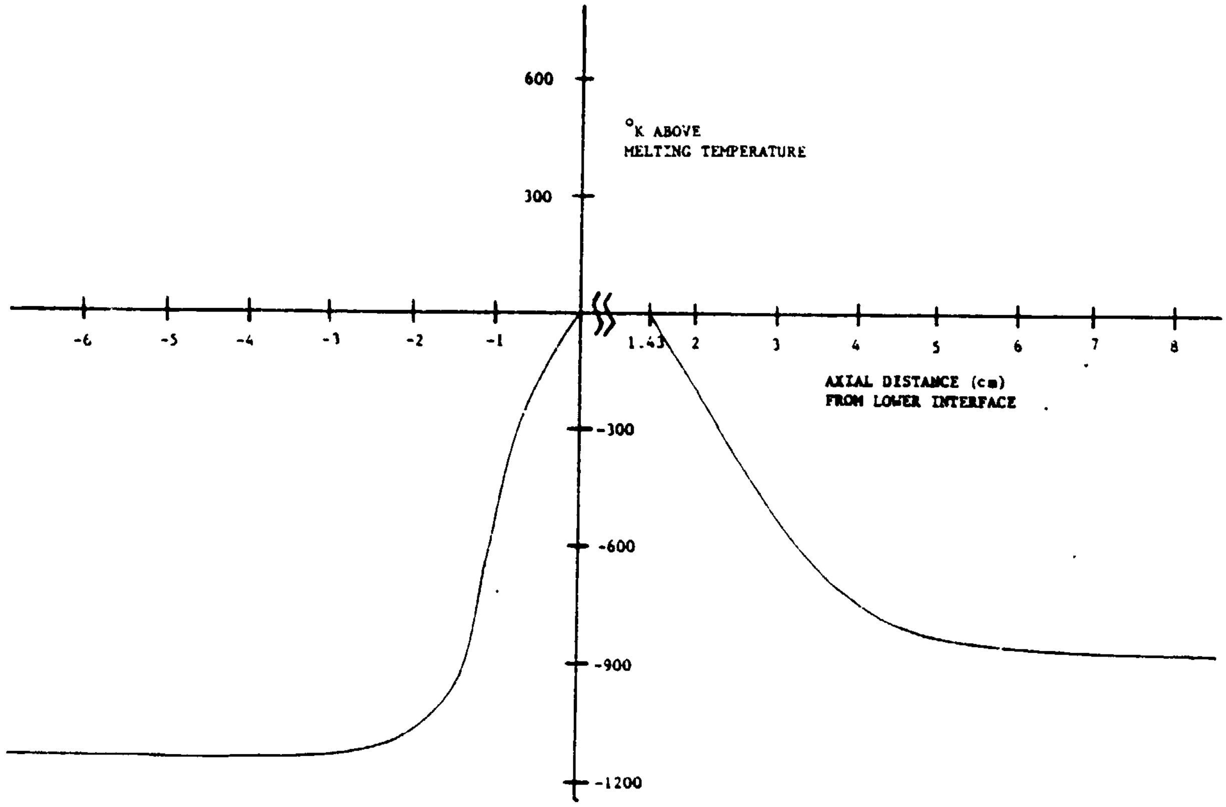


Figure 5-4 Solid Region's Surface Temperature

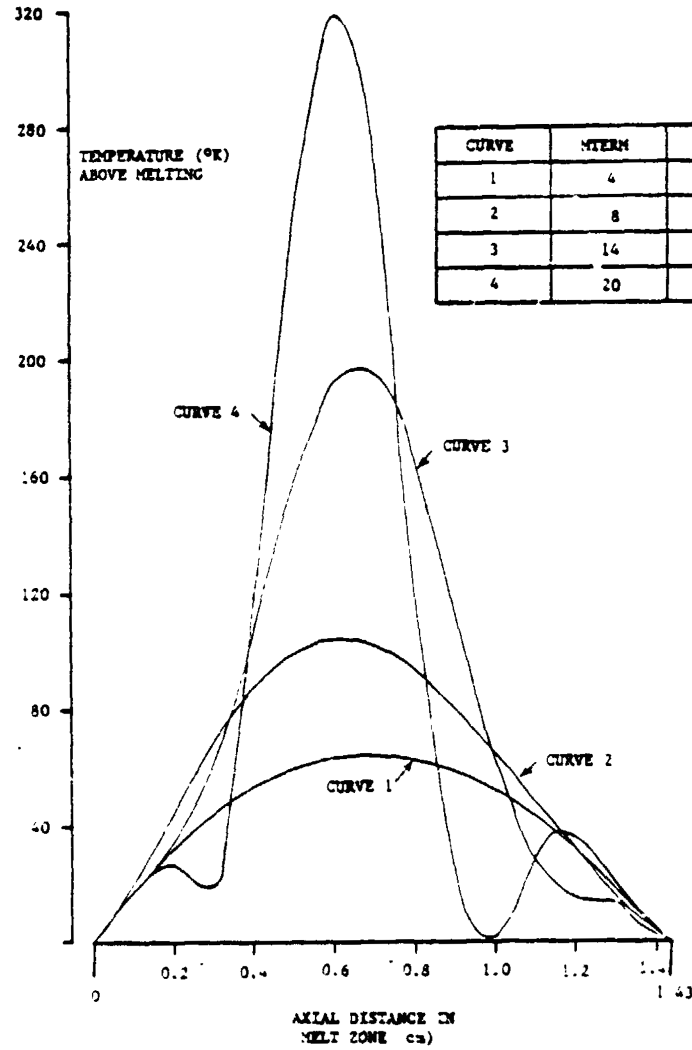


Figure 5-5 Melt Zone Surface
Control Function

TABLE 5-4 RELATIVE DIFFERENCES BETWEEN THE REQUIRED INTERFACE GRADIENTS AND THOSE RESULTING FROM THE USE OF THE MELT ZONE SURFACE CONTROL FUNCTION

MTERM	NSYS	MELT-SOLID INTERFACE	RELATIVE DIFFERENCE (in L^2 norm)
4	3	Upper Lower	.53 .24
8	6	Upper Lower	.28 .07
14	9	Upper Lower	.11 .13
20	12	Upper Lower	.05 .04

In a series of test cases over a range of values for MTERM and NSYS, the relative differences between the required interface gradients and those obtained using the surface control functions first decreased and then increased as NSYS (or MTERM) was increased while fixing the value of MTERM (or NSYS). As in the previous section, this is to be expected because the solution technique employed uses over-posed systems to approximately solve an ill-posed problem. As before, the computed melt zone surface control function should be checked for reasonableness. For example, it is quite possible that the surface control function could be less than the melting temperature on part of the melt surface in which case the control function should be modified or rejected. In addition, the relative differences between the required melt zone interface gradients and those obtained using a candidate melt zone surface control function should be examined before accepting the control function as an approximate solution of Problem P1-3. Incidentally, these relative differences are approximated and displayed by the software developed for Problem P1-3.

6.0 FUTURE WORK AND UNRESOLVED ISSUES

Although the work presented in this report is, in itself, rather complete, several side issues remain unresolved and should be included in any continuation of this type of research. In this chapter, some of these issues are addressed.

6.1 VERIFICATION USING FREE BOUNDARY ALGORITHMS

The basic idea of the three methods described in Chapters 3 and 4 was to determine the properties a surface control function must have if a flat interface was to be maintained. Unfortunately, both methods involved many numerical approximations and some simplifying assumptions. As an example, for the method described in Chapter 4, the thermal distributions in the assumed infinitely long solid regions were approximated by numerical methods designed for finite length regions. In addition, the interface gradients were approximated by finite differences (a second source of error) followed by least squares Bessel function fits of these approximate interface gradients (a third possible source of error.) Thereafter, the surface control function was approximated by solving an overposed system of equations using only a finite number of terms in the control function (another source of error). Given these several possible sources of error, the actual interface shapes maintained using the computed surface control function should be constructed using some multiphase free boundary algorithm (for a survey, see [19]). The results of such numerical experiments should hopefully further verify the methods discussed in this report and should indicate some future areas to be studied with error reduction in mind.

6.2 MAINTAINING CURVED INTERFACES

Although thermal stresses, which can generate defects in the crystal, are generally minimal for a planar interface [20], a slightly curved interface shape is also quite desirable in some cases. Specifying the desired shapes, the required surface control functions could probably (with sufficient investigation) be constructed using methods similar to those in Chapters 3 and 4 after the introduction of transformations similar to those described in [12].

6.3 NON-DIRICHLET BOUNDARY CONDITIONS

Boundary conditions other than the Dirichlet type (Equations (FZ6), (FZ8), and (FZ10) of Figure 1-2) should be investigated. Fortunately, much of the work for this type of problem will probably be straightforward. For example, suppose a question like Problem P1-3 is to be solved where the Dirichlet boundary condition (see Equation (1.2.6))

$$T(x,1) = h(x), \quad 0 < x < Q$$

is replaced by a boundary condition of the type

$$T_r(x,1) = K\left[T^{\alpha}(x,1) - S^{\alpha}(x)\right] \tag{6.3.1}$$

ORIGINAL PAGE IS
OF POOR QUALITY

where $S(x)$ is the desired surface control function (for example, if $\alpha=4$, then $S(x)$ might be the temperature of a furnace wall providing radiant heating). To solve this problem, first solve Problem P1-3 as stated in Section 1.2. Using the computed Dirichlet type surface control function $h(x)$, next solve Problem P2-3 (let $x_0 = 0$ and $x_N = Q$). Then place the resulting temperature distribution $T(x,r)$ into the boundary condition (6.3.1) and solve for the desired surface control function $S(x)$.

6.4 BASIS FUNCTIONS USED TO EXPAND THE CONTROL FUNCTIONS

The exponentially decaying functions used to expand the solid regions' surface control functions ($f(x)$ and $g(x)$ of Equations (3.0.23) and 3.0.31) respectively) were selected because they represented what a typical control function would be intuitively expected to resemble and because they allowed for simple integrations in Equations (3.0.22) and (3.0.28). However, from a computational point of view, these basis functions are not the best†. For example, some preliminary experiments indicate that replacing Equations (3.0.23) and (3.0.31) by

$$f(x) \approx c_1 + \sum_{k=2}^{NSYS} c_k \, EXP\left(-(t+k-2)^2\right)$$

and

$$g(x) \approx c_1 + \sum_{k=2}^{NSYS} c_k \, EXP\left(-(t-k+2+Q)^2\right)$$

respectively can significantly reduce the least squares residuals of overposed systems like Equation (3.0.25). More study is needed to find basis functions that both further reduce the least squares residuals and are not too difficult to integrate in equations like (3.0.22) and (3.0.28).

6.5 APPLICATIONS OF LINEAR PROGRAMMING

The linear system of equations (3.0.25) will be underposed if MTERM+2 < NSYS. However, the desired coefficient vector $\bar{C}$ may still be determined as follows. Let $\bar{r}$ be the residual vector of Equation (3.0.25), i.e., $\bar{r} = \bar{b} - L\bar{C}$.

† It is sometimes dangerous to approximate a function $f(x)$ by a sum:

$$f(x) \approx \sum_{k=1}^{N} c_k f_k(x)$$

where all or most of the functions $f_k(x)$ "resemble" each other, e.g., $f_k(x) = Exp(kx)$. This, for example, is why Chebyshev polynomials are preferred over the so-called standard basis, $f_k(x) = x^k$, for polynomial approximation on certain domains.

ORIGINAL PAGE IS
OF POOR QUALITY

Then solve the linear programming problem [13, pg 15]

$$\left.\begin{array}{l} \min \sum_i |r_i| \\ \text{subject to} \quad L\bar{C} + \bar{r} = \bar{b} \end{array}\right\} \qquad (6.5.1)$$

In fact, such a technique might be used to reduce the chance of say, $f(x)$ of (3.0.23), becoming positive † (recall that the melting temperature was translated to zero) as x approaches negative infinity. To accomplish this, first select a grid,

$$x_1 < x_2 < \cdots < x_N < 0$$

partitioning a portion of the lower solid region, and then adjoin to (6.5.1) the additional N constraints (see Equation (3.0.23)):

$$\sum_{k=1}^{NSYS} c_k \, \mathrm{EXP}\,((k-1)x_i) < 0 \ , \ i=1, \cdots , N$$

Some preliminary numerical experiments suggest this idea has sufficient potential to warrant further investigation. Although this discussion has centered on the lower solid region, these ideas are applicable to either of the solid regions or to the melt zone.

6.6 MODELS AND REALITY

The FZ process was modeled in this effort as a steady state process on an infinitely long boule. Unfortunately for the modeler (but fortunately for the commercial FZ operator), the boule has finite length *. For finite length boules, the problem of finding the proper surface control functions to maintain flat interfaces would now involve end effects and various time transients.

† The partial differential equations used to establish the desired surface control functions are quite ignorant of the fact that surface control functions for solid regions should always be below the material melting point. In fact, in some numerical experiments where MTERM and MSYS were large, the computed surface control functions for one of the solid regions became greater than the melting temperature. This is only one of the dangers in trying to solve an overposed problem.

* Fortunately, the assumptions and results of this effort are still quite reasonable for long boules with slow growth rates.

However, the basic ideas discussed in this report could probably be extended to cover such difficulties. The resulting partial differential equations would involve the additional term

$$\frac{\partial}{\partial t} T(x,r,t)$$

(where t represents time) and hence would be parabolic instead of elliptic. The boundary conditions would also be time dependent. However, the dual integral transform pairs used in this report should still provide enough information concerning the surface control functions (required for flat interfaces) to allow for their construction.

In addition, the fluid dynamics of the melt zone should be incorporated in the computation of the surface control functions. Of the topics discussed in this chapter, this is undoubtedly the most difficult one to model and resolve.

BIBLIOGRAPHY

1. Boyce, W.E., and DiPrima, R.C., Elementary Differential Equations and Boundary Value Problems. New York: John Wiley and Sons (1977).

2. Brown, R., "Fluid Flow in Melt Crystal Growth Analysis of Floating Zone and Bridgman Systems," presented at the Float Zone Science Working Group Meeting, Marshal Space Flight Center, NASA, March 23, 1983.

3. Carnahan, B., Luther, H.A., and Wiles, J.O., Applied Numerical Methods. New York: John Wiley and Sons (1969).

4. Cothran, E., Private Communication (1983).

5. Daniel, J.W. and Moore, R.E., Computation and Theory in Ordinary Differential Equations. San Francisco: W.H. Freeman and Company (1970).

6. Flemings, M.C., Solidification Processing. New York: McGraw-Hill (1974).

7. Foster, L.M., "Surface Control Temperatures for the Bridgman-Stockbarger Technique," in Developments in Theoretical and Applied Mechanics, SECTAM XI Conference Proceedings, pril 1982, pp 603-612.

8. Foster, L.M., "Solid Region Surface Temperatures Required for Flat Interfaces in Float Zone Processes," presented at the Float Zone Science Working Group Meeting, Marshal Space Flight Center, NASA, March 1983.

9. Isaacson, E., and Keller, H.B., Analysis of Numerical Methods. New York: John Wiley and Sons (1966).

10. Kern, E.L., "Increased Uniformity of Silicon Needed for Critical Devices; How Can We Improve the Float Zone Process?," presented at the Materials Research Society Annual Meeting, Nov 16-19, 1981, Boston.

11. Kern, E.L., Private Communication (1983).

12. Kobayaski, N., "Power Required to Form a Floating Zone and the Zone Shape," Journal of Crystal Growth, Vol 43, pp 417-424 (1978).

13. Marty, K.G., Linear and Combinatorial Programming. New York: John Wiley and Sons (1976).

14. Naumann, R., "An Analytical Approach to Thermal Modeling of Bridgman-Type Crystal Growth II, Two Dimensional Analysis," Journal of Crystal Growth, Vol 58, pp 569-584 (1982).

15. Ozisik, M.N., Heat Conduction. New York: John Wiley and Sons (1980).

16. Rice, John R. (editor), Mathematical Software. New York: Academic Press (1971).

17. Stewart, G.W., Introduction to Matrix Computation, New York: Academic Press (1973).

18. Tolstov, G.P., Fourier Series. Englewood Cliffs, New Jersey: Prentice-Hall, Inc. (1962).

19. Wilson, D.G., Solomon, A.D., and Boggs, P.T., editors. Moving Boundary Problems. New York: Academic Press (1978).

20. Wilcox, W.R., and Chang, C.E., "Control of Interface is the Vertical Bridgman-Stockbarger Technique," Journal of Crystal Growth, Vol 21, pp 135-140 (1974).

APPENDIX A.0 THE OWNERS MANUAL
(or a user guide to developed software)

A.1 INTRODUCTION

The data input procedures and output interpretations for the software developed to approximate the solutions of Problems P1-1, P1-2, P1-3 and P2-1 is the subject of this appendix. To begin, Problem P2-1 is covered in Appendix A.2 and is followed by Problems P1-1 and P1-2 in Appendix A.3. To finish, Problem P1-3 is the subject of Appendix A.4.

A.2 USER CONSIDERATIONS FOR PROBLEM P2-1 SOFTWARE

The data input procedure and output interpretation for the software developed for Problem P2-1 is the subject of this section. To begin, all the required data are input in the form of punched cards. The definitions and formats of this input data are summarized in Table A-1.

TABLE A-1 PROBLEM P2-1 SOFTWARE INPUT

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS†
	READ(5,16)IHFC,M 16 FORMAT(2I10)	
IHFC	= 1 if a cubic spline will be used to approximate the boundary function h(x) in Condition (2.2.4). = 0 if the user will supply a functional form of h(x) (see Condition (2.2.4)). In this case, the user must insert this functional form of h(x) in the subroutine HFC (see the software list in Appendix C.2).	
M	Number of knots used to approximate h(x) (see Condition (2.2.4)) by a cubic spline (IHFC=1). If IHFC=0, set M=0.	
	DO 32 I = 1,M READ(5,22)XD(I),YD(I) 32 CONTINUE 22 FORMAT(4E20.10)	
XD(I)	The dimensionless axial position of the Ith knot used to approximate h(x) (see Condition (2.2.4)). Ignore if IHFC=1. XD(I)<XD(I+1).	x/R

† x, r and R will denote the axial distance, the radial position and the rod radius respectively. °T will denote whatever temperature scale the user prefers.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE A-1 PROBLEM P2-1 SOFTWARE INPUT (CONT)

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS
YD(I)	The surface temperature represented by the Ith knot used to approximate h(x) (see Condition (2.2.4)). Ignore if IHFC=0.	^{o}T
	READ(5,10)P,X0,XN,MSUM,NGRID,NR 10 FORMAT(3F10.5,4I10)	
P	Peclet number	dimen-sionless
X0	Dimensionless axial position of bottom boundary displayed in Figure 2-1. Remark: The boundary temperature distribution, A(r/R) at X0 (see Condition (2.2.2)) is user supplied in Subroutine AFC (see Appendix C.2).	x/R
XN	Dimensionless axial position of top boundary displayed in Figure 2-1. Remark: The boundary temperature distribution, B(r/R) at XN (see Condition (2.2.3)) is user supplied in Subroutine BFC (see Appendix C.2).	x/R
MSUM	The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate θ (x,r). MSUM must be less than 21.	
NGRID	(XN-X0)/NGRID is the grid size employed in System (2.2.20). In addition, the final temperature distribution is output for NGRID+1 axial values from X0 to XN. NGRID may not exceed 500.	
NR	The final temperature distribution is output for NR+1 radial values (r/R) from 0 to 1. NR may not exceed 100.	

An input sample is illustrated next in Figure A-1.

The output is labeled clearly for ease of use. The input data are first viewed followed by the thermal distribution (the approximate solution of Problem P2-3) given in table format (see Figure A-2). Incidentally, the axial and radial positions in Figure A-2 are given in dimensionless form (x/R and r/R). The thermal gradients at X0 and XN are given last in a table format (again, see Figure A-2).

```
          1               5                   -2.0
        -1.0                                  -0.375
        -0.5                                   0.0
         0.0                                  -0.125
         0.5                                   0.0
         1.0
0.00734     -1.0            1.00                            20          500          8
```

Figure A-1 Sample Input For Problem P2-1 Software

INPUT DATA

THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIMATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING 5 (X, TEMP) DATA POINTS

X	SURFACE TEMP.
-.1000000000E+01	-.2000000000E+01
-.5000000000E+00	-.3750000000E+00
.0000000000E+00	.0000000000E+00
.5000000000E+00	-.1250000000E+00
.1000000000E+01	.0000000000E+00

P	X0	XN	MSUM	NGRID	NR
.7340E-02	-.1000E+01	.1000E+01	20	500	8

ORIGINAL PAGE IS OF POOR QUALITY

TEMPERATURE DISTRIBUTION

		R= .00000000	R= .12500000	R= .25000000	R= .37500000	R= .50000000	R= .62500000
X=	1.000000	-.14506914E-13	-.14506914E-13	-.14506914E-13	-.14506914E-13	-.14506914E-13	-.14506914E-13
X=	.996000	-.11337865E-02	-.11033876E-02	-.10858318E-02	-.10921535E-02	-.11344015E-02	-.12325876E-02
X=	.992000	-.22597673E-02	-.22053124E-02	-.21730316E-02	-.21849399E-02	-.22672332E-02	-.24645230E-02
X=	.988000	-.33797012E-02	-.33061072E-02	-.32612799E-02	-.32781644E-02	-.33987348E-02	-.36957395E-02
X=	.984000	-.44949557E-02	-.44060317E-02	-.43503272E-02	-.43716695E-02	-.45290746E-02	-.49261430E-02

Figure A-2 Sample Output For Problem P2-1 Software

```
X= -.996000  *  -.19912480E+01  -.19912276E+01  -.19910154E+01  -.19905872E+01  -.19899154E+01  -.19889561E+01
X= -1.000000 *  -.20000000E+01  -.20000000E+01  -.20000000E+01  -.20000000E+01  -.20000000E+01  -.20000000E+01


               R= .75000000     R= .87500000     R=1.00000000
             ****************************************************
X=  1.000000 *  -.14506914E-13  -.14506914E-13  -.14506914E-13
X=   .996000 *  -.14726681E-02  -.20732674E-02  -.39680640E-02



X=  -.988000 *  -.19625840E+01  -.19555937E+01  -.19405743E+01
X=  -.992000 *  -.19750585E+01  -.19703917E+01  -.19602555E+01
X=  -.996000 *  -.19875312E+01  -.19851947E+01  -.19800639E+01
X= -1.000000 *  -.20000000E+01  -.20000000E+01  -.20000000E+01



                       T H E R M A L   G R A D I E N T S

         R      GRAD. AT X=  -1.00000   GRAD. AT X=  1.00000

      .000000      .219027753E+01         .288593389E+00
      .010000      .218896843E+01         .283641429E+00
      .020000      .218361427E+01         .281063989E+00
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-2 Sample Output For Problem P2-1 Software (Cont)

.030000	.217722980E+01	.277605782E+00
.040000	.217129103E+01	.2.4236855E+00
.050000	.216760219E+01	.271843131F+00
.060000	.216723893E+01	.i70952681E+00
.070000	.217024747E+01	.271588180E+00
.080000	.217569428E+01	.2 3291266E+00
.090000	.218203717E+01	.275304714E+00
.100000	.218767990E+01	.276844741E+00
.110000	.219151375E+01	.277366915E+00
.120000	.219326081E+01	.276734677E+00
.130000	.219350933E+01	.275236739E+00
.140000	.219344378E+01	.273454545E+00
.150000	.219437938E+01	.272033759E+00
.160000	.219727608E+01	.271445624E+00
.170000	.220240702E+01	.271824184E+00
.180000	.220929427E+01	.272934730E+00
.190000	.221692360E+01	.274279201E+00
.200000	.222414790E+01	.275294271E+00
.210000	.223012349E+01	.275565548E+00
.220000	.223461656E+01	.274978056E+00
.230000	.223807048E+01	.273749292E+00
.240000	.224141666E+01	.272336389E+00
.250000	.224570763E+01	.271255977E+00
.260000	.225171514E+01	.270886858E+00
.870000	.366838317E+01	.508041063E+00
.880000	.373531188E+01	.529119790E+00
.890000	.380443935E+01	.551148833E+00
.900000	.387553602E+01	.573034108E+00
.910000	.394152229E+01	.594245310E+00
.920000	.400932740E+01	.615238414F+00
.930000	.408019058E+01	.637605103E+00
.940000	.415922006E+01	.663858303E+00
.950000	.425225093E+01	.696873747E+00
.960000	.436427316E+01	.739120244E+00
.970000	.449785950E+01	.791889160F+00
.980000	.465205549E+01	.854749698E+00
.990000	.482208444E+01	.925403066E+00
1.000000	.500000000E+01	.100000000F+01

Figure A-2 Sample Output For Problem P2-1 Software (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

To finish, the user must supply an algorithm to fit (in the least squares sense) a linear combination of functions to a given set of data points (see the end of Section 2.2 for a short discussion). This algorithm is required in the subroutine COEFS. In addition, an algorithm to evaluate the J_0 Bessel function (required in the subroutine J0) must be provided. These required algorithms are generally available from the host computer library† or may be obtained from various software packages such as the IMSL and FUNPACK.

---

† The user is warned, however, that many host computer mathematics libraries (with the general exception of IBM) still contain numerous faux pas that were well known years ago and still remain uncorrected.

A.3 USER CONSIDERATIONS FOR PROBLEMS P1-1 and P1-2 SOFTWARE

The data input procedure and output interpretation for the software developed for Problems P1-1 and P1-2 are the subjects of this section. Recall that the general problem is to find the solid regions' surface control functions ($f(x)$ and $g(x)$ in Problems P1-1 and P1-2 respectively) which, for the sake of flat interfaces, are compatible with the a priori given melt zone surface temperature distribution ($h(x)$). At present, all the required data are input in the form of punched cards. The definitions and formats of the input data are summarized in Table A-2.

TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS†
	READ(5,80)IHFC,M 80 FORMAT(2I10)	
IHFC	= 1 if a cubic spline will be used to approximate the melt zone surface temperature distribution, h(x). = 0 if the user will supply a functional form of h(x). In this case, the user must insert this functional form of h(x) in the subroutine HFC (see the software list in Appendix C.3)	
M	Number of knots used to approximate h(x) by a cubic spline (IHFC=1). If IHFC=0, set M=0.	
	DO 32 I = 1,M READ(5,22)XD(I),YD(I) 32 CONTINUE 22 FORMAT (2E20.10)	
XD(I)	The axial position of the Ith knot used to approximate h(x). Ignore if IHFC=0. In addition, XD(I)<XD(I+1) and it is recommended that the axial position be measured from the lower melt-solid interface, i.e., XD(1)=0.0.	rad
YD(I)	The surface temperature represented by the Ith knot used to approximate h(x). Ignore if IHFC=0.	°K above melting temp.
	READ(5,10)P,XO,XN,MTERM,MSUM,NGRID,NR 10 FORMAT(3F10.5,4I10)	
P	Melt Zone Peclet Number	dimension-less
XO	Axial position of lower melt-solid interface; XO=0.0 is recommended.	rad
XN	Axial position of upper melt-solid interface	rad
MTERM	Set to Zero	
MSUM	The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate the melt zone temperature distribution, T(x,r). MSUM must be less than 21.	
NGRID	(XN-XO)/NGRID is the grid size employed in System (2.2.20). In addition, the melt zone temperature distribution is output for NGRID/10+1 axial values from XO to XN. NGRID may not exceed 500 and must be divisible by 10.	
NR	The melt zone temperature distribution is output for NR+1 radial values (rad) from 0 to 1. NR may not exceed 100.	

† All lengths will be measured in radius (rad) units, e.g., 2 rad is as long as the rod is wide. The temperature scale will be °K above or below the material melting point.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS
	READ(5,10)P,XO,XN,MTERM,MSUM,NGRID,NR 10 FORMAT(3F10.5,4I10)	
P	Lower solid region Peclet number	dimensionless
XO	The semi-infinite lower solid region is, for computational purposes, truncated to a finite length. XO is the axial position of lower end of this truncated region. Review the end of Section 4.3 for details.	rad
XN	Axial position of lower melt-solid interface.	rad
MTERM	Integer parameter determining the size of system used to compute the lower solid region's surface control function. See Equation (3.0.24).	
MSUM	The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate the lower solid temperature distribution, T(x,r). MSUM must be less than 21.	
NGRID	(XN-XO)/NGRID is the grid size employed in System (2.2.20). In addition, the lower solid temperature distribution is output for NGRID/10+1 axial values from XO to XN. NGRID may not exceed 500 and must be divisible by 10.	
NR	The lower solid temperature distribution is output for NR+1 radial values (rad) from 0 to 1. NR may not exceed 100.	
	READ(5,90)RKS,RKL,RL,NSYS 10 FORMAT(3E20.10,I10)	
RKS	Conductivity of material in lower solid region	$\frac{\text{cal}}{{}^{o}\text{K rad sec}}$
RKL	Conductivity of material in melt zone	$\frac{\text{cal}}{{}^{o}\text{K rad sec}}$
RL	$\mathcal{L}$ of Equation FZ4, Figure 1-2. RL is the product of the growth rate, the solid materials density and the latent heat.	$\frac{\text{cal}}{\text{sec rad}}$
NSYS	Number of terms in expansion of lower solid region's surface control function (see Equation (3.0.23)). NSYS may not exceed 20.	

TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS
	`READ(5,21)MINTERM,MAXTERM,DELTERM,MINNSYSM,MAXNSYS,DELNSYS` `21 FORMAT(8I10)`	
	The software is designed to compute the lower solid region's surface control function for many combinations of MTERM and NSYS. To do this, the user must supply the bounds and increments for the cases desired.	
MINTERM	Lower bound on MTERM.	
MAXTERM	Upper bound on MTERM.	
DELTERM	Integer increment for MTERM	
MINNSYS	Lower bound on NSYS	
MAXNSYS	Upper bound on NSYS	
DELNSYS	Integer increment for NSYS	
	`READ(5,50)IOPTION,CLIP` `50 FORMAT(I10,F10.5)`	
	After the lower solid region's surface control function, f(x), is determined, the user may specify certain modifications of the surface control function. These options are principally of use when the surface control function is above the material's melting point on portions of the lower solid region's surface. If f(x) is interpreted as the $^{\circ}$K above or below the melting point, then let A be the lower endpoint of the largest subinterval of the form [A,XN] on which f(x) is not positive. If A < X0, reassign A to be X0. Let (XMIN,FMIN) be the minimum point of f(x) on the interval [A,XN].	
IOPTION	= 0 if f(x) is not to be modified. In this case, CLIP may be assigned any value, e.g., zero. = 1 if f(x) is to be redefined as: $f(x) \longleftarrow \begin{cases} f(x) \text{ if } x > XMIN \\ FMIN \text{ otherwise} \end{cases}$ = 2 if f(x) is to be redefined as: $f(x) \longleftarrow \begin{cases} f(x) \text{ if } x > XMIN \\ \min\{f(x), CLIP\} \text{ otherwise} \end{cases}$	

TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

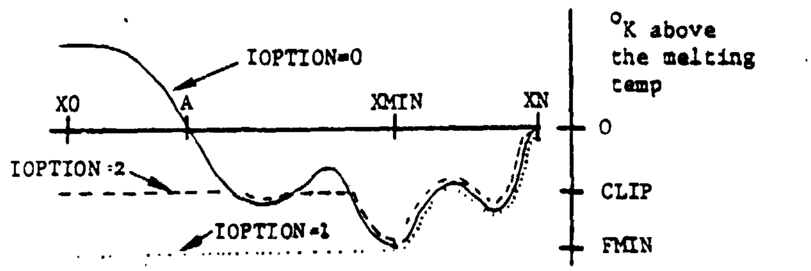
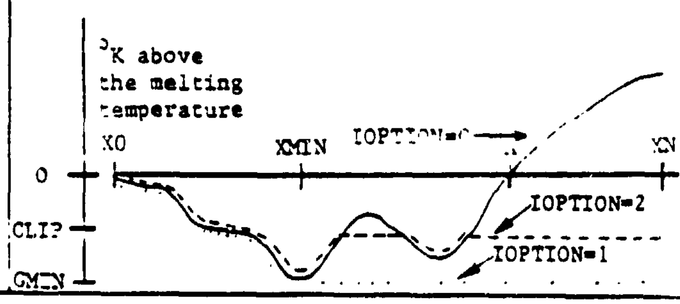
PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS[+]
	These options are graphically illustrated below. 	
	READ(5,10)P,XO,XN,ITERM,MTERM,MSUM,NGRID,NR 10 FORMAT(3F10.5,4I10)	
P	Peclet number for upper solid region	Dimension-less
XO	Axial position of upper interface	rad
XN	The semi-infinite upper solid region is, for computational purposes, truncated to a finite length. XN is the axial position of the upper end of this truncated region.	rad
MTERM	Number of equations (n=1,2,···,MTERM) of the type given in Equation (3.0.28) used in least squares generation of upper solid region's surface control function.	
MSUM	The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate the upper solid regions' temperature distribution, T(x,r). MSUM must be less than 21.	
NGRID	(XN-XO)/NGRID is the grid size employed in System (2.2.20). In addition, the upper solid region's temperature distribution is output for NGRID/10+1 axial values from XO to XN.NGRID may not exceed 500 and must be divisible by 10.	
NR	The upper solid temp. distribution is output for NR+1 radial values (rad) from 0 to 1. NR may not exceed 100.	
	READ(5,90)RKS,RKL,RL,NSYS 90 FORMAT(3E20.10,I10)	
RKS	Conductivity of material in upper solid region	$\frac{cal}{^{o}K\ rad\ sec}$
RKL	Conductivity of material in melt zone	$\frac{cal}{^{o}K\ rad\ sec}$
RL	ℓ of Equation FZ2, Figure 1-2. RL is the product of the growth rate, the solid material's density and the latent heat	$\frac{cal}{sec\ rad^2}$

TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS
NSYS	Number of terms in exponential expansion of the upper solid region's surface control function (see Equation (3.0.31))	
	READ(6,25)MINTERM,MAXTERM,DELTERM,MINNSYS,MAXNSYS,DELNSYS 21 FORMAT(8I10)	
MINTERM MAXTERM DELTERM MINNSYS MAXNSYS DELNSYS	As previously defined above but applied to the upper solid region.	
	READ(5,50)IOPTION,CLIP 50 FORMAT(I10,F10.5)	
IOPTION	After the upper solid region's surface control function, g(x), is determined the user may specify certain modifications of this surface control function. The options are principally of use when the surface control function is above the material's melting point on portions of the upper solid region's surface. The definitions of IOPTION and CLIP are similar to their previous definitions above and are illustrated below. 	

ORIGINAL PAGE IS
OF POOR QUALITY

An input sample is illustrated in Figure A-3.

The output is clearly labeled for ease of use. The melt zone input data is first displayed followed by the melt zone temperature distribution (given in table format) and interface gradients (see Figure A-4). The lower solid region's material and system parameters and the values of MINTERM,···,DELNSYS are next displayed followed by the required lower solid region interface gradient, B(r) of Equation (1.2.1) (again see Figure A-4).

For each of various acceptable combinations of MTERM and NSYS (recall the definitions of MINTERM,···,DELNSYS), a lower solid region surface control function is computed. For each of these cases, the values of MTERM and NSYS are first displayed followed by the coefficients (see Equation (3.0.23)) used to determine the surface control function. Using the surface control function, the temperature distribution in the lower solid region is next displayed (in table form†). The lower solid region's interface gradient is then given followed last by the relative difference (in the L^2 norm) between the required lower solid region interface gradient and the interface gradient obtained by use of the surface control function (see Figure A-5).

After all the lower solid region cases (various combinations of MTERM and NSYS) are given, the results for the upper solid region are displayed in a similar fashion (see Figure A-6).

To finish, the user must supply the two algorithms described at the end of Appendix A.2. In addition, an algorithm to solve (in a least squares sense) an overposed system of linear equations must also be provided for use in the subroutines SOLID2 and SOLID3 (see Appendix C.3).

† The surface control function values can be read from this table in the R=1 column.

ORIGINAL PAGE IS
OF POOR QUALITY

```
   1
     24
  0.0                          0.0
  0.2                          1.8
  0.4                          3.8
  0.6                          5.6
  0.8                          7.0
  1.0                          8.4
  1.2                          9.3
  1.4                         10.0
  1.6                          9.8
  1.8                          9.5
  2.1                          9.0
  2.3                          8.4
  2.5                          8.0
  2.7                          7.4
  2.9                          6.9
  3.1                          6.2
  3.3                          5.7
  3.5                          5.0
  3.7                          4.3
  4.0                          3.4
  4.2                          2.7
  4.4                          1.9
  4.6                          1.04027
  4.842                        0.0
  0.00421     0.0             4.842             10          20          500          1
  0.00734    -10.0            0.0               10          20          500          1
  0.0180975                   0.038608            0.23841                            5
              4             4             2      3           3           3
              0   -0.5
 0.00734     4.842          14.842              10          20          500          1
 0.0180975                  0.038608            0.23841                              5
             4             4             1       3           3           1
             0   0.0
```

Figure A-3 Sample Input For Problems P1-1 And P1-2 Software

MELT ZONE

INPUT DATA

THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIMATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING 24 (X, TEMP) DATA POINTS

X	SURFACE TEMP.
.0000000000E+00	.0000000000E+00
.2000000000E+00	.1800000000E+01
.4000000000E+00	.3800000000E+01
.6000000000E+00	.5600000000E+01
.[illegible]00000000E+00	.7000000000E+01
.10[illegible]00000E+01	.8400000000E+01
.[illegible]90000E+01	.9300000000E+01
.[illegible]000000E+01	.1000000000E+02
.1600000000E+01	.9800000000E+01
.1800000000E+01	.9500000000E+01
.2100000000E+01	.9000000000E+01
.2300000000E+01	.8400000000E+01
.2500000000E+01	.8000000000E+01
.2700000000E+01	.7400000000E+01
.2900000000E+01	.6900000000E+01
.3100000000E+01	.6200000000E+01
.3300000000E+01	.5700000000E+01
.3500000000E+01	.5000000000E+01
.3700000000E+01	.4300000000E+01
.4000[illegible]000E+01	.3400000000E+01
.4200[illegible]000E+01	.2700000000E+01
.4400000000E+01	.1900000000E+01
.4600000000E+01	.1040270000E+01
.4842000000E+01	.0000000000E+00

P	XO	XN	ITERM	MSUM	NGRID	NR
.4210000000E-02	.0000000000E+00	.4842000000E+01	10	20	500	1

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-4 Sample Melt Zone Output For Problems P1-1 And P1-2 Software

```
                         MELT   ZONE
                 TEMPERATURE   DISTRIBUTION


          R= .00000000      R=1.00000000

                  **********************************
X=  4.842000   *   .00000000E+00     .00000000E+00
X=  4.745160   *   .34009614E+00     .81022303E+00
```

```
X=   .096840   *   .75664366E+00     .82216196E+00
X=   .000000   *   .00000000E+00     .00000000E+00
```

MELT ZONE
THERMAL GRADIENTS

R	GRAD. AT X= .00000	GRAD. AT X= 4.84200
.000000	.779307696E+01	-.392454267E+01
.100000	.783548244E+01	-.393102197E+01
.200000	.788833393E+01	-.394195240E+01
.300000	.797372134E+01	-.396019296E+01
.400000	.809126[illegible]01E+01	-.398620179E+01
.500000	.823836198E+01	-.402037199E+01
.600000	.840832765E+01	-.406279632E+01
.700000	.858482066E+01	-.411258421E+01
.800000	.872634777E+01	-.416600690E+01
.900000	.871140417E+01	-.421000913E+01
1.000000	.785486434E+01	-.416511214E+01

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-4 Sample Melt Zone Output For Problems P1-1 And P1-2 Software (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

LOWER SOLID

INPUT DATA

P	XO	XN	ITERM	MSUM	NGRID	NR
.7340000000E-02	-.1000000000E+02	.0000000000E+00	10	20	500	1

RKS	RKL	RL
.1809750000E-01	.3860800000E-01	.2384100000E+00

MAXTERM	MINTERM	DELTERM	MAXNSYS	MINNSYS	DELNSYS
4	4	2	3	3	3

LOWER SOLID THERMAL GRADIENTS

R	GRAD
.1000000000E+01	.3583401152E+01
.9900000000E+00	.4000064697E+01
.9800000000E+00	.4386697918E+01

R	GRAD
.9000000000E-01	.3547174852E+01
.8000000000E-01	.3556482835E+01
.7000000000E-01	.3564160391E+01
.6000000000E-01	.3564588019E+01
.5000000000E-01	.3554363047E+01
.4000000000E-01	.3533650885E+01
.3000000000E-01	.3506365751E+01
.2000000000E-01	.3479076492E+01
.1000000000E-01	.3458976406E+01
.0000000000E+00	.3451588088E+01

Figure A-4 Sample Melt Zone Output For Problems P1-1 And P1-2 Software (Cont)

FOR ITERM = 4 AND NSYS = 3

L O W E R   S O L I D   S U R F A C E   C O N T R O L   C O E F F I C I E N T S

K	C(K)
1	-.9773956816E+01
2	.1596373069E+02
3	-.6189773870E+01

L O W E R   S O L I D
T E M P E R A T U R E   D I S T R I B U T I O N

R= .00000000   R=1.00000000

X= .000000	-.19895197E-12	-.19895197E-12
X= -.200000	-.90526753E+00	-.85308907E+00
X= -.400000	-.18021796E+01	-.18543928E+01
X= -.600000	-.26799748E+01	-.28771997E+01
X= -.800000	-.35250253E+01	-.38506840E+01
X= -1.000000	-.43230198E+01	-.47389233E+01
X= -1.200000	-.50618501E+01	-.55272971E+01
X= -1.400000	-.57333396E+01	-.62137498E+01
X= -1.600000	-.63336247E+01	-.68032440E+01
X= -1.800000	-.68626430E+01	-.73042975E+01
X= -2.000000	-.73232281E+01	.2687056E+01
X= -2.200000	-.77201536E+01	.0811190E+01
X= -2.400000	-.80593022E+01	.83767001E+01
X= -2.600000	-.83470189E+01	-.86224197E+01
X= -2.800000	-.85896518E+01	-.88260903E+01
X= -3.000000	-.87932502E+01	-.89945124E+01
X= -3.200000	-.89633875E+01	-.91335246E+01
X= -3.400000	-.91050727E+01	-.92480889E+01

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-5 Sample Lower Solid Zone Output For Problems P1-1 And P1-2 Software (Cont)

```
X= -3.600000   *   -.92227250E+01   -.93423895E+01
X= -3.800000   *   -.93201870E+01   -.94199335E+01
X= -4.000000   *   -.94007627E+01   -.94836473E+01
```

```
X= -9.600000   *   -.97726364E+01   -.97728756E+01
X= -9.800000   *   -.97729439E+01   -.97730716E+01
X=-10.000000   *   -.97732321E+01   -.97732321E+01
```

LOWER SOLID
THERMAL GRADIENTS

R	GRAD. AT X= -10.00000	GRAD. AT X= .00000
.000000	.142264443E-02	.451780214E+01
.100000	.141793337E-02	.453329105E+01
.200000	.140024616E-02	.453975359E+01
.300000	.137056667E-02	.454828248E+01
.400000	.132875189E-02	.455740613E+01
.500000	.127447161E-02	.456383689E+01
.600000	.120709081E-02	.456141912E+01
.700000	.112537476E-02	.453820483E+01
.800000	.102673132E-02	.446841891E+01
.900000	.904582559E-03	.428282606E+01
1.000000	.724726722E-03	.358418838E+01

RELATIVE DIFFERENCES BETWEEN REQUIRED
AND OBTAINED GRADIENTS

L-2 ERROR
.17177

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-5 Sample Lower Solid Zone Output For Problems P1-1 And P1-2 Software (Cont)

```
                         U P P E R   S O L I D

                            I N P U T   D A T A

       P                 X0                 XN          ITERM   MSUM   NGRID   NR
.7340000000E-02   .4842000000E+01   .1484200000E+02      10      20     500     1

      RKS               RKL                RL
.1809750000E-01   .3860800000E-01   .2384100000E+00

   MAXTERM  MINTERM  DELTERM  MAXNSYS  MINNSYS  DELNSYS
      4        4        1        3        3        1


            U P P E R   S O L I D   T H E R M A L   G R A D I E N T S

                             R                      GRAD
                      .1000000000E+01       -.2205921534E+02
                      .9900000000E+00       -.2209549430E+02

                      .1000000000E-01       -.2154688899E+02
                      .0000000000E+00       -.2154600046E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-6 Sample Upper Solid Zone Output For Problems P1-1 And P1-2 Software

ORIGINAL PAGE IS
OF POOR QUALITY

UPPER SOLID SURFACE CONTROL COEFFICIENTS

K	C(K)
1	-.4159986163E+02
2	.6114089295E+02
3	-.1954103132E+02

UPPER SOLID
TEMPERATURE DISTRIBUTION

	R= .00000000	R=1.00000000
X= 14.842000 *	-.41597086E+02	-.41597086E+02
X= 14.642000 *	-.41595972E+02	-.41596471E+02
X= 14.442000 *	-.41594787E+02	-.41595721E+02
X= 14.242000 *	-.41593456E+02	-.41594804E+02
X= 5.642000 *	-.16039604E+02	-.18072754E+02
X= 5.442000 *	-.12293675E+02	-.13930674E+02
X= 5.242000 *	-.83230783E+01	-.93962468E+01
X= 5.042000 *	-.41987147E+01	-.46406773E+01
X= 4.842000 *	.00000000E+00	.00000000E+00

Figure A-6 Sample Upper Solid Zone Output For Problems P1-1 And P1-2 Software (Cont)

UPPER SOLID
THERMAL GRADIENTS

R	GRAD. AT X= 4.84200	GRAD. AT X= 14.84200
.000000	-.210128544E+02	-.550416054E-02
.100000	-.210632513E+02	-.548567893E-02
.200000	-.211411428E+02	-.541636330E-02
.300000	-.212686965E+02	-.530007010E-02
.400000	-.214461601E+02	-.513626343E-02
.500000	-.216719499E+02	-.492368754E-02
.600000	-.219411191E+02	-.465991006E-02
.700000	-.222406162E+02	-.434017523E-02
.800000	-.225354839E+02	-.395445971E-02
.900000	-.227154628E+02	-.347725757E-02
1.000000	-.220588471E+02	-.277571156E-02

RELATIVE DIFFERENCES BETWEEN REQUIRED
AND OBTAINED GRADIENTS

L-2 ERROR
.01752

FOR IITERM = 4 AND NSYS = 3

Figure A-6 Sample Upper Solid Zone Output For Problems P1-1 And P1-2 Software (Cont)

A.4 USER CONSIDERATIONS FOR PROBLEM P1-3 SOFTWARE

The data input procedure and output interpretation for the software developed for Problem P1-3 are the subjects of this section. Recall that the problem is to find a melt zone surface control function ($h(x)$ in Problem P1-3) which, for the sake of flat interfaces, is compatible with the a priori given surface temperature distributions of the two solid regions. At present, all the required data are input in the form of punched cards. The definitions and formats of the input data are summarized in Table A-3.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE A-3 PROBLEM P1-3 SOFTWARE INPUT

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS[†]
	READ(5,12)P,MSUM,NGRID,NR 12 FORMAT(E20.10,6I10)	
P	Peclet number of upper solid region	Dimensionless
MSUM	The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate the upper solid region's temperature distribution, T(x,r). MSUM must be less than 21.	
NGRID	For computational purposes, the semi-infinite upper solid region is truncated to a finite length (SLENGTH). SLENGTH/NGRID is the grid size employed in System (2.2.20) which is used to approximate the temperature distribution in this truncated upper solid region. In addition, the upper solid region's temperature distribution is displayed for NGRID/10+1 uniformly spaced axial values. NGRID may not exceed 500 and must be divisible by 10.	
NR	The upper solid region's temperature distribution is output for NR+1 radial values (rad) from 0 to 1. NR may not exceed 100.	
	READ(5,22)RKS,RKL,RL,SLENGTH 22 FORMAT(4E20.10)	
RKS	Conductivity of material in upper solid region	$\frac{cal}{^oK\ rad\ sec}$
RKL	Conductivity of material in melt zone	$\frac{cal}{^oK\ rad\ sec}$
RL	$\mathcal{L}$ of Equation FZ2, Figure 1-2. RL is the product of the growth rate, the solid material's density and the latent heat.	$\frac{cal}{sec\ rad^2}$
SLENGTH	For computational purposes, the semi-infinite upper solid region is truncated to a finite length, SLENGTH. For details, review the end of Section 2.3.	rad
	READ(5,22)Q 22 FORMAT(4E20.10)	
Q	Q is the length of the melt zone	rad
	READ(5,16)IHFC,M 16 FORMAT(10I5)	

† All lengths will be measured in radius (rad) units, e.g., 2 rad is as long as the rod is wide. The temperature scale will be ^oK above or below the material melting point.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE A-3 PROBLEM P1-3 SOFTWARE INPUT (CONT)

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS
IHFC	= 1 if a cubic spline will be used to approximate the surface temperature distribution, g(x), of the upper solid region. = 0 if the user will supply a functional form of g(x). In this case, the user must insert this functional form of g(x) in the subroutine HFC (see the software list in Appendix C.4).	
M	Number of knots used to approximate g(x) by a cubic spline (IHFC=1). If IHFC=0, set M=0.	
	DO 32 I = 1,M READ(5,22)XD(I),YD(I) 32 CONTINUE 22 FORMAT (4E20.10)	
XD(I)	The axial position of the Ith knot used to approximate g(x). Ignore if IHFC=0. In addition, XD(I) < XD(I+1), XD(1)=Q and XD(M)=Q+SLENGTH	rad
YD(I)	The surface temperature represented by the Ith knot used to approximate g(x). Ignore if IHFC=0.	^{o}K below melting temp
	READ(5,12)P,MSUM,NGRID,NR 12 FORMAT(E20.10,6I10)	
P MSUM NGRID NR	Same definitions as above but applied to the lower solid region	
	READ(5,22)RKS,RKL,RL,SLENGTH 22 FORMAT(4E20.10)	
RKS	Conductivity of material in lower solid region	$\frac{\text{cal}}{^{o}\text{K rad sec}}$
RKL	Conductivity of material in melt zone	$\frac{\text{cal}}{^{o}\text{K rad sec}}$
RL	$\mathcal{L}$ of Equation FZ4, Figure 1-2. RL is the product of the growth rate, solid material's density, and latent heat	$\frac{\text{cal}}{\text{sec rad}^2}$
SLENGTH	For computational purposes, the semi-infinite lower solid region is truncated to a finite length, SLENGTH. For details, review the end of Section 2.3.	rad

TABLE A-3 PROBLEM P1-3 SOFTWARE INPUT (CONT)

PROGRAM SYMBOL	VARIABLE DEFINITION	UNITS
	READ(5,16)IHFC,M 16 FORMAT(10I5)	
IHFC	Same definitions as previously given but applied to the surface temperature distribution, f(x), of the lower solid region.	
	DO 32 I = 1,M READ(5,22)XD(I),YD(I) 32 CONTINUE 22 FORMAT(4E20.10)	
XD(I)	The axial position of the I^{th} knot used to approximate f(x). Ignore if IHFC=0. In addition, XD(I)<XD(I+1), XD(1) =-SLENGTH, and XD(M)=0.0.	rad
YD(I)	The surface temperature represented by the I^{th} knot used to approximate f(x). Ignore if IHFC=0.	°K below melting temp.
	READ(5,12)P,MSUM,NGRID,NR 12 FORMAT(E20.10,6I10)	
P MSUM NGRID NR	Same definitions as previously given but applied to the melt zone of length Q	
	READ(5,16)MAXTERM,MINTERM,MAXNSYS,MINNSYS,DELTERM,DELNSYS 16 FORMAT(10I5)	
	The software is designed to compute the melt zone surface control function for various combinations of MTERM and NSYS. To do this, the user must apply the bounds and increments for the cases desired.	
MAXTERM	Upper bound on MTERM	
MINTERM	Lower bound on MTERM	
MAXNSYS	Upper bound on NSYS	
MINNSYS	Lower bound on NSYS	
DELNSYS	Integer increment for NSYS	

An input sample is illustrated in Figure A-7.

The output is clearly labeled for ease of use. The input data for the upper solid region is output first followed by displays of the upper solid region's temperature distribution (given in table format) and interface gradient (see Figure A-8). The lower solid region follows in a similar fashion (Figure A-9). Using Equations FZ2 and FZ4 of Figure 1-2, the required melt zone interface gradients are computed and then displayed along with the melt zone input data (see Figure A-10). For each of various combinations of MTERM and NSYS (recall the definition of MINTERM, $\cdots$, DELNSYS), the expansion coefficients of the melt zone surface control function (see Equation (4.0.18)) are output. Using the computed surface control function, the melt zone temperature distribution (given in table† form) and interface gradients are displayed next followed last by the relative difference (in the L^2 norm) between the required melt zone interface gradients and those obtained by use of the surface control function (see Figure A-11).

To finish, the user must supply the two algorithms described at the end of Appendix A.2. Also, an algorithm to solve (in a least squares sense) an overposed system of linear equations (for example, see [17, Chapter 5]) must be provided for use in the subroutine MELT1 (see Appendix C.4 for code listing). In addition, a numerical integration routine is required for use in the subroutines INTEGL1 and INTEGL2 (during software verification, the numerical quadrature code CADRE was used and is available in the IMSL package or from the open literature [16, Chapter 7]).

---

† The melt zone surface control function can be read from this table in the R=1 column.

```
        0.01685              20      500          2
        0.05175              0.1104               1.55981316        10.0
        2.072


     1   23
         2.072                0.0
         2.172              -20.0
         2.272              -40.0
         2.372              -60.0
         2.472              -80.0
         2.572             -100.0
         2.747             -150.0
         2.922             -200.0
         3.122             -250.0
         3.347             -300.0
         3.597             -350.0
         3.822             -400.0
         4.072             -450.0
         4.297             -500.0
         4.547             -550.0
         4.822             -600.0
         5.122             -650.0
         5.447             -700.0
         5.822             -750.0
         6.622             -800.0
         7.322             -825.0
        10.072             -850.0
        12.072             -860.0
         0.01685              20      500          2
         0.05175              0.1104               1.55981316        10.0
     1   32
       -10.0               -1140.0
        -8.0               -1138.0
        -6.0               -1135.0
        -5.0               -1130.0
        -4.0               -1120.0
        -3.15              -1100.0
        -2.6               -10[illegible]
        -2.3               -10[illegible].0
        -2.05               -950.0
        -1.9                -900.0
        -1.8                -850.0
        -1.75               -800.0
        -1.7                -750.0
        -1.65               -700.0
        -1.6                -650.0
        -1.50               -600.0
        -1.45               -550.0
        -1.375              -500.0
        -1.3                -450.0
        -1.2                -400.0
        -1.1                -350.0
        -1.0                -300.0
         -.90               -250.0
         -.80               -200.0
 -.7                    -175.
 -.6                    -150.
 -.5                   -125.
 -.4                        -100.
 -.3                   -75.
   -.2                 -50.
 -.1                  -25.
         0.0                  0.0
         0.009638             20      500          2
    20   18     15     12     2     3
```


Figure A-7 Sample Input For Problem P1-3 Software

U P P E R   S O L I D

I N P U T   D A T A

P	NGRID	NP	MSUM
.0168500000	500	2	20

RKS	RKL	RL	SLENGTH	MELT LENGTH
.5175000000E-01	.1104000000E+00	.1559513160E+01	.1000000000E+02	.2072000000E+01

THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIMATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING 23 (X, TEMP) DATA POINTS

X	SURFACE TEMP.
.2072000000E+01	.0000000000E+00
.2172000000E+01	-.2000000000E+02
.2272000000E+01	-.4000000000E+02
.2372000000E+01	-.6000000000E+02
.2472000000E+01	-.8000000000E+02
.25[illegible]2000000E+01	-.1000000000E+03
.2747000000E+01	-.1500000000E+03
.2922000000E+01	-.2000000000E+03
.3122000000E+01	-.2500000000E+03
.3347000000E+01	-.3000000000E+03
.3597000000E+01	-.3500000000E+03
.3822000000E+01	-.4000000000E+03
.4072000000E+01	-.4500000000E+03
.4297000000E+01	-.5000000000E+03
.4547000000E+01	-.5500000000E+03
.4822000000E+01	-.6000000000E+03
.5172000000E+01	-.6500000000E+03
.5447000000E+01	-.7000000000E+03
.5822000000E+01	-.7500000000E+03
.6622000000E+01	-.8000000000E+03
.7322000000E+01	-.8250000000E+03
.1007200000E+02	-.8500000000E+03
.1207200000E+02	-.8600000000E+03

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-8 Sample Upper S[illegible]id Region Output For Problem P1-3 Software

```
                              U P P E R   S O L I D
                     T E M P E R A T U R E   D I S T R I B U T I O N


         R= .00000000      R= .50000000      R= 1.00000000

             ***********************************************
X= 12.072000  *  -.86000000E+03  -.86000000E+03  -.86000000E+03
X= 11.872000  *  -.85808630E+03  -.85789401E+03  -.85696743E+03
X= 11.672000  *  -.85629005E+03  -.85595233E+03  -.85456960E+03

X=  2.272000  *  -.44625524E+02  -.43844929E+02  -.40000000E+02
X=  2.072000  *   .00000000E+00   .00000000E+00   .00000000E+00


                              U P P E R   S O L I D
                          T H E R M A L   G R A D I E N T S


         R        GRAD. AT X=   2.07200    GRAD. AT X=  12.07200

       .000000    -.222371981E+03          -.971529027E+01
       .100000    -.222254549E+03          -.972389500E+01
       .200000    -.221827668E+03          -.983403289E+01
       .300000    -.221059383E+03          -.100259137E+02
       .400000    -.219867163E+03          -.103083528E+02
       .500000    -.218138215E+03          -.106968433E+02
       .600000    -.215745790E+03          -.112164683E+02
       .700000    -.212591393E+03          -.119091341E+02
       .800000    -.208686203E+03          -.128520574E+02
       .900000    -.204250563E+03          -.142215304E+02
      1.000000    -.199372960E+03          -.168641380E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-8 Sample Upper Solid Region Output For Problem P1-3 Software (Cont)

```
                                   L O W E R   S O L I D

                                      I N P U T   D A T A
      P               NGRID      NR      MSUM
 .0158500000           500        2       20

     RKS               RKL              RL                  SLENGTH          MELT LENGTH
.5175000000E-01   .5175000000E-01   .1559513160E+01    .1000000000E+02   .2072000000E+01


THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIMATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING 32 (X, TEMP) DATA POINTS

       X              SURFACE TEMP.
 -.1000000000E+02   -.1140000000E+04
 -.8000000000E+01   -.1138000000E+04
 -.6000000000E+01   -.1135000000E+04
 -.5000000000E+01   -.1130000000E+04
 -.4000000000E+01   -.1120000000E+04
 -.3150000000E+01   -.1100000000E+04
 -.2600000000E+01   -.1050000000E+04
 -.2300000000E+01   -.1000000000E+04
 -.2050000000E+01   -.9500000000E+03
 -.1900000000E+01   -.9000000000E+03
 -.1800000000E+01   -.8500000000E+03
 -.1750000000E+01   -.8000000000E+03
 -.1700000000E+01   -.7500000000E+03

 -.5000000000E+00   -.1250000000E+03
 -.4000000000E+00   -.1000000000E+03
 -.3000000000E+00   -.7500000000E+02
 -.2000000000E+00   -.5000000000E+02
 -.1000000000E+00   -.2500000000E+02
  .0000000000E+00    .0000000000E+00
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-9 Sample Lower Solid Region Output For Problem P1-3 Software

```
                              LOWER   SOLID
                        TEMPERATURE   DISTRIBUTION


              R= .00000000     R= .50000000     R=1.00000000
             ****************************************************
X=   .000000  *  -.14213675E-10  -.14213675E-10  -.14213675E-10
X=  -.200000  *  -.63482945E+02  -.59698975E+02  -.50000000E+02
X=  -.400000  *  -.12917169E+03  -.12153109E+03  -.10000000E+03

X= -9.200000  *  -.11389864E+04  -.11389501E+04  -.11388211E+04
X= -9.400000  *  -.11392140E+04  -.11391779E+04  -.11390412E+04
X= -9.600000  *  -.11394621E+04  -.11394322E+04  -.11393058E+04
X= -9.800000  *  -.11397266E+04  -.11397094E+04  -.11396228E+04
X=-10.000000  *  -.11400000E+04  -.11400000E+04  -.11400000E+04

                          LOWER   SOLID
                        THERMAL   GRADIENTS


         R      GRAD. AT X= -10.00000   GRAD. AT X=    .00000

      .000000      .137679848E+01           .315917773F+03
      .100000      .137724220E+01           .315162847F+03
      .200000      .138700340E+01           .312884927F+03
      .300000      .140409795E+01           .309096024E+03
      .400000      .142940732E+01           .303821849F+03
      .500000      .146346655E+01           .297125500E+03
      .600000      .151177541E+01           .289133960E+03
      .700000      .157551231E+01           .280066463E+03
      .800000      .166338707E+01           .270252853F+03
      .900000      .179302660E+01           .260118980E+03
     1.000000      .204920656E+01           .250049519E+03
```

Figure A-9 Sample Lower Solid Region Output For Problem P1-3 Software (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS OF POOR QUALITY

MELT ZONE THERMAL GRADIENTS

AT INTERFACE

R	GRAD. AT X= 0.0	GRAD. AT X= Q
.000000	.137679848E+01	.315917773F+03
.010000	.137646440E+01	.315910814F+03
.020000	.137556996E+01	.315889748E+03
.980000	.197251855E+01	.252056694E+03
.990000	.201007205E+01	.251050231E+03
1.000000	.204920656E+01	.250049519E+03

MELT ZONE

INPUT DATA

P	NGRID	NP	NSUM
.0096380000	500	2	20

MAXTERM	MINTERM	DELTERM	MAXNSYS	MINNSYS	DELNSYS
20	18	2	15	12	3

Figure A-10 Sample Output Of Required Melt Zone Interface Gradients For Problem P1-3 Software

```
MELT  ZONE  SURFACE  CONTROL  COEFFICIENTS

              FOR IITERM = 18 AND NSYS = 12

              K              C(K)
              1          .2480592493E+03
              2         -.9228902871E+03
              3         -.1755568850E+04
              4          .7261786888E+04
              5          .5399929955E+04
              6         -.2110858581E+05
              7         -.7872762606E+04
              8          .2885342107E+05
              9          .5394050399E+04
             10         -.1882092097E+05
             11         -.1410088937E+04
             12          .4734664904E+04

                       MELT  ZONE
              TEMPERATURE  DISTRIBUTION

     R= .00000000     R= .50000000     R=1.00000000
   ****************************************************
X=  2.072000  *  -.14060862E+00  -.14060862E+00  -.14060862E+00
X=  2.030560  *   .38684312E+01   .36125410E+01   .35606315E+01
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-11 Sample Output Of Melt Zone Surface Control Function, Temperature Distribution, And Interface Gradients For Problem P1-3 Software

X=	1.989120	*	.78426276E+01	.73462466E+01	.64726203E+01
X=	1.947680	*	.11800446E+02	.11076742E+02	.87502738E+01
X=	1.906240	*	.15786131E+02	.14820840E+02	.11757661E+02
X=	.041440	*	.65227593E+01	.66688843E+01	.47322123E+01
X=	.000000	*	.37696892E+00	.37696892E+00	.37696892E+00

M E L T   Z O N E
T H E R M A L   G R A D I E N T S

R	GRAD. AT X= .00000	GRAD. AT X= 2.07200
.000000	.151746660E+03	-.954084126E+02
.100000	.149702239E+03	-.961318347E+02
.200000	.150510050E+03	-.954860236E+02
.300000	.151649883E+03	-.943028950E+02
.400000	.152395169E+03	-.926328125E+02
.500000	.151829668E+03	-.906170883E+02
.600000	.149040287E+03	-.885583762E+02
.700000	.143830096E+03	-.868133096E+02
.800000	.138088331E+03	-.852163587E+02
.900000	.134798834E+03	-.825575721E+02
1.000000	.131384040E+03	-.793498849E+02

RELATIVE DIFFERENCES BETWEEN REQUIRED
AND OBTAINED GRADIENTS

	L-2 ERROR
AT 0	.04904
AT Q	.04659

ORIGINAL PAGE IS
OF POOR QUALITY

Figure A-11 Sample Output Of Melt Zone Surface Control Function, Temperature Distribution, And Interface Gradients For Problem P1-3 Software (Cont)

APPENDIX B. CONVERGENCE OF EQUATION (2.3.11)

Recall, from Section 2.3, Equation (2.3.11)

$$\left.\begin{aligned}\bar{\theta}_n(x) = &\left[A_n + \frac{1}{S_n}\int_0^x \bar{G}_n e^{-\alpha_n t}dt\right] e^{\alpha_n x} \\ &+ \left[B_n - \frac{1}{S_n}\int_0^x \bar{G}_n e^{-\beta_n t}dt\right] e^{\beta_n x}\end{aligned}\right\} \quad (2.3.11)$$

where

$$S_n = \sqrt{P^2 + 4\lambda_n^2}$$

$$\alpha_n = (P + S_n)/2$$

$$\beta_n = (P - S_n)/2$$

$$B_n = \frac{-1}{S_n}\int_{-\infty}^0 \bar{G}_n e^{-\beta_n t}dt$$

and

$$A_n = -B_n + \mathcal{A}_n \frac{J_1^2(\lambda_n)}{2}$$

In this appendix, under the assumptions of Section 2.3, it will be shown that

$$\lim_{x\to-\infty} \bar{\theta}_n(x) = 0$$

For convenience, suspend the use of the "n" subscript and define

$$\|\bar{G}\|_{(a,b)} = \max_{a<x<b}|\bar{G}(x)|$$

Since $\bar{G}$ approaches zero as x proceeds to negative infinity, if given some $\varepsilon>0$, then there exists some $N<0$ such that $|\bar{G}(t)| < \varepsilon$ if $t < N$. Then since $\alpha>0$,

$$\lim_{x\to-\infty}\left|e^{\alpha x}\int_0^x \bar{G}(t)e^{-\alpha t}dt\right|$$

$$\leq \lim_{x\to-\infty} e^{\alpha x}\left|\varepsilon\int_x^N e^{-\alpha t}dt + \|\bar{G}\|_{(N,0)}\int_N^0 e^{-\alpha t}dt\right|$$

$$= \varepsilon/\alpha$$

ORIGINAL PAGE IS
OF POOR QUALITY

Hence, since the above ε is arbitrary and $\alpha>0$, the first summand of (2.3.11) converges to zero as x proceeds to negative infinity. Next, for the second summand of (2.3.11),

$$\lim_{x\to-\infty} \left| \left[B - \frac{1}{S} \int_0^x \bar{G} \, e^{-\beta t} dt \right] e^{\beta x} \right|$$

$$\leq \lim_{x\to-\infty} \left| \frac{1}{S} \int_{-\infty}^{y} \bar{G} \, e^{-\beta t} dt \; e^{\beta x} \right|$$

$$\leq \lim_{x\to-\infty} \frac{e^{-\infty} - e^{-\beta x}}{\beta} \frac{e^{\beta x}}{S} \|\bar{G}\|_{(-\infty,x)} \qquad = 0$$

because $\lim_{x\to-\infty} \bar{G} = 0$. Hence $\lim_{x\to-\infty} \bar{\theta}(x) = 0$.

A clever man understands the need for proof. --Proverbs 14:15

APPENDIX C.0 COMPUTER CODE LISTS

C.1 INTRODUCTION

The computer codes developed to solve Problems P1-1, P1-2, P1-3, and P2-1 are given in this appendix. The codes themselves contain numerous comments correlating portions of the codes with sections and equations in this report. The code for Problem P2-1 is listed in Appendix C.2 and is followed by the code for Problems P1-1 and P1-2 in Appendix C.3. To finish, the code for Problem P1-3 is listed in Appendix C.4.

C.2 COMPUTER CODE LIST FOR PROBLEM P2-1

The computer code for Problem P2-1 is listed in Figure C-1. Before using this code, the user should review the remarks given at the end of Appendix A.2.

ORIGINAL PAGE IS
OF POOR QUALITY

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC  PROGRAM PURPOSE-                                                  CC
CC    COMPUTE STEADY-STATE TEMPERATURE DISTRIBUTION AND THERMAL       CC
CC    GRADIENTS FOR FINITE LENGTH TRANSLATING CYLINDER, THIS PROBLEM  CC
CC    IS DESCRIBED IN DETAIL IN SECTION 2.2 OF FINAL REPORT (TO NASA) CC
CC    - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED   CC
CC    BOUNDARY CONDITIONS - BY SCIENCE APPLICATIONS, INC.             CC
CC  SOURCE-                                                           CC
CC    SCIENCE APPLICATIONS, INC.                                      CC
CC    HUNTSVILLE, ALABAMA                                             CC
CC  AUTHORS-                                                          CC
CC    LARRY M. FOSTER                                                 CC
CC    JOHN MCINTOSH                                                   CC
CC  REFERENCE-                                                        CC
CC    - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED   CC
CC    BOUNDARY CONDITIONS -                                           CC
CC    (FINAL REPORT - SAI-83/5034 + HU)                               CC
CC    SCIENCE APPLICATIONS, INC                                       CC
CC  REMARKS-                                                          CC
CC    - SOFTWARE DEVELOPED AND TESTED ON CDC 7600/6400 AND            CC
CC    UNIVAC 1108                                                     CC
CC    - ALL EQUATIONS REFERENCED IN CODE BELOW ARE CONTAINED IN THE   CC
CC    FINAL REPORT-                                                   CC
CC            - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF    CC
CC            SELECTED BOUNDARY CONDITIONS                            CC
CC  INPUT VARIABLES AND FUNCTIONS-                                    CC
CC    P               - PECLET NUMBER                                 CC
CC    MSUM            - NUMBER OF TERMS IN SERIES EXPANSION OF        CC
CC                    TEMPERATURE DISTRIBUTION (THE DESIRED SOLUTION) CC
CC    X0              - AXIAL POSITION OF LOWER END OF CYLINDER       CC
CC    XN              - AXIAL POSITION OF UPPER END OF CYLINDER       CC
CC    NGRID           - NUMBER OF DIVISIONS OF CYLINDER AXIS USED IN  CC
CC                    SOLUTION OF O. D. E. BOUNDARY VALUE PROBLEM     CC
CC                    RESULTING FROM TRANSFORMATION OF THE PDE MODELING CC
CC                    THE TEMPERATURE                                 CC
CC    NR              - NUMBER DIVISIONS OF CYLINDER RADIUS USED IN   CC
CC                    OUTPUT OF TEMPERATURE DISTRIBUTION              CC
CC    IHFC            - 1 IF A DISCRETE DATA POINT FORM OF THE SURFACE CC
CC                    TEMPERATURE IS USER PROVIDED                    CC
CC                    - 0 IF A USER DEFINED FUNCTIONAL FORM OF THE    CC
CC                    SURFACE TEMPERATURE IS PROVIDED                 CC
CC    (XD,YD)         - USER PROVIDED DATA PTS FOR THE AXIAL DISTANCE CC
CC                    (XD) AND CORRESPONDING SURFACE TEMPERATURE (YD) CC
CC    M               - NUMBER OF DATA PTS, INPUT IF IHFC = 1         CC
CC                    SET TO 0 IF IHFC = 0                            CC
CC    HFC             - USER PROVIDED (IF IHFC = 0) SURFACE TEMPERATURE CC
CC                    DISTRIBUTION                                    CC
CC    AFC             - USER PROVIDED TEMPERATURE DISTRIBUTION ON THE CC
CC                    LOWER (AXIAL POSITION = X0) END OF THE CYLINDER CC
CC    BFC             - USER PROVIDED TEMPERATURE DISTRIBUTION ON THE CC
CC                    UPPER (AXIAL POSITION = XN) END OF THE CYLINDER CC
CC  OUTPUT VARIABLES-                                                 CC
CC    THOLD           - TEMPERATURE DISTRIBUTION ARRAY IN THE CYLINDER CC
CC                    (FROM X0 TO XN AXIALLY, WITH NR DIVISIONS OF THE CC
CC                    RADIUS)                                         CC
CC    GRADX0          - AXIAL THERMAL GRADIENT ARRAY AT X0            CC
```

Figure C-1. Computer Code List for Problem P2-1

```
CC     GRADXN        - AXIAL THERMAL GRADIENT ARRAY AT XN                      CC
CC   USER SUPPLIED MATHMATICAL SOFTWARE-                                       CC
CC     - A LEAST SQUARES ALGORITHM TO SOLVE OVER POSED SYSTEMS OF             CC
CC     LINEAR EQUATIONS (REQUIRED IN SUBROUTINE COEFS.)                        CC
CC     - AN ALGORITHM TO EVALUATE BESSEL FUNCTIONS (REQUIRED IN               CC
CC     SUBROUTINE J0)                                                          CC
CC                                                                             CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL J1,J1LAM
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/C10/ASCRIP(20),BSCRIP(20)
      COMMON/READ1/P,MSUM,X0,XN,NGRID,NR
      COMMON/C5/P(101),PSI(20,101),SQ.I(20)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
      COMMON/C21/THETAB(20,505),THOLD(101),T(10),GRADXN(101),GRADX0(101)
      CHARACTER*17 RIS,ALPHA(6)
      CHARACTER*18 STARS,STAR(6)
      DATA RIS/'R=               '/,STARS/'*****************'/
      DO 210 L=1,6
      ALPHA(L)=RIS
      STAR(L)=STARS
  210 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                             CC
CC                                                                             CC
CC     RLAMD(M)=ROOT OF J0 BESSEL FCN                                          CC
CC     J1(M)=J1(RLAMD(M)) WHERE J1 IS BESSEL FCN                               CC
CC     J1LAM(M)=J1(M)/RLAMD(M)                                                 CC
CC                                                                             CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RLAMD( 1)=2.4048255577
      RLAMD( 2)=5.5200781103
      RLAMD( 3)=8.6537279129
      RLAMD( 4)=11.7915344391
      RLAMD( 5)=14.9309177086
      RLAMD( 6)=18.0710639679
      RLAMD( 7)=21.2116366299
      RLAMD( 8)=24.3524715308
      RLAMD( 9)=27.4934791320
      RLAMD(10)=30.6346064684
      RLAMD(11)=33.7758202136
      RLAMD(12)=36.9170983537
      RLAMD(13)=40.0584257646
      RLAMD(14)=43.1997917132
      RLAMD(15)=46.3411883717
      RLAMD(16)=49.4826098974
      RLAMD(17)=52.6240518411
      RLAMD(18)=55.7655107550
      RLAMD(19)=58.9069839261
      RLAMD(20)=62.0484691902
      J1( 1)=0.5191474973
      J1( 2)=-0.3402648065
      J1( 3)=0.2714522999
      J1( 4)=-0.2324598314
      J1( 5)=0.2065464331
      J1( 6)=-0.1877288030
      J1( 7)=0.1732658942
      J1( 8)=-0.1617015507
      J1( 9)=0.1521812138
      J1(10)=-0.1441659777
      J1(11)=0.1372969434
      J1(12)=-0.1313246267
      J1(13)=0.1260694971
      J1(14)=-0.1213986248
      J1(15)=0.1172111989
      J1(16)=-0.1134291926
```


Figure C-1. Computer Code List for Problem P2-1 (Cont)

```
      J1(17)=0.1099911430
      J1(18)=-0.1068478883
      J1(19)=0.1039595729
      J1(20)=-0.1012934989
C
      DO 10 I=1,20
        J1LAM(I)=J1(I)/RLAMD(I)
        SQJ1(I)=J1(I)*J1(I)
   10 CONTINUE
      CALL INPUT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    FIND COEFS FOR BESSEL EXPANSIONS OF A(R)-A(1) AND B(R)-B(1)     CC
CC    SEE EQUATIONS (2.2.17) AND (2.2.18) OF FINAL REPORT             CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL AFC(1.0,AOF1)
      CALL BFC(1.0,BOF1)
      DO 20 I=1,101
        R(I)=(I-1)*0.01
        RHOLD=R(I)
        CALL AFC(RHOLD,ANS)
        A(I)=ANS-AOF1
        CALL BFC(RHOLD,ANS)
        B(I)=ANS-BOF1
   20 CONTINUE
      CALL COEFS(R,A,101,20,ASCRIP)
      CALL COEFS(R,B,101,20,BSCRIP)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    SOLVE FOR THETA BAR OF EQUATIONS (2.2.19) BY SOLVING THE        CC
CC    TRIDIAGONAL SYSTEM (2.2.20) - SEE FINAL REPORT                  CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DX=(XN-X0)/NGRID
      DX2=DX*DX
      L=NGRID-1
      DO 30 M=1,MSUM
        DO 40 I=1,L
          A(I)=1.0+DX*P/2.0
          B(I)=-2.0-DX2*RLAMD(M)*RLAMD(M)
          C(I)=1.0-DX*P/2.0
      X=X0+I*DX
          CALL GBAR(M,X,ANS)
          D(I)=DX2*ANS
   40   CONTINUE
        D(1)=D(1)-(1.0+DX*P/2.0)*ASCRIP(M)*SQJ1(M)*0.5
        D(L)=D(L)-(1.0-DX*P/2.0)*BSCRIP(M)*SQJ1(M)*0.5
      CALL TRIDAG(L)
        DO 50 I=2,NGRID
          II=I-1
          THETAB(M,I)=V(II)
   50   CONTINUE
        NSTOP=NGRID+1
        THETAB(M,1)=ASCRIP(M)*SQJ1(M)/2.0
        THETAB(M,NSTOP)=BSCRIP(M)*SQJ1(M)/2.0
   30 CONTINUE
      DR=1.0/NR
      NRSTOP=NR+1
      DO 60 I=1,NRSTOP
        R(I)=(I-1)*DR
        DO 65 M=1,MSUM
        PSI(M,I)=F(M,R(I))
65      CONTINUE
60    CONTINUE
CC    PRINT TEMPERATURES                                              CC
```


Figure C-1. Computer Code List for Problem P2-1 (Cont)

```
      WRITE(6,70)
   70 FORMAT(1H1,45X,49H T E M P E R A T U R E    D I S T R I B U T I O
     1N)
      IFLAG=0
      MRIGHT=6
      MLEFT=1
  180 CONTINUE
      IF(NRSTOP.LE.MRIGHT)IFLAG=1
      MRIGHT=MIN0(NRSTOP,MRIGHT)
      WRITE(6,190)(R(J),J=MLEFT,MRIGHT)
  190 FORMAT(//////,1H ,17X,6(F12.8,5X))
      IHOLD=MRIGHT-MLEFT+1
      WRITE(6,267)(ALPHA(L),L=1,IHOLD)
  267 FORMAT(1H+,17X,6A17)
      WRITE(6,268)(STAR(L),L=1,IHOLD)
  268 FORMAT(1H0,15X,6A17)
      DO 200 I=1,NSTOP
        II=NSTOP+1-I
      X=X0+(II-1)*DX
        DO 202 J=MLEFT,MRIGHT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    DETERMINE TEMPERATURE AT (X,R(J))                                CC
CC    SEE EQUATION (2.2.14) OF FINAL REPORT                            CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          THOLD(J)=0.0
          DO 204 M=1,MSUM
            THOLD(J)=THOLD(J)+2.0*PSI(M,J)*THETAB(M,II)/SQJ1(M)
  204     CONTINUE
        CALL HFC(X,ANS)
          THOLD(J)=THOLD(J)+ANS
  202     CONTINUE
          WRITE(6,207)X,(THOLD(J),J=MLEFT,MRIGHT)
  207 FORMAT(3H X=,F10.6,5H    ,6(F15.8,2X))
  200   CONTINUE
        IF(IFLAG.EQ.1)GO TO 220
        MRIGHT=MRIGHT+6
        MLEFT=MLEFT+6
        GO TO 180
  220   CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    COMPUTE THERMAL GRADIENTS AT X=X0 AND XN                         CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,71)
   71 FORMAT(1H1,47X,35H T H E R M A L    G R A D I E N T S)
      WRITE(6,72)X0,XN
   72 FORMAT(///,44X,1HR,5X,11HGRAD. AT X=,F10.5,14H   GRAD. AT X=,F10.5
     2,//)
      DO 230 I=1,101
      R(I)=(I-1)*0.01
      DO 240 M=1,MSUM
      PSI(M,I)=F(M,R(I))
  240 CONTINUE
  230 CONTINUE
      DD=-DX
      DO 250 I=1,101
        DO 260 J=1,5
          T(J)=0.0
          DO 270 M=1,MSUM
            T(J)=T(J)+2.0*PSI(M,I)*THETAB(M,J)/SQJ1(M)
  270     CONTINUE
          X=X0+(J-1)*DX
          CALL HFC(X,ANS)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-1. Computer Code List for Problem P2-1 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
          T(J)=T(J)+ANS
260     CONTINUE
        DO 280 J=6,10
          T(J)=0.0
          JHOLD=NSTOP-10+J
            DO 290 M=1,MSUM
            T(J)=T(J)+2.0*PSI(M,I)*THETAB(M,JHOLD)/SQJ1(M)
290       CONTINUE
        X=X0+(JHOLD-1)*DX
        CALL HFC(X,ANS)
        T(J)=T(J)+ANS
280   CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    APPROXIMATE THERMAL GRADIENTS AT CYLINDER ENDS                  CC
CC    - SEE EQUATIONS (2.2.21) AND (2.2.22) OF FINAL REPORT           CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GRADXN(I)=(-3*T(6)+16*T(7)-36*T(8)+48*T(9)-25*T(10))/(12*DX)
      GRADX0(I)=(-3*T(5)+16*T(4)-36*T(3)+48*T(2)-25*T( 1))/(12*DX)
      WRITE(6,300)R(I),GRADX0(I),GRADXN(I)
  300 FORMAT(1H ,39X,F8.6,3X,E17.9,7X,E17.9)
250   CONTINUE
      STOP
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    THIS SUBROUTINE APPROXIMATES (BY FINITE DIFFERENCE) GBAR OF     CC
CC    EQUATION (2.2.16) OF FINAL REPORT                               CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE GBAR(M,X,ANS)
      REAL J1,J1LAM
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/READ1/P,MSUM,X0,XN,NGRID,NR
      EPSLON=0.01
      X1=X-EPSLON
      X2=X+EPSLON
      CALL HFC(X,ANS)
      CALL HFC(X1,ANS1)
      CALL HFC(X2,ANS2)
      G=P*(ANS2-ANS1)/(2.0*EPSLON)
      G=G-(ANS2+ANS1-2.0*ANS)/(EPSLON*EPSLON)
      ANS=G*J1LAM(M)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    THIS SUBROUTINE PROVIDES FOR DATA INPUT                         CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE INPUT
      COMMON/READ1/P,MSUM,X0,XN,NGRID,NR
      COMMON/C26/XD(100),YD(100),C1(4,100),M
      COMMON/C25/IHFC
      DIMENSION C(4,100)
      EQUIVALENCE(C1(1,1),C(1,1))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    SPLINE INPUT OPTION                                             CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,5)
    5 FORMAT(1H1,57X,20HI N P U T     D A T A)
      READ(5,16)IHFC,M
      IF(IHFC.NE.1) GOTO60
```

Figure C-1. Computer Code List for Problem P2-1 (Cont)

C-2

```
      WRITE(6,30)M
   30 FORMAT(//////,95H THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIM
     1ATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING,I4,23H  (X, TEMP) D
     *ATA POINTS,///,37H          X                    SURFACE TEMP.)
      DO 32 I=1,M
      READ(5,22)XD(I),YD(I)
   22 FORMAT(4E20.10)
   16 FORMAT(2I10)
      WRITE(6,34)XD(I),YD(I)
   34 FORMAT(1H ,2E20.10)
   32 CONTINUE
      CALL COFGEN
  40  CONTINUE
      READ(5,10)P,X0,XN,MSUM,NGRID,NR
   10 FORMAT(3F10.5,4I10)
      WRITE(6,20)P,X0,XN,MSUM,NGRID,NR
20    FORMAT(///,56H           P          X0          XN       MSUM   NGRID
     1  NR,//,1H ,3E12.4,I5,2I7)
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    THIS SUBROUTINE SUPPLIES LATERAL SURFACE TEMPERATURE            cc
cc    - SEE EQUATION (2.2.4) OF FINAL REPORT                          cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      SUBROUTINE HFC(X,ANS)
      COMMON/C25/IHFC
      DIMENSION C(7)
      IF(IHFC.EQ.1) GOTO60
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    USER SUPPLIED LATERAL SURFACE TEMPERATURE                       cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      RETURN
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    LATERAL SURFACE TEMPERATURE PROVIDED BY SPLINE                  cc
cc    FIT OF USER SUPPLIED DATA (XD,YD)                               cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
   60 CALL SPLINE(X,ANS)
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION        cc
cc    ON LOWER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL REPORT cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      SUBROUTINE AFC(R,ANS)
      COMMON/READ1/P,MSUM,X0,XN,NGRID,NR
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    USER SUPPLIED LOWER END TEMPERATURE A(R).                       cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      CALL HFC(X0,ANS)
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION        cc
cc    ON UPPER END OF CYLINDER - SEE EQUATION (2.2.3) OF FINAL REPORT cc
cc                                                                    cc
```


Figure C-1. Computer Code List for Problem P2-1 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      SUBROUTINE BFC(R,ANS)
      COMMON/READ1/P,HSUM,X0,XN,NGRID,NR
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    USER SUPPLIED UPPER END TEMPERATURE B(R).                       cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      CALL HFC(XN,ANS)
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    THIS SUBROUTINE FITS BESSEL SERIES TO DATA BY LEAST SQUARES     cc
cc    METHOD - SEE EQUATIONS (2.2.17) , (2.2.18) AND (2.2.23)         cc
cc    OF FINAL REPORT                                                 cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      SUBROUTINE COEFS(R,Y,NR,NCOEF,COEF)
      INTEGER NR,NCOEF
      REAL F,R(101),Y(101),COEF(20),WK(460)
      EXTERNAL F
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    USER SUPPLIED LEAST SQUARES METHOD FOLLOWS HERE TO DETERMINE    cc
cc    THE COEFFICIENTS OF EQUATIONS (2.2.17) AND (2.2.18). THE        cc
cc    SUBROUTINE IFLSQ BELOW IS THE IMSL LEAST SQUARES FUNCTION       cc
cc    FIT ROUTINE                                                     cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      CALL IFLSQ(F,R,Y,NR,COEF,NCOEF,WK,IER)
      IF(IER.EQ.129.OR.IER.EQ.130)WRITE(6,10)
   10 FORMAT(54H TERMINAL ERROR IN LEAST SQUARES METHOD,SUBROUTINE COEFS
     1)
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    THIS FUNCTION EVALUATES THE ZERO ORDER BESSEL FUNCTION          cc
cc    DENOTED IN NOTATION N2=1 (III) - SEE FINAL REPORT               c
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      REAL FUNCTION F(N,R)
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      X=RLAMD(N)*R
      CALL J0(X,Y)
      F=Y
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    THIS SUBROUTINE COMPUTES THE J0 BESSEL FUNCTION Y=J0(X)         cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      SUBROUTINE J0(X,Y)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                    cc
cc    USER SUPPLIED J0 FUNCTION PLACED HERE. IN THIS EXAMPLE, THE     cc
cc    IMSL BESSEL FUNCTION MMBSJ0 IS ILLUSTRATED                      cc
cc                                                                    cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      REAL MMBSJ0
      Y=MMBSJ0(X,IER)
      RETURN
      END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
```

Figure C-1. Computer Code List for Problem P2-1 (Cont)

```
CC                                                                    CC
CC     SUBROUTINE FOR SOLVING A SYSTEM OF LINEAR SIMULTANEOUS           CC
CC     EQUATIONS HAVING A TRIDIAGONAL COEFFICIENT MATRIX. DIAGONALS     CC
CC     ARE STORED IN THE ARRAYS A, B, AND C. THE COMPUTED              CC
CC     SOLUTION VECTOR V(1) . . . V(L) IS STORED IN THE ARRAY V.        CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE TRIDAG(L)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
      BETA(1)=B(1)
      GAMMA(1)=D(1)/BETA(1)
      IFP1=2
      DO 1 I=IFP1,L
        BETA(I)=B(I)-A(I)*C(I-1)/BETA(I-1)
        GAMMA(I)=(D(I)-A(I)*GAMMA(I-1))/BETA(I)
    1 CONTINUE
      V(L)=GAMMA(L)
      LAST=L-1
      DO 2 K=1,LAST
        I=L-K
        V(I)=GAMMA(I)-C(I)*V(I+1)/BETA(I)
    2 CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC     THIS SUBROUTINE ESTIMATES LATERAL SURFACE TEMPERATURE           CC
CC     BY USE OF CUBIC SPLINE IF THE USER SUPPLIES A DISCRETE SET OF    CC
CC     LATERAL SURFACE TEMPERATURES.                                  CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SPLINE(XINT,YINT)
      COMMON/C26/XD(100),YD(100),C1(4,100),M
      DIMENSION C(4,100)
      EQUIVALENCE(C1(1,1),C(1,1))
      IF(XINT-XD(1))2,1,2
    1 YINT=YD(1)
      RETURN
    2 K=1
    3 IF(XINT-XD(K+1))6,4,5
    4 YINT=YD(K+1)
      RETURN
    5 K=K+1
      IF((M-K).GT.0) GOTO3
      IF((M-K).LE.0) K=M-1
    6 YINT=(XD(K+1)-XINT)*(C(1,K)*(XD(K+1)-XINT)**2+C(3,K))
      YINT=YINT+(XINT-XD(K))*(C(2,K)*(XINT-XD(K))**2+C(4,K))
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC     FIND THE SPLINE CURVE FIT COEFFICIENTS. FOR USE IN CONJUNCTION  CC
CC     WITH SUBROUTINE SPLINE.                                         CC
CC     INPUTS -                                                        CC
CC     M    = NO. OF DATA PAIRS                                        CC
CC     XD   = ARRAY OF X (ABCISSA) VALUES                              CC
CC     YD   = ARRAY OF Y (ORDINATES) VALUES                            CC
CC     OUTPUTS -                                                       CC
CC     C    = 2-D ARRAY OF SPLINE FIT COEFFICIENTS (4 COEFFICIENTS     CC
CC            PER TRIPLET OF DATA POINTS).                             CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE COFGEN
      COMMON/C26/XD(100),YD(100),C1(4,100),M
      DIMENSION C(4,100)
      DIMENSION P(100),E(100),A(100,3),B(100),Z(100),D(100)
```


Figure C-1. Computer Code List for Problem P2-1 (Cont)

```
      EQUIVALENCE(C1(1,1),C(1,1))
C
      ND=M
      M=M-1
       DO 2 K=1,M
       D(K)=XD(K+1)-XD(K)
       P(K)=D(K)/6.
    2  E(K)=(YD(K+1)-YD(K))/D(K)
       DO 3 K=2,M
    3  B(K)=E(K)-E(K-1)
       A(1,2)=-1.-D(1)/D(2)
       A(1,3)=D(1)/D(2)
       A(2,2)=2.*(P(1)+P(2))-P(1)*A(1,2)
       A(2,3)=(P(2)-P(1)*A(1,3))/A(2,2)
       B(2)=B(2)/A(2,2)
       DO 4 K=3,M
       A(K,2)=2.*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
       B(K)=B(K)-P(K-1)*B(K-1)
       A(K,3)=P(K)/A(K,2)
    4  B(K)=B(K)/A(K,2)
       Q=D(M-1)/D(M)
       A(ND,1)=1.+Q+A(M-1,3)
       A(ND,2)=-Q-A(ND,1)*A(M,3)
       B(ND)=B(M-1)-A(ND,1)*B(M)
       Z(ND)=B(ND)/A(ND,2)
       DO 6 I=1,ND-2
       K=ND-I
    6  Z(K)=B(K)-A(K,3)*Z(K+1)
       Z(1)=-A(1,2)*Z(2)-A(1,3)*Z(3)
       DO 7 K=1,M
       Q=1./(6.*D(K))
       C(1,K)=Z(K)*Q
       C(2,K)=Z(K+1)*Q
       C(3,K)=YD(K)/D(K)-Z(K)*P(K)
    7  C(4,K)=YD(K+1)/D(K)-Z(K+1)*P(K)
      M=M+1
       RETURN
C
C      END COEFGEN
       END
```


Figure C-1. Computer Code List for Problem P2-1 (Cont)

C.3 COMPUTER CODE LIST FOR PROBLEMS P1-1 and P1-2

The computer code for Problems P1-1 and P1-2 is listed in Figure C-2. Before using this code, the user should review the remarks made at the end of Appendix A.3.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC  PROGRAM PURPOSE-                                                   CC
CC    COMPUTE THE UPPER AND LOWER REGIONS SURFACE CONTROL FUNCTIONS    CC
CC    SUCH THAT FLAT SOLID-MELT INTERFACES ARE ACHIEVED AS FORMULATED  CC
CC    IN PROBLEMS P1-1 AND P1-2 OF FINAL REPORT (TO NASA) THE CONTROL  CC
CC  OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED BOUNDARY           CC
CC    CONDITIONS- BY SCIENCE APPLICATIONS, INC.                        CC
CC  SOURCE-                                                            CC
CC    SCIENCE APPLICATIONS, INC,                                       CC
CC    HUNTSVILLE, ALABAMA                                              CC
CC  AUTHORS-                                                           CC
CC    LARRY M. FOSTER                                                  CC
CC    JOHN MCINTOSH                                                    CC
CC  REFERENCE-                                                         CC
CC    - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED    CC
CC    BOUNDARY CONDITIONS -                                            CC
CC    (FINAL REPORT - SAI-83/5034+HU)                                  CC
CC    SCIENCE APPLICATIONS                                             CC
CC  REMARKS-                                                           CC
CC    - SOFTWARE DEVELOPED AND TESTED ON CDC 7600/6400                 CC
DD    UNIVAC 1108                                                      CC
CC    - ALL EQUATIONS REFERENCED IN CODE BELOW ARE CONTAINED IN THE    CC
CC    FINAL REPORT-                                                    CC
CC            - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF     CC
CC            SELECTED BOUNDARY CONDITIONS                             CC
CC  INPUT VARIABLES AND FUNCTIONS-                                     CC
CC    P              - PECLET NUMBER                                   CC
CC    MSUM           - NUMBER OF TERMS IN SERIES EXPANSION OF          CC
CC                   TEMPERATURE DISTRIBUTION (THE DESIRED SOLUTION)   CC
CC    X0             - AXIAL POSITION OF LOWER END OF CYLINDER         CC
CC    XN             - AXIAL POSITION OF UPPER END OF CYLINDER         CC
CC    NGRID          - NUMBER OF GRID POINTS USED IN NUMERICAL         CC
CC                   SOLUTION OF O. D. E. BOUNDARY VALUE PROBLEM       CC
CC                   RESULTING FROM TRANSFORMATION OF PDE THE MODELING CC
CC                   TEMPERATURE                                       CC
CC    NR             - NUMBER DIVISIONS OF CYLINDER RADIUS USED IN     CC
CC                   OUTPUT OF TEMPERATURE DISTRIBUTION                CC
CC    RKS            - SOLID THERMAL CONDUCTIVITY                      CC
CC    RKL            - LIQUID THERMAL CONDUCTIVITY                     CC
CC    RL             - PRODUCT OF CRYSTAL GROWTH RATE, SOLID DENSITY,  CC
CC                   AND LATENT HEAT OF FUSION                         CC
CC    IHFC           - 1 IF A DISCRETE DATA POINT FORM OF THE SURFACE  CC
CC                   TEMPERATURE IS USER PROVIDED                      CC
CC                   0 IF A USER DEFINED FUNCTIONAL FORM OF THE        CC
CC                   SURFACE TEMPERATURE IS PROVIDED                   CC
CC    (XD,YD)        - USER PROVIDED DATA PTS FOR THE AXIAL DISTANCE   CC
CC                   (XD) AND CORRESPONDING SURFACE TEMPERATURE (YD)   CC
CC    M              - NUMBER OF DATA PTS. INPUT IF IHFC = 1           CC
CC                   SET TO 0 IF IHFC = 0                              CC
CC    HFC            - USER PROVIDED (IF IHFC = 0) SURFACE TEMPERATURE CC
CC                   FUNCTION                                          CC
CC    CASE LIMITS    - THIS PROGRAM GENERATES THE SURFACE CONTROL      CC
CC                   FUNCTIONS FOR VARIOUS COMBINATIONS OF THE INDEX   CC
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2

ORIGINAL PAGE IS
OF POOR QUALITY

```
CC                    LIMITS MTERM AND NSYS (SEE EQ. (4.0.19) - (4.0.23)  CC
CC                    OF FINAL REPORT), TO DEFINE THESE COMBINATIONS      CC
CC                         MINTERM    - THE MINIMUM ALLOWED VALUE         CC
CC                                      OF MTERM                          CC
CC                         MAXTERM    - THE MAXIMUM ALLOWED VALUE         CC
CC                                      OF MTERM                          CC
CC                         DELTERM    - INCREMENT OF MTERM FROM MINTERM   CC
CC                                      TO MAXTERM                        CC
CC                         MINNSYS    - THE MINIMUM ALLOWED VALUE OF NSYS CC
CC                         MAXNSYS    - THE MAXIMUM ALLOWED VALUE OF NSYS CC
CC                         DELTERM    - INCREMENT OF NSYS FROM MINNSYS    CC
CC                                      TO MAXNSYS                        CC
CC       IOPTION          - 0 IF SOLID SURFACE CONTROL FUNCTION IS TO     CC
CC                        REMAIN UNCHANGED                                CC
CC                        - 1 IF SOLID SURFACE CONTROL FUNCTION IS TO BE  CC
CC                        CLIPPED (SEE DEFN. IN APPENDIS A.3) AT ITS      CC
CC                        MINIMUM VALUE (MINIMUM VALUE FOUND IN SUBROUTINE CC
CC                        LINSRCH)                                        CC
CC                        - 2 IF FUNCTION IS TO BE CLIPPED (SEE DEFN. IN  CC
CC                        APPENDIS A93) AT SOME USER SPECIFIED VALUE (SEE CC
CC                        CLIP)                                           CC
CC       CLIP             - USER SUPPLIED VALUE OF MODIFIED SURFACE CONTROL CC
CC                        FUNCTION (SEE APPENDIX A.3 FOR DEFN.)           CC
CC    OUTPUT VARIABLES=                                                   CC
CC       THOLD            - TEMPERATURE DISTRIBUTION ARRAY FOR EACH REGION CC
CC                        - SEE SUBROUTINE MELT OF CODE                   CC
CC       GRAD2            - MELT ZONE LOWER INTERFACE GRADIENT            CC
CC       GRAD3            - MELT ZONE UPPER INTERFACE GRADIENT            CC
CC       GRADX0           - THERMAL GRADIENT AT X0 FOR EACH REGION        CC
CC       GRADXN           - THERMAL GRADIENTS AT XN FOR EACH REGION       CC
CC       COEF             - ARRAY OF COEFFICIENTS OF SOLID REGIONS SURFACE CC
CC                        CONTROL. (SEE EQUATIONS (3.0.23) AND (3.0.31) )
CC       ERRL2            - THE L2 RELATIVE DIFFERENCE BETWEEN THE REQUIRED CC
CC                        SOLID REGIONS INTERFACE GRADIENTS AND THE       CC
CC                        INTERFACE GRADIENTS RESULTING FROM THE USE OF   CC
CC                        THE SOLID REGIONS SURFACE CONTROL FUNCTIONS.    CC
CC    USER SUPPLIED MATHMATICAL SOFTWARE=                                 CC
CC       - A LEAST SQUARES ALGORITHM TO FIT A FUNCTION TO A LINEAR        CC
CC       COMBINATION OF SELECTED FUNCTIONS (REQUIRED IN SUBROUTINE        CC
CC       COEFS.)                                                          CC
CC       - AN ALGORITHM TO EVALUATE BESSEL FUNCTIONS (REQUIRED IN         CC
CC       SUBROUTINE J0)                                                   CC
CC       - A NUMERICAL INTEGRATION ROUTINE (REQUIRED IN SUBROUTINES       CC
CC       INTEGL1 AND INTEGL2)                                             CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      INTEGER DELTERM,DELNSYS
      REAL J1,J1LAM,MMBSJ0
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/C5/R(101),PSI(20,101),SQJ1(20)
      COMMON/C9/COEF(20),RM
      COMMON/C10/ASCRIP(20),BSCRIP(20)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
      COMMON/C21/THETAB(20,505),THOLD(101),T(10),GRADXN(101),GRADX0(101)
      COMMON/C22/ICASE,THELT(3)
      COMMON/C23/GRAD2(101),GRAD3(101)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/C25/GRADAT0(101),GRADATN(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(40,10),RHS(40),WWK(1500),IIWK(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/FIXPT/IOPTION,XMIN,GMIN,CLIP
      COMMON/C30/MINTERM,MAXTERM,DELTERM,MINNSYS,MAXNSYS,DELNSYS
      COMMON/C31/THFC
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      COMMON/C76/IFLAG2,IFLAG3
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
CC                                                                            CC
CC     COMPUTE THERMAL DISTRIBUTION AND INTERFACE GRADIENTS FOR               CC
CC     MELT ZONE                                                              CC
CC                                                                            CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IFLAG2=1
      IFLAG3=1
      ICASE=1
      WRITE(6,983)
  983 FORMAT(1H1,54X,18HM E L T    Z O N E)
      CALL INPUT
      CALL HFC(X0,ANS)
      TMELT(2)=ANS
      CALL HFC(XN,ANS)
      TMELT(3)=ANS
      CALL MELT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                            CC
CC     USING THE LOWER SOLID REGION SURFACE CONTROL FUNCTION, COMPUTE         CC
CC     THE LOWER SOLID REGION THERMAL DISTRIBUTION AND INTERFACE              CC
CC     GRADIENT.                                                              CC
CC                                                                            CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IHFC=0
      ICASE=2
      WRITE(6,30)
   30 FORMAT(1H1,52X,22HL O W E R    S O L I D)
      CALL INPUT
      DO 21 MTER =MINTERM,MAXTERM,DELTERM
      MTERM=MAXTERM+MINTERM-MTER
      DO 22 NSY =MINNSYS,MAXNSYS,DELNSYS
      NSYS=MAXNSYS+MINNSYS-NSY
      NN=MTERM+2
      IF(NN.LT.NSYS)GO TO 22
      CALL SOLID2
      IF(IOPTION.EQ.0) GOTO70
      CALL LINSRCH(XMIN)
      CALL FUNC(XMIN,GMIN)
      IF(XMIN.GT.0.0.AND.GMIN.LE.0.0) GOTO40
      XMIN=100000.0
   40 XMIN=YN-XMIN
   70 CONTINUE
      CALL MELT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                            CC
CC     DETERMINE RELATIVE DIFFERENCE BETWEEN REQUIRED LOWER SOLID             CC
CC     REGION INTERFACE GRADIENT AND THE INTERFACE GRADIENT RESULTING         CC
CC     FROM USE OF THE LOWER SOLID REGION SURFACE CONTROL FUNCTION            CC
CC                                                                            CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL ERROR
      WRITE(6,789) MTERM,NSYS
  789 FORMAT(//,1H ,50X,12HFOR MTERM = ,I2,12H AND NSYS = ,I2)
22    CONTINUE
21    CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                            CC
CC     USING THE UPPER SOLID REGION SURFACE CONTROL FUNCTION, COMPUTE         CC
CC     THE UPPER SOLID REGION THERMAL DISTRIBUTION AND INTERFACE              CC
CC     GRADIENT.                                                              CC
CC                                                                            CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ICASE=3
      WRITE(6,10)
   10 FORMAT(1H1,52X,22HU P P E R    S O L I D)
      CALL INPUT
```

Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
      DO 31 MTER =MINTERM,MAXTERM,DELTERM
      MTERM=MAXTERM+MINTERM-MTER
      DO 32 NSY =MINNSYS,MAXNSYS,DELNSYS
      NSYS=MAXNSYS+MINNSYS-NSY
      NN=MTERM+2
      IF(NN.LT.NSYS)GO TO 32
      CALL SOLID3
      IF(IOPTION.EQ.0) GOTO80
      CALL LINSRCH(XMIN)
      CALL FUNC(XMIN,GMIN)
      IF(XMIN.GT.0.0.AND.GMIN.LE.0.0) GOTO90
      XMIN=100000.0
   90 XMIN=X0+XMIN
   80 CONTINUE
      CALL MELT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    DETERMINE RELATIVE DIFFERENCE BETWEEN REQUIRED UPPER SOLID     CC
CC    REGION INTERFACE GRADIENT AND THE INTERFACE GRADIENT RESULTING CC
CC    FROM USE OF THE UPPER SOLID REGION SURFACE CONTROL FUNCTION    CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL ERROR
      WRITE(6,120)MTERM,NSYS
  120 FORMAT(//,1H ,50X,12HFOR MTERM = ,I2,12H AND NSYS = ,I2)
32    CONTINUE
31    CONTINUE
      STOP
      END
      SUBROUTINE MELT
C
C
      REAL J1,J1LAM,MMBSJ0
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/C10/ASCRIP(20),BSCRIP(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/C5/R(101),PSI(20,101),SQJ1(20)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
      COMMON/C21/THETAB(20,505),THOLD(101),T(10),GRADXN(101),GRADX0(101)
      COMMON/C22/TCASE,TMELT(3)
      COMMON/C23/GRAD2(101),GRAD3(101)
      CHARACTER*17 RIS,ALPHA(6)
      CHARACTER*18 STARS,STAR(6)
      DATA RIS/'R=               '/,STARS/'******************'/
      DO 207 L=1,6
      ALPHA(L)=RIS
      STAR(L)=STARS
  207 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    RLAMD(M)=ROOT OF J0 BESSEL FCN                                 CC
CC    J1(M)=J1(RLAMD(M)) WHERE J1 IS BESSEL FCN                      CC
CC    J1LAM(M)=J1(M)/RLAMD(M)                                        CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RLAMD( 1)=2.4048255577
      RLAMD( 2)=5.5200781103
      RLAMD( 3)=8.6537279129
      RLAMD( 4)=11.7915344391
      RLAMD( 5)=14.9309177086
      RLAMD( 6)=18.0710639679
      RLAMD( 7)=21.2116366299
      RLAMD( 8)=24.3524715308
      RLAMD( 9)=27.4934791320
      RLAMD(10)=30.6346064684
      RLAMD(11)=33.7758202136
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
      RLAMD(12)=36.9170983537
      RLAMD(13)=40.0584257646
      RLAMD(14)=43.1997917132
      RLAMD(15)=46.3411883717
      RLAMD(16)=49.4826098974
      RLAMD(17)=52.6240518411
      RLAMD(18)=55.7655107550
      RLAMD(19)=58.9069839261
      RLAMD(20)=62.0484691902
      J1( 1)=0.5191474973
      J1( 2)=-0.3402648065
      J1( 3)=0.2714522999
      J1( 4)=-0.2324598314
      J1( 5)=0.2065464331
      J1( 6)=-0.1877288030
      J1( 7)=0.1732658942
      J1( 8)=-0.1617015507
      J1( 9)=0.1521812138
      J1(10)=-0.1441659777
      J1(11)=0.1372969434
      J1(12)=-0.1313246267
      J1(13)=0.1260694971
      J1(14)=-0.1213986248
      J1(15)=0.1172111989
      J1(16)=-0.1134291926
      J1(17)=0.1099911430
      J1(18)=-0.1068478883
      J1(19)=0.1039595729
      J1(20)=-0.1012934989
      DO 555 I=1,20
        J1LAM(I)=J1(I)/RLAMD(I)
        SQJ1(I)=J1(I)*J1(I)
  555 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    FIND COEFS FOR BESSEL EXPANSIONS OF A(R)-A(1) AND B(R)-B(1)    CC
CC    SEE EQUATIONS (2.2.17) AND (2.2.18) OF FINAL REPORT            CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL AFC(1.0,AOF1)
      CALL BFC(1.0,BOF1)
      DO 20 I=1,101
        R(I)=(I-1)*0.01
        RHOLD=R(I)
        CALL AFC(RHOLD,ANS)
        A(I)=ANS-AOF1
        CALL BFC(RHOLD,ANS)
        B(I)=ANS-BOF1
   20 CONTINUE
      CALL COEFS(R,A,101,20,ASCRIP)
      CALL COEFS(R,B,101,20,BSCRIP)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    SOLVE FOR THETA BAR OF EQUATIONS (2.2.19) BY SOLVING THE        CC
CC    TRIDIAGONAL SYSTEM (2.2.20) - SEE FINAL REPORT                 CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DX=(XN-X0)/NGRID
      DX2=DX*DX
      L=NGRID-1
      DO 556 M=1,MSUM
        DO 40 I=1,L
          A(I)=1.0+DX*P/2.0
          B(I)=-2.0-DX2*RLAMD(M)*RLAMD(M)
          C(I)=1.0-DX*P/2.0
      X=X0+I*DX
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
          CALL GBAR(M,X,ANS)
          D(I)=DX2*ANS
   40   CONTINUE
        D(1)=D(1)-(1.0+DX*P/2.0)*ASCRIP(M)*SQJ1(M)*0.5
        D(L)=D(L)-(1.0-DX*P/2.0)*BSCRIP(M)*SQJ1(M)*0.5
      CALL TRIDAG(L)
        DO 50 I=2,NGRID
          II=I-1
          THETAB(M,I)=V(II)
   50   CONTINUE
        NSTOP=NGRID+1
        THETAB(M,1)=ASCRIP(M)*SQJ1(M)/2.0
        THETAB(M,NSTOP)=BSCRIP(M)*SQJ1(M)/2.0
  956 CONTINUE
      GOTO999
  999 CONTINUE
      DR=1.0/NR
      NRSTOP=NR+1
      DO 60 I=1,NRSTOP
        R(I)=(I-1)*DR
        DO 65 M=1,MSUM
        VAR=R(I)*RLAMD(M)
        CALL JO(VAR,Y)
        PSI(M,I)=Y
65      CONTINUE
60    CONTINUE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC    PRINT TEMPERATURES                                                  CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GO TO (21,22,23) ICASE
   21 WRITE(6,983)
  983 FORMAT(1H1,54X,18HM E L T     Z O N E)
      GOTO24
   22 WRITE(6,30)
   30 FORMAT(1H1,52X,22HL O W E R     S O L I D)
      GOTO24
   23 WRITE(6,10)
   10 FORMAT(1H1,52X,22HU P P E R     S O L I D)
   24 CONTINUE
      WRITE(6,70)
   70 FORMAT(1H ,45X,50HT E M P E R A T U R E     D I S T R I B U T I O N
     * )
      IFLAG=0
      MRIGHT=6
      MLEFT=1
180   CONTINUE
      IF(NRSTOP.LE.MRIGHT)IFLAG=1
      MRIGHT=MINO(NRSTOP,MRIGHT)
      WRITE(6,190)(R(J),J=MLEFT,MRIGHT)
  190 FORMAT(/////,1H ,17X,6(F12.8,5X))
      WRITE(6,267) (ALPHA(L),L=1,MRIGHT)
  267 FORMAT(1H+,17X,6A17)
      WRITE(6,268) (STAR(L),L=1,MRIGHT)
  268 FORMAT(1H0,15X,6A17)
      DO 200 I=1,NSTOP
      ISKIP=I-1
      IHOLD=(ISKIP+.000001)/10.0
      XHOLD=(ISKIP/10.0)-IHOLD
      IF(XHOLD.GT.0.005) GOTO200
        II=NSTOP+1-I
      X=X0+(II-1)*DX
        DO 202 J=MLEFT,MRIGHT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
CC                                                                      CC
CC     DETERMINE TEMPERATURE AT (X,R(J))                                CC
CC     SEE EQUATION (2.2.14) OF FINAL REPORT                            CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          THOLD(J)=0.0
          DO 204 M=1,MSUM
            THOLD(J)=THOLD(J)+2.0*PSI(M,J)*THETAB(M,I)/SQJ1(M)
204       CONTINUE
        CALL HFC(X,ANS)
          THOLD(J)=THOLD(J)+ANS
202       CONTINUE
          WRITE(6,210)X,(THOLD(J),J=MLEFT,MRIGHT)
  210 FORMAT(3H X=,F10.6,5H    * ,6(E15.8,2X))
200     CONTINUE
        IF(IFLAG.EQ.1)GO TO 220
        MRIGHT=MRIGHT+6
        MLEFT=MLEFT+6
        GO TO 180
220     CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC     COMPUTE THERMAL GRADIENTS AT X=X0 AND XN                         CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GO TO (31,32,33) ICASE
   31 WRITE(6,983)
      GOTO34
   32 WRITE(6,30)
      GOTO34
   33 WRITE(6,10)
   34 CONTINUE
      WRITE(6,71)
   71 FORMAT(1H ,47X,35H T H E R M A L     G R A D I E N T S)
      WRITE(6,72)X0,XN
   72 FORMAT(///,44X,1HR,5X,11HGRAD  AT X=,F10.5,14H   GRAD. AT X=,F10.5
     2,//)
      DO 230 I=1,101
      R(I)=(I-1)*0.01
      DO 240 M=1,MSUM
        VAR=R(I)*RLAMD(M)
        CALL J0(VAR,Y)
        PSI(M,I)=Y
240   CONTINUE
230   CONTINUE
      DD=-DX
      DO 250 I=1,101
        DO 260 J=1,5
          T(J)=0.0
          DO 270 M=1,MSUM
            T(J)=T(J)+2.0*PSI(M,I)*THETAB(M,J)/SQJ1(M)
270       CONTINUE
          X=X0+(J-1)*DX
          CALL HFC(X,ANS)
          T(J)=T(J)+ANS
260     CONTINUE
        DO 280 J=6,10
          T(J)=0.0
          JHOLD=NSTOP-10+J
            DO 290 M=1,MSUM
            T(J)=T(J)+2.0*PSI(M,I)*THETAB(M,JHOLD)/SQJ1(M)
290       CONTINUE
        X=X0+(JHOLD-1)*DX
        CALL HFC(X,ANS)
        T(J)=T(J)+ANS
280   CONTINUE
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC     APPROXIMATE THERMAL GRADIENTS AT CYLINDER ENDS                     CC
CC     - SEE EQUATIONS (2.2.24) AND (2.2.22) OF FINAL REPORT              CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GRADXN(I)=(-3*T(6)+16*T(7)-36*T(8)+48*T(9)-25*T(10))/(12*DX)
      GRADX0(I)=(-3*T(5)+16*T(4)-36*T(3)+48*T(2)-25*T( 1))/(12*DX)
      ISKIP=I-1
      IHOLD=(ISKIP+.000001)/10.0
      XHOLD=(ISKIP/10.0)-IHOLD
      IF(XHOLD.GT.0.005) GOTO251
      WRITE(6,300)R(I),GRADX0(I),GRADXN(I)
  300 FORMAT(1H ,39X,F8.6,3X,E17.9,7X,E17.9)
251   CONTINUE
      IF(ICASE.NE.1)GO TO 250
      GRAD2(I)=GRADX0(I)
      GRAD3(I)=GRADXN(I)
250   CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC     THIS SUBROUTINE APPROXIMATES (BY FINITE DIFFERENCE) G BAR OF       CC
CC     EQUATION (2.2.16) OF FINAL REPORT                                  CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE GBAR(M,X,ANS)
      REAL J1,J1LAM
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      EPSLON=0.01
      X1=X-EPSLON
      X2=X+EPSLON
      CALL HFC(X,ANS)
      CALL HFC(X1,ANS1)
      CALL HFC(X2,ANS2)
      G=P (ANS2-ANS1)/(2.0*EPSLON)
      G=G-(ANS2+ANS1-2.0*ANS)/(EPSLON*EPSLON)
      ANS=G*J1LAM(M)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC   PURPOSE                                                              CC
CC     - PROVIDE INPUT DATA FOR SOFTWARE                                  CC
CC     - SEE APPENDIX A.3 FOR DETAILS                                     CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE INPUT
      INTEGER DELTERM,DELNSYS
      COMMON/C22/ICASE,THELT(3)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/FIXPT/IOPTION,XMIN,GMIN,CLIP
      COMMON/C31/THFC
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      COMMON/C30/MINTERM,MAXTERM,DELTERM,MINNSYS,MAXNSYS,DELNSYS
      DIMENSION C(4,100)
      DIMENSION XHOLD(100),YHOLD(100)
      EQUIVALENCE(C1(1,1),C(1,1))
      WRITE(6,5)
    5 FORMAT(//,1H ,56X,20HI N P U T    D A T A)
      IF(ICASE.NE.1) GOTO60
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

```
CC     INPUT MELT ZONE SURFACE TEMP. DISTRIBUTION IN A DATA SET             CC
CC     FORMAT FOR USE IN A CUBIC SPLINE                                     CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       READ(5,80)IHFC
   80  FORMAT(I2)
       IF(IHFC.NE.1) GOTO60
       READ(5,499) M
  499  FORMAT(I5)
       WRITE(6,30)M
   30  FORMAT(//////,95H THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIM
      *ATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING,I4,23H  (X, TEMP) D
      *ATA POINTS,///,37H           X                  SURFACE TEMP.)
       DO 32 I=1,M
       READ(5,22)XD(I),YD(I)
   22  FORMAT(2E20.10)
       WRITE(6,34)XD(I),YD(I)
   34  FORMAT(2E20.10)
   32  CONTINUE
       CALL COFGEN
   60  CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC     INPUT MELT ZONE PARAMETERS                                           CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       READ(5,10)P,X0,XN,MTERM,MSUM,NGRID,NR
   10  FORMAT(3F10.5,4I10)
       WRITE(6,953)
  953  FORMAT(/,1H ,14X,1HP,20X,2HX0,18X,2HXN,7X,5HMTERM,6X,4HMSUM,5X,5HN
      *GRID,6X,2HNR)
       WRITE(6,20)P,X0,XN,MTERM,MSUM,NGRID,NR
   20  FORMAT(1H ,3E20.10,4I10)
       IF(ICASE.EQ.1)RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC     INPUT MATERIAL CONDUCTIVITIES                                        CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       READ(5,90)RKS,RKL,RL,NSYS
90     FORMAT(3E20.10,I10)
       WRITE(6,888)
  888  FORMAT(//,1H ,10X,3HRKS,17X,3HRKL,17X,2HRL)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC     INPUT MATERIAL CONDUCTIVITIES                                        CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       WRITE(6,40) RKS,RKL,RL
   40  FORMAT(1H ,3E20.10)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC     INPUT CASE LIMITS                                                    CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       READ(5,21)MINTERM,MAXTERM,DELTERM,MINNSYS,MAXNSYS,DELNSYS
21     FORMAT(8I10)
       WRITE(6,799)
  799  FORMAT(//,1H ,9X,57HMAXTERM    MINTERM    DELTERM    MAXNSYS    MINNSY
      *S    DELNSYS)
       WRITE(6,18)MAXTERM,MINTERM,DELTERM,MAXNSYS,MINNSYS,DELNSYS
   18  FORMAT(1H ,5X,6(5X,I5))
       READ(5,50)IOPTION,CLIP
   50  FORMAT(I10,F10.5)
       RETURN
       END
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC  PURPOSES                                                            CC
CC    * PROVIDE USER ENTRY OF FUNCTIONAL FORM OF MELT ZONE SURFACE      CC
CC    TEMP. DISTRIBUTION                                                CC
CC    * EVALUATE SOLID REGIONS SURFACE CONTROL FUNCTIONS                CC
CC    * MODIFY SOLID REGIONS SURFACE CONTROL FUNCTIONS USING IOPTION    CC
CC    AND CLIP AS DETAILED IN APPENDIX A.3                              CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE HFC(X,ANS)
      COMMON/C9/COEF(20),RH
      COMMON/C22/ICASE,TMELT(3)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/FIXPT/IOPTION,XMIN,GMIN,CLIP
      COMMON/C30/ICKOUT
      COMMON/C26/CPOLY(20)
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      COMMON/C31/IHFC
      DIMENSION C(4,100)
      DIMENSION Z(20)
      EQUIVALENCE(C1(1,1),C(1,1))
      IF(IHFC.EQ.1) GOTO60
      IF(ICASE.EQ.2)GOTO20
      IF(ICASE.EQ.3) GOTO40
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    PLACE USER SUPPLIED MELT ZONE SURFACE TEMP HERE                   CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RETURN
   60 CALL SPLINE(X,ANS)
      RETURN
   20 CONTINUE
      ANS=0.0
      DO 10 K=1,NSYS
      Z(K)=(1-K)*(XN-X)
      RHOLD=0.0
      IF(Z(K).GT.-250.0) RHOLD=EXP(Z(K))
      ANS=ANS+COEF(K)*RHOLD
   10 CONTINUE
      IF(IOPTION.EQ.0) GOTO45
      IF(X.GE.XMIN) GOTO45
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    MODIFY LOWER SOLID REGION SURFACE CONTROL FUNCTION AS DEFINED     CC
CC    BY VALUE OF IOPTION                                               CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ANS=GMIN*(2-IOPTION)+(IOPTION-1)*AMIN1(ANS,CLIP)
   45 RETURN
40    CONTINUE
      ANS=0.0
      DO 50 K=1,NSYS
      Z(K)=(1-K)*(X-X0)
      RHOLD=0.0
      IF(Z(K).GT.-250.0) RHOLD=EXP(Z(K))
      ANS=ANS+COEF(K)*RHOLD
   50 CONTINUE
      IF(IOPTION.EQ.0) GOTO55
      IF(X.LE.XMIN) GOTO55
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    MODIFY UPPER SOLID REGION SURFACE CONTROL FUNCTION AS DEFINED     CC
CC    BY VALUE OF IOPTION                                               CC
```


Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ANS=GMIN*(2-IOPTION)+(IOPTION-1)*AMIN1(ANS,CLIP)
   55 RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION              CC
CC    ON LOWER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL REPORT       CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE AFC(R,ANS)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    USER SUPPLIED LOWER E'  TEMPERATURE A(R)                              CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL HFC(X0,ANS)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION              CC
CC    ON UPPER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL REPORT       CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE BFC(R,ANS)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    USER SUPPLIED UPPER END TEMPERATURE B(R)                              CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL HFC(XN,ANS)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    THIS SUBROUTINE FITS BESSEL SERIES TO DATA BY LEAST SQUARES           CC
CC    METHOD - SEE EQUATIONS (2.2.17) , (2.2.18) AND (2.2.23)               CC
CC    OF FINAL REPORT                                                       CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE COEFS(R,Y,NR,NCOEF,COEF)
      INTEGER NR,NCOEF
      REAL F,R(101),Y(101),COEF(20),WK(460)
      EXTERNAL F
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    USER SUPPLIED LEAST SQUARES METHOD FOLLOWS HERE TO DETERMINE          CC
CC    THE COEFFICIENTS OF EQUATIONS (2.2.17) AND (2.2.18). THE              CC
CC    SUBROUTINE IFLSQ BELOW IS THE IMSL LEAST SQUARES FUNCTION FIT         CC
CC    ROUTINE                                                               CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL IFLSQ(F,R,Y,NR,COEF,NCOEF,WK,IER)
      IF(IER.EQ.129.OR.IER.EQ.130)WRITE(6,10)
   10 FORMAT(58H TERMINAL ERROR IN LEAST SQUARES METHOD,SUBROUTINE COEFS
     1)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC    FUNCTION F USED IN SUBROUTINE COEFS F(N,R)=J0(LAMDA(N)*R)              CC
CC                                                                          CC
```

Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL FUNCTION F(N,R)
      INTEGER N
      REAL R,RLAMD(20)
      RLAMD( 1)=2.4048255577
      RLAMD( 2)=5.5200781103
      RLAMD(3)=8.6537279129
      RLAMD( 4)=11.7915344391
      RLAMD( 5)=14.9309177086
      RLAMD( 6)=18.0710639679
      RLAMD( 7)=21.2116366299
      RLAMD( 8)=24.3524715308
      RLAMD( 9)=27.4934791320
      RLAMD(10)=30.6346064684
      RLAMD(11)=33.7758202136
      RLAMD(12)=36.9170983537
      RLAMD(13)=40.0584257646
      RLAMD(14)=43.1997917132
      RLAMD(15)=46.3411883717
      RLAMD(16)=49.4826098974
      RLAMD(17)=52.6240518411
      RLAMD(18)=55.7655107550
      RLAMD(19)=58.9069839261
      RLAMD(20)=62.0484691902
      X=RLAMD(N)*R
      CALL J0(X,Y)
      F=Y
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    THIS SUBROUTINE COMPUTES THE J0 BESSEL FCN. Y=J0(X)             CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE J0(X,Y)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    USER SUPPLIED J0 FCN. PLACED HERE. THE IMSL ROUTINE MMBSJ0      CC
CC    IS ILLUSTRATED BELOW                                            CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL MMBSJ0
      Y=MMBSJ0(X,IER)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    SUBROUTINE FOR SOLVING A SYSTEM OF LINEAR SIMULTANEOUS          CC
CC    EQUATIONS HAVING A TRIDIAGONAL COEFFICIENT MATRIX. DIAGONALS    CC
CC    ARE STORED IN THE ARRAYS A, B, AND C. THE COMPUTED              CC
CC    SOLUTION VECTOR V(1) . . . V(L) IS STORED IN THE ARRAY V.       CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE TRIDAG(L)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    COMPUTE INTERMEDIATE ARRAYS BETA AND GAMMA                       CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      BETA(1)=B(1)
      GAMMA(1)=D(1)/BETA(1)
      IFP1=2
      DO 1 I=IFP1,L
        BETA(I)=B(I)-A(I)*C(I-1)/BETA(I-1)
        GAMMA(I)=(D(I)-A(I)*GAMMA(I-1))/BETA(I)
```


Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
    1  CONTINUE
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                      cc
cc     COMPUTE FINAL SOLN, VBECTOR V                                    cc
cc                                                                      cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
       V(L)=GAMMA(I)
       LAST=L-1
       DO 2 K=1,LAST
         I=L-K
         V(I)=GAMMA(I)-C(I)*V(I+1)/BETA(I)
    2  CONTINUE
       RETURN
       END
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                      cc
cc     THIS SUBROUTINE DETERMINES THE LOWER SOLID REGIONS SURFACE       cc
cc     CONTROL FUNCTION AS OUTLINED IN CHAPTER 3 OF FINAL REPORT,       cc
cc                                                                      cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
       SUBROUTINE SOLID2
       REAL J1,J1LAM,MMBSJ0
       COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
       COMMON/C5/R(101),PSI(20,101),SQJ1(20)
       COMMON/C9/COEF(20),RH
       COMMON/C22/TCASE,TMELT(3)
       COMMON/C23/GRAD2(101),GRAD3(101)
       COMMON/C24/RKS,RKL,RL,NSYS
       COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
       COMMON/C53/F(4)
       COMMON/C76/IFLAG2,IFLAG3
       COMMON/C77/AMAT(20,20),RHS(20)
       DIMENSION Y(101),C(4),IWK(20),WK(950)
       DIMENSION ALPHA(20)
       IF(IFLAG2.EQ.0) GOTO120
       WRITE(6,444)
  444  FORMAT(1H1,31X,60HL O W E R    S O L I D    T H E R M A L    G R A
     * D I E N T S)
       WRITE(6,333)
  333  FORMAT(///,1H ,52X,1HR,27X,4HGRAD)
       DO 30 JJ=1,101
       J=102-JJ
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                      cc
cc     DETERMINE LOWER SOLID REGION INTERFACE GRADIENT SEE EQUATION     cc
cc     (F24), FIGURE 1-2 OF FINAL REPORT                                cc
cc                                                                      cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
       GRAD2(J)=(RKL*GRAD2(J)-RL)/RKS
       WRITE(6,555)R(J),GRAD2(J)
  555  FORMAT(1H ,39X,E20.10,10X,E20.10)
       Y(J)=GRAD2(J)-GRAD2(101)
   30  CONTINUE
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                      cc
cc     DETERMINE COEFFICIENTS IN BESSEL EXPANSION (3.0.14)              cc
cc                                                                      cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
       CALL COEFS(R,Y,101,20,COEF)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc                                                                      cc
cc     DETERMINE MATRIX AND VECTOR ELEMENTS AS DEFINED IN               cc
cc     EQUATION (3.0.24)                                                cc
cc                                                                      cc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
       DO 80 M=1,MTERM
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
      L=M+2
         RHS(L)=RLAMD(M)*J1(M)*COEF(M)/2.0
 80   CONTINUE
      BOF1=GRAD2(101)
      AOF1=THELT(ICASE)
      DO 103 M=1,MTERM
      ALPHA(M)=P*P+4.0*RLAMD(M)*RLAMD(M)
      ALPHA(M)=(P-SQRT(ALPHA(M)))/2.0
      L=M+2
      RHS(L)=RHS(L)+(ALPHA(M)-P)*AOF1+BOF1
      RHS(L)=RHS(L)/(ALPHA(M)*(P-ALPHA(M)))
103   CONTINUE
      DO 106 M=1,MTERM
      DO 107 K=1,NSYS
      RKHOLD=K-1
      L=M+2
      AMAT(L,K)=1.0/(RKHOLD-ALPHA(M))
107   CONTINUE
106   CONTINUE
      DO 108 K=1,NSYS
      AMAT(1,K)=1.0
108   CONTINUE
      RHS(1)=AOF1
      DO 109 K=1,NSYS
      AMAT(2,K)=-(K-1)
109   CONTINUE
      RHS(2)=-BOF1
120   CONTINUE
      IFLAG2=0
      M=MTERM+2
      DO 111 I=1,4
      E(I)=0.0
111   CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    SOLVE THE OVER POSED LINEAR SYSTEM OF EQUATIONS (3.0.20) IN      CC
CC    THE LEAST SQUARES SENSE. THE IMSL ROUTINE LLBQF IS ILLUSTRATED  CC
CC    BELOW (SEE REMARKS AT THE END OF APPENDIX A.3)                   CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL LLBQF(AMAT,20,M,NSYS,RHS,20,1,0,E,COEF,20,IWK,WK,IER)
      RHOLD=0.0
      DO 113 K=2,NSYS
      RHOLD=RHOLD+COEF(K)
113   CONTINUE
      COEF(1)=AOF1-RHOLD
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    DISPLAY COEFFICIENTS USED IN THE EXPANSION OF THE LOWER          CC
CC    SOLID REGION SURFACE CONTROL FUNCTION (SEE EQUATION (3.0.23) )  CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,140)
140   FORMAT(1H1)
      WRITE(6,777)
  777 FORMAT(1H1,19X,83HL O W E R     S O L I D     S U R F A C E     C O N
     * T R O L     C O E F F I C I E N T S)
      WRITE(6,90)
   90 FORMAT(///,1H ,49X,1HK,22X,4HC(K))
      DO 778 I=1,NSYS
      WRITE(6,686) I, COEF(I)
  686 FORMAT(/,1H ,48X,I2,10X,E20.10)
  778 CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

```
CC                                                                      CC
CC    THIS SUBROUTINE DETERMINES THE UPPER SOLID REGIONS SURFACE        CC
CC    CONTROL FUNCTION AS OUTLINED IN CHAPTER 3 OF FINAL REPORT.        CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SOLID3
      REAL J1,J1LAM,MMBSJ0
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/C5/R(101),PSI(20,101),SQJ1(20)
      COMMON/C9/COEF(20),RM
      COMMON/C22/ICASE,TMELT(3)
      COMMON/C23/GRAD2(101),GRAD3(101)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/C53/E(4)
      COMMON/C76/IFLAG2,IFLAG3
      COMMON/C77/AMAT(20,20),RHS(20)
      DIMENSION Y(101),C(4),IWK(20),WK(950)
      DIMENSION ALPHA(20)
      IF(IFLAG3.EQ.0) GOTO120
      WRITE(6,444)
  444 FORMAT(1H1,31X,60HU P P E R    S O L I D    T H E R M A L    G R A
     * D I E N T S)
      WRITE(6,333)
  333 FORMAT(///,1H ,52X,1HR,27X,4HGRAD)
      DO 30 JJ=1,101
      J=102-JJ
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    DETERMINE UPPER SOLID REGION INTERFACE GRADIENT SEE EQUATION      CC
CC    (FZ2), FIGURE 1-2 OF FINAL REPORT                                 CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GRAD3(J)=(RKL*GRAD3(J)-RL)/RKS
      WRITE(6,555)R(J),GRAD3(J)
  555 FORMAT(1H ,39X,E20.10,10X,E20.10)
      Y(J)=GRAD3(J)-GRAD3(101)
   30 CONTINUE
      CALL COEFS(R,Y,101,20,COEF)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    DETERMINE MATRIX AND VECTOR ELEMENTS AS DEFINED IN EQUATIONS      CC
CC    (3.0.26) - (3.0.28)                                               CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 80 M=1,MTERM
      L=M+2
        RHS(L)=RLAMD(M)*J1(M)*COEF(M)/(-2.0)
   80 CONTINUE
      BOF1=GRAD3(101)
      AOF1=TMELT(ICASE)
      DO 103 M=1,MTERM
      ALPHA(M)=P*P+4.0*RLAMD(M)*RLAMD(M)
      ALPHA(M)=(P+SQRT(ALPHA(M)))/( 2.0)
      L=M+2
      RHS(L)=RHS(L)+(P-ALPHA(M))*AOF1-BOF1
      RHS(L)=RHS(L)/(ALPHA(M)*(P-ALPHA(M)))
  103 CONTINUE
      DO 106 M=1,MTERM
      DO 107 K=1,NSYS
      L=M+2
      AMAT(L,K)=1.0/(-1.0+K+ALPHA(M))
  107 CONTINUE
  106 CONTINUE
      DO 108 K=1,NSYS
      AMAT(1,K)=1.0
```

ORIGINAL PAGE IS OF POOR QUALITY

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
108 CONTINUE
      RHS(1)=AOF1
      DO 109 K=1,NSYS
      AMAT(2,K)=1-K
109 CONTINUE
      RHS(2)=BOF1
120 CONTINUE
      IFLAG3=0
      M=MTERM+2
      DO 111 I=1,4
      E(I)=0.0
111 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC    SOLVE THE OVER POSED LINEAR SYSTEM OF EQUATIONS (3.0.26) -          CC
CC    (3.0.28) FOR THE COEFFICIENTS TO BE USED IN (3.0.31). THE           CC
CC    IMSL ROUTINE LLBQF IS ILLUSTRATED BELOW                             CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL LLBQF(AMAT,20,M,NSYS,RHS,20,1,0,E,COEF,20,IWK,WK,IER)
      RHOLD=0.0
      DO 113 K=2,NSYS
      RHOLD=RHOLD+COEF(K)
113   CONTINUE
      COEF(1)=AOF1-RHOLD
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC    DISPLAY COEFFICIENTS USED IN THE EXPANSION OF THE UPPER SOLID       CC
CC    REGION SURFACE CONTROL FUNCTION (SEE EQUATION (3.0.31) )
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,140)
140   FORMAT(1H1)
      WRITE(6,777)
  777 FORMAT(1H1,19X,83HU P P E R    S O L I D    S U R F A C E    C O N
     * T R O L    C O E F F I C I E N T S)
      WRITE(6,90)
   90 FORMAT(///,1H ,49X,1HK,22X,4HC(K))
      DO 778 I=1,NSYS
      WRITE(6,686) I, COEF(I)
  686 FORMAT(/,1H ,48X,I2,10X,E20.10)
  778 CONTINUE
      GOTO180
180   CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC  PURPOSE                                                               CC
CC    - PERFORM LINE SEARCH TO DETERMINE MIN. PT. ON THE SURFACE          CC
CC    CONTROL FUNCTION. USED IF IOPTION =1 OR 2.                          CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE LINSRCH(XMIN)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      DIMENSION F78(105)
      A=0.0
      B=XN-X0
      STORE=-1.0
      DELB=B/200.0
      DO 90 I=1,200
      X=I*DELB
      CALL FUNC(X,Y)
      RHOLD=Y*STORE
      IF(RHOLD.LE.0.0)GO TO 100
90    CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
100   B=X
      ALPHA=0.01
      OFL=B-A
      FIB(1)=1.0
      FIB(2)=2.0
5     CONTINUE
      BB=1.0/ALPHA
      IF(BB-2.0)10,10,11
10    GO TO 14
11    CONTINUE
      JJ=2
12    JJ=JJ+1
      FIB(JJ)=FIB(JJ-1)+FIB(JJ-2)
      CC=FIB(JJ)
      IF(CC-BB)13,15,15
13    GO TO 12
14    WRITE(6,2)
2     FORMAT(///,10X,39HMUST CHANGE ALPHA IN SUBROUTINE LINSRCH)
15    I=0
      KK=JJ-2
      IK=JJ-2
      BL=B-A
      ALL=FIB(IK)*BL/FIB(JJ)
      W=A+ALL
      V=B-ALL
      CALL FUNC(W,T)
      CALL FUNC(V,U)
      JK=1
      IK=IK-1
      JJ=JJ-1
      DO 70 I=1,KK
      IF(U-T)20,20,22
20    A=A+ALL
      BL=B-A
      W=V
      CALL FUNC(W,T)
      ALL=FIB(IK)*BL/FIB(JJ)
      V=B-ALL
      CALL FUNC(V,U)
      II=I+1
      IK=IK-1
      JJ=JJ-1
      IF(IK-1)28,29,29
28    IK=1
29    CONTINUE
      GO TO 70
22    B=B-ALL
      BL=B-A
      V=W
      CALL FUNC(V,U)
      ALL =FIB(IK)*BL/FIB(JJ)
      W=A+ALL
      CALL FUNC(W,T)
      II=I+1
      IK=IK-1
      JJ=JJ-1
      IF(IK-1)30,31,31
30    IK=1
31    CONTINUE
   70 CONTINUE
      EPS=0.001*W
      DL=W+EPS
      CALL FUNC(DL,YL)
      IF(YL-T) 80,80,81
   80 CALL FUNC(B,BF)
      XMIN=(W+B)/2.0
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
      GOTO87
   81 CALL FUNC(A,AF)
      XMIN=(W+A)/2.0
    1 E10.4,2X,2HX=,E10.4)
   87 ACC=(W-A)/(DEL)
  999 CONTINUE
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC   PURPOSE                                                          CC
CC     - EVALUATE BASIS FUNCTIONS USED IN EQUATION (3.A.23)           CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE FUNC(X,Y)
      COMMON/C9/COEF(20),RH
      COMMON/C24/RKS,RKL,RL,NSYS
      Y=0.0
      DO 10 K=1,NSYS
      Z=(1-K)*X
      IF(Z.LE.-250.0)GO TO 10
      Y=Y+COEF(K)*EXP(Z)
   10 CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC   PURPOSE                                                          CC
CC     - EVALUATE L2 DIFFENENCE BETWEEN THE REQUIRED SOLID REGIONS    CC
CC     INTERFACE GRADIENTS AND THOSE OBTAINED BY USE OF THE SOLID     CC
CC     REGIONS SURFACE CONTROL FUNCTIONS.                             CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE ERROR
      COMMON/C22/ICASE,TMELT(3)
      COMMON/C23/GRAD2(101),GRAD3(101)
      COMMON/C21/THETAB(20,505),THOLD(101),T(10),GRADXN(101),GRADX0(10)
      IF(ICASE.EQ.3) GOTO50
      GXNL2=0.0
      GXNLINF=0.0
      DO 10 J=1,101
      GXNL2=GXNL2+GRADXN(J)*GRADXN(J)
      XMAG1=ABS(GRADXN(J))
      GXNLINF=AMAX1(XMAG1,GXNLINF)
   10 CONTINUE
      GXNL2=SQRT(GXNL2)
      E2NUM=0.0
      EINFNUM=0.0
      DO 20 K=1,101
      E2NUM=(GRADXN(K)-GRAD2(K))**2.0+E2NUM
      XMAG3=ABS(GRADXN(K)-GRAD2(K))
      EINFNUM=AMAX1(XMAG3,EINFNUM)
   20 CONTINUE
      E2NUM=SQRT(E2NUM)
      ERRL2=E2NUM/GXNL2
      ERRLINF=EINFNUM/GXNLINF
      GOTO80
   50 GX0L2=0.0
      GX0LINF=0.0
      DO 60 JJ=1,101
      GX0L2=GX0L2+GRADX0(JJ)*GRADX0(JJ)
      XMAG1=ABS(GRADX0(JJ))
      GX0LINF=AMAX1(XMAG1,GX0LINF)
   60 CONTINUE
      GX0L2=SQRT(GX0L2)
      E2NUM=0.0
      EINFNUM=0.0
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

```
      DO 70 KK=1,101
      E2NUM=(GRADX0(KK)-GRAD3(KK))**2.0+E2NUM
      XMAG3=ABS(GRADX0(KK)-GRAD3(KK))
      EINFNUM=AMAX1(XMAG3,EINFNUM)
   70 CONTINUE
      E2NUM=SQRT(E2NUM)
      ERRL2=E2NUM/GXOL2
      ERRLINF=EINFNUM/GXOLINF
      WRITE(6,669)
  669 FORMAT(///,1H ,46X,37HRELATIVE DIFFERENCES BETWEEN REQUIRED)
      WRITE(6,670)
  670 FORMAT(1H ,53X,23H AND OBTAINED GRADIENTS)
      WRITE(6,666)
  666 FORMAT(//1H ,58X,9HL-2 ERROR)
   80 WRITE(6,30)ERRL2
   30 FORMAT(1H ,46X,2(9X,F10.5))
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    THIS SUBROUTINE ESTIMATES LATERAL SURFACE TEMPERATURE            CC
CC    BY USE OF CUBIC SPLINE IF THE USER SUPPLIES A DISCRETE SET OF    CC
CC    LATERAL SURFACE TEMPERATURES.                                    CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SPLINE(XINT,YINT)
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      DIMENSION C(4,100)
      EQUIVALENCE(C1(1,1),C(1,1))
      IF(XINT-XD(1))2,1,2
    1 YINT=YD(1)
      RETURN
    2 K=1
    3 IF(XINT-XD(K+1))6,4,5
    4 YINT=YD(K+1)
      RETURN
    5 K=K+1
      IF((M-K).GT.0) GOTO3
      IF((M-K).LE.0) K=M-1
    6 YINT=(XD(K+1)-XINT)*(C(1,K)*(XD(K+1)-XINT)**2+C(3,K))
      YINT=YINT+(XINT-XD(K))*(C(2,K)*(XINT-XD(K))**2+C(4,K))
      RETURN
      END
      SUBROUTINE COFGEN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    FIND THE SPLINE CURVE FIT COEFFICIENTS, FOR USE IN CONJUNCTION   CC
CC    WITH SUBROUTINE SPLINE.                                          CC
CC    INPUTS -                                                         CC
CC    M    = NO. OF DATA PAIRS                                         CC
CC    XD   = ARRAY OF X (ABCISSA) VALUES                               CC
CC    YD   = ARRAY OF Y (ORDINATES) VALUES                             CC
CC    OUTPUTS -                                                        CC
CC    C    = 2-D ARRAY OF SPLINE FIT COEFFICIENTS (4 COEFFICIENTS      CC
CC           PER TRIPLET OF DATA POINTS).                              CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      DIMENSION C(4,100)
      DIMENSION P(100),E(100),A(100,3),B(100),Z(100),D(100)
      EQUIVALENCE(C1(1,1),C(1,1))
C
      ND=M
      M=M-1
       DO 2 K=1,M
       D(K)=XD(K+1)-XD(K)
```

ORIGINAL PAGE IS OF POOR QUALITY

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
      P(K)=D(K)/6.
    2 E(K)=(YD(K+1)-YD(K))/D(K)
      DO 3 K=2,M
    3 B(K)=E(K)-E(K-1)
      A(1,2)=-1.-D(1)/D(2)
      A(1,3)=D(1)/D(2)
      A(2,2)=2.*(P(1)+P(2))-P(1)*A(1,2)
      A(2,3)=(P(2)-P(1)*A(1,3))/A(2,2)
      B(2)=B(2)/A(2,2)
      DO 4 K=3,M
      A(K,2)=2.*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
      B(K)=B(K)-P(K-1)*B(K-1)
      A(K,3)=P(K)/A(K,2)
    4 B(K)=B(K)/A(K,2)
      Q=D(M-1)/D(M)
      A(ND,1)=1.+Q+A(M-1,3)
      A(ND,2)=-Q-A(ND,1)*A(M,3)
      B(ND)=B(M-1)-A(ND,1)*B(M)
      Z(ND)=B(ND)/A(ND,2)
      DO 6 I=1,ND-2
      K=ND-I
    6 Z(K)=B(K)-A(K,3)*Z(K+1)
      Z(1)=-A(1,2)*Z(2)-A(1,3)*Z(3)
      DO 7 K=1,M
      Q=1./(6.*D(K))
      C(1,K)=Z(K)*Q
      C(2,K)=Z(K+1)*Q
      C(3,K)=YD(K)/D(K)-Z(K)*P(K)
    7 C(4,K)=YD(K+1)/D(K)-Z(K+1)*P(K)
     M=M+1
      RETURN
C
C     END COFGEN
      END
```

Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

C.4 COMPUTER CODE LIST FOR PROBLEM P1-3

The comptuer code for Problem P1-3 is listed in Figure C-3. Before using this code, the user should review the remarks made at the end of Appendix A.4.

ORIGINAL PAGE IS
OF POOR QUALITY

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC  PROGRAM PURPOSE-                                                      CC
CC    COMPUTE THE MELT ZONE SURFACE CONTROL FUNCTION REQUIRED FOR         CC
CC    FLAT SOLID-MELT INTERFACES AS DESCRIBED IN PROBLEM P1-3 AND         CC
CC    SOLVED IN CHAPTER 4 OF THE FINAL REPORT (TO NASA) ENTITLED THE      CC
CC    CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED             CC
CC    BOUNDARY CONDITIONS- BY SCIENCE APPLICATIONS, INC.                  CC
CC  SOURCE-                                                               CC
CC    SCIENCE APPLICATIONS, INC.                                          CC
CC    HUNTSVILLE, ALABAMA                                                 CC
CC  AUT  RS-                                                              CC
CC    LA    M. FOSTER                                                     CC
CC    JOHN MCINTOSH                                                       CC
CC  REFERENCE-                                                            CC
CC    - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED       CC
CC    BOUNDARY CONDITIONS -                                               CC
CC    (FINAL REPORT - SAI-83/5034+HU)                                     CC
CC    SCIENCE APPLICATIONS                                                CC
CC  REMARKS-                                                              CC
CC    - SOFTWARE DEVELOPED AND TESTED ON CDC 7600/6400                    CC
CC    UNIVAC 1108                                                         CC
CC    - ALL EQUATIONS REFERENCED IN CODE BELOW ARE CONTAINED IN THE       CC
CC    FINAL REPORT-                                                       CC
CC            - THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF        CC
CC            SELECTED BOUNDARY CONDITIONS                                CC
CC  INPUT VARIABLES AND FUNCTIONS-                                        CC
CC    P              - PECLET NUMBER                                      CC
CC    MSUM           - NUMBER OF TERMS IN SERIES EXPANSION OF             CC
CC                   TEMPERATURE DISTRIBUTION (THE DESIRED SOLUTION)      CC
CC    BASIS          -USER PROVIDED FUNCTIONS TO EXPAND THE MELT ZONE     CC
CC                   SURFACE CONTROL FUNCTION- SEE EQ. (4.0.18) OF        CC
CC                   FINAL REPORT AND SUBROUTINE BASIS OF THIS CODE.      CC
CC    X0             - AXIAL POSITION OF LOWER END OF CYLINDER            CC
CC    XN             - AXIAL POSITION OF UPPER END OF CYLINDER            CC
CC    NGRID          - NUMBER OF GRID POINTS USED IN NUMERICAL            CC
CC                   SOLUTION OF O. D. E. BOUNDARY VALUE PROBLEM          CC
CC                   RESULTING FROM TRANSFORMATION OF PDE THE MODELING    CC
CC                   TEMPERATURE                                          CC
CC    NR             - NUMBER DIVISIONS OF CYLINDER RADIUS USED IN        CC
CC                   OUTPUT OF TEMPERATURE DISTRIBUTION                   CC
CC    RKS            - SOLID THERMAL CONDUCTIVITY                         CC
CC    RKL            - LIQUID THERMAL CONDUCTIVITY                        CC
CC    RL             - PRODUCT OF CRYSTAL GROWTH RATE, SOLID DENSITY,     CC
CC                   AND LATENT HEAT OF FUSION                            CC
CC    SLENGTH        - LENGTH OF SOLID REGIONS TO BE CONSIDERED           CC
CC    Q              - LENGTH OF MELT ZONE                                CC
CC    IHFC           - 1 IF A DISCRETE DATA POINT FORM OF THE SURFACE     CC
CC                   TEMPERATURE IS USER PROVIDED                         CC
CC                   0 IF A USER DEFINED FUNCTIONAL FORM OF THE           CC
CC                   SURFACE TEMPERATURE IS PROVIDED                      CC
CC    (XD,YD)        - USER PROVIDED DATA PTS FOR THE AXIAL DISTANCE      CC
CC                   (XD) AND CORRESPONDING SURFACE TEMPERATURE (YD)      CC
CC    M              - NUMBER OF DATA PTS. INPUT IF IHFC = 1              CC
```

Figure C-3. Computer Code List For
Problem P1-3

```
CC                    SET TO 0 IF IHFC = 0                                   CC
CC    HFC             - USER PROVIDED (IF IHFC = 0) SURFACE TEMPERATURE      CC
CC                    FUNCTION                                               CC
CC    CASE LIMITS     - THIS PROGRAM GENERATES THE SURFACE CONTROL           CC
CC                    FUNCTIONS FOR VARIOUS COMBINATIONS OF THE INDEX        CC
CC                    LIMITS MTERM AND NSYS (SEE EQ. (4.0.18) - (4.0.23)     CC
CC                    OF FINAL REPORT). TO DEFINE THESE COMBINATIONS         CC
CC                          MINTERM    - THE MINIMUM ALLOWED VALUE           CC
CC                                     OF MTERM                              CC
CC                          MAXTERM    - THE MAXIMUM ALLOWED VALUE           CC
CC                                     OF MTERM                              CC
CC                          DELTERM    - INCREMENT OF MTERM FROM MINTERM     CC
CC                                     TO MAXTERM                            CC
CC                          MINNSYS    - THE MINIMUM ALLOWED VALUE OF NSYS   CC
CC                          MAXNSYS    - THE MAXIMUM ALLOWED VALUE OF NSYS   CC
CC                          DELTERM    - INCREMENT OF NSYS FROM MINNSYS      CC
CC                                     TO MAXNSYS                            CC
CC  OUTPUT VARIABLES-                                                        CC
CC    THOLD           - TEMPERATURE DISTRIBUTION ARRAY FOR EACH REGION       CC
CC                    - SEE SUBROUTINE MELT OF CODE                          CC
CC    GRADX0          - AXIAL THERMAL GRADIENT AT X0 FOR REGIONS.            CC
CC                    (X0 IS SET TO Q FOR UPPER SOLID REGION, AND TO         CC
CC                    NEGATIVE SLENGTH FOR LOWER SOLID REGION)               CC
CC    GRADXN          - AXIAL THERMAL GRADIENT AT XN FOR REGIONS.            CC
CC                    (XN IS SET TO SLENGTH + Q FOR UPPER SOLID REGION,      CC
CC                    AND TO 0 FOR LOWER SOLID REGION)                       CC
CC    GRADAT0         - AXIAL THERMAL GRADIENT AT BOTTOM OF MELT ZONE        CC
CC                    (SEE MAIN OF CODE)                                     CC
CC    GRADATQ         - AXIAL THERMAL GRADIENT AT TOP OF MELT ZONE           CC
CC                    (SEE MAIN OF CODE)                                     CC
CC    CPOLY           - ARRAY OF COEFFICIENTS USED TO EXPAND THE MELT        CC
CC                    SURFACE CONTROL FUNCTION (SEE EQ. (4.0.18) OF          CC
CC                    FINAL REPORT AND SUBROUTINES MELT1 AND HFC OF          CC
CC                    CODE)                                                  CC
CC    ERRL20          - THE RELATIVE L2 DIFFERENCE BETWEEN THE               CC
CC                    DESIRED GRADIENT AT X=X0 AND THE GRADIENT              CC
CC                    OBTAINED BY USING THE SURFACE CONTROL FUNCTION.        CC
CC    ERRL2Q          - THE RELATIVE L2 DIFFERENCE BETWEEN THE               CC
CC                    DESIRED GRADIENT AT X=XN AND THE GRADIENT              CC
CC                    OBTAINED BY USING THE SURFACE CONTROL FUNCTION.        CC
CC  USER SUPPLIED MATHMATICAL SOFTWARE-                                      CC
CC    - A LEAST SQUARES ALGORITHM TO FIT A FUNCTION TO A LINEAR              CC
CC    COMBINATION OF SELECTED FUNCTIONS                                      CC
CC    - AN ALGORITHM TO EVALUATE BESSEL FUNCTIONS (REQUIRED IN               CC
CC    SUBROUTINE J0)                                                         CC
CC    - A NUMERICAL INTEGRATION ROUTINE (REQUIRED IN SUBROUTINES             CC
CC    INTEGL1 AND INTEGL2)                                                   CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C     MAIN***DRIVER
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/C21/THETAB(20,505),THOLD(101),T(10),GRADXN(101),GRADX0(101)
      COMMON/C22/TCASF,THELT(3)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/C25/GRADAT0(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(44,20),RHS(44),WWK(1500),IWK(20)
      COMMON/C40/SLENGTH
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      DIMENSION R(101)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                           CC
CC    DETERMINE TEMPERATURE DISTRIBUTION AND GRADIENTS FOR UPPER             CC
CC    SOLID REGION                                                           CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Figure C-3. Computer Code List For Problem P1-3 (Cont)

```
      ICASE=3
      WRITE(6,10)
   10 FORMAT(1H1,48X,22HU P P E R    S O L I D)
      CALL INPUT
      XN=Q+SLENGTH
      X0=Q
      CALL MELT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    COMPUTE GRADIENT IN MELT ZONE AT UPPER INTERFACE (SEE EQUATION  CC
CC    FIG. 1-2) AND STORE RESULT IN GRADATQ                           CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 20 I=1,101
      GRADATQ(I)=(RKS*GRADX0(I)+RL)/RKL
   20 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    DETERMINE TEMPERATURE DISTRIBUTION AND GRADIENT FOR LOWER       CC
CC    SOLID REGION                                                    CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ICASE=2
      WRITE(6,30)
   30 FORMAT(1H1,48X,22HL O W E R    S O L I D)
      CALL INPUT
      XN=0
      X0=-SLENGTH
      CALL MELT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    COMPUTE GRADIENT IN MELT ZONE AT LOWER INTERFACE (SEE EQUATION  CC
CC    FZ4, FIG. 1-2) AND STORE RESULT IN GRADATO                      CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 40 I=1,101
      GRADAT0(I)=(RKS*GRADXN(I)+RL)/RKL
   40 CONTINUE
C
C
      WRITE(6,400)
  400 FORMAT(1H1,38X,56HM E L T    Z O N E    T H E R M A L    G R A D I
     * E N T S)
  401 FORMAT(/,1H ,50X,20HA T    I N T E R F A C E)
      WRITE(6,401)
      WRITE(6,72)
   72 FORMAT(///,44X,1HR,7X,15HGRAD. AT X= 0.0,12X,13HGRAD. AT X= Q)
      WRITE(6,402)
  402 FORMAT(/,1H )
      DO 947 KK=1,101
      R(KK)=(KK-1)*0.01
      WRITE(6,300)R(KK),GRADX0(KK),GRADXN(KK)
  300  FORMAT(1H ,39X,F8.6,3X,E17.9,7X,E17.9)
  947 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    DETERMINE MELT ZONE CONTROL FUNCTION (EQUATION 4.0.18) AND      CC
CC    RESULTING TEMPERATURE DISTRIBUTION                              CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ICASE=1
      WRITE(6,983)
  983 FORMAT(1H1,50X,18HM E L T    Z O N E)
      CALL INPUT
      CALL MELT1
      STOP
```


Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                          CC
CC  PURPOSE                                                                 CC
CC    - DETERMINE MELT ZONE TEMPERATURE DISTRIBUTION                        CC
CC    - COMPUTE RELATIVE ERRORS BETWEEN GRADIENTS REQUIRED AT THE           CC
CC      MELT ZONE INTERFACES AND THOSE OBTAINED USING THE SURFACE           CC
CC      CONTROL FUNCTION                                                    CC
CC                                                                          CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE FOSTER
      COMMON/RFAC1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/C21/THETAR(20,505),THCLD(101),T(10),GRADXN(101),GRADX0(101)
      COMMON/C22/ICASE,THELT(3)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/C25/GRADATO(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(44,20),RHS(44),WNW(1500),IIWK(20)
      COMMON/C40/SLENGTH
      X0=0
      XN=0
      CALL MELT
C
      GATOL2=0.0
      GATOLIN=0.0
      GATQL2=0.0
      GATQLIN=0.0
      DO 50 J=1,101
      GATOL2=GATOL2+GRADATO(J)*GRADATO(J)
      GATQL2=GATQL2+GRADATQ(J)*GRADATQ(J)
      XMAG1=ABS(GRADATO(J))
      XMAG2=ABS(GRADATQ(J))
      GATOLIN=AMAX1(XMAG1,GATOLIN)
      GATQLIN=AMAX1(XMAG2,GATQLIN)
   50 CONTINUE
      GATOL2=SQRT(GATOL2)
      GATQL2=SQRT(GATQL2)
      E2NUMO=0.0
      E2NUMQ=0.0
      ENFNO=0.0
      ENFNQ=0.0
      DO 60 K=1,101
      E2NUMO=(GRADATO(K)-GRADX0(K))**2.0+E2NUMO
      E2NUMQ=(GRADATQ(K)-GRADXN(K))**2.0+E2NUMQ
      XMAG3=ABS(GRADATO(K)-GRADX0(K))
      XMAG4=ABS(GRADATQ(K)-GRADXN(K))
      ENFNO=AMAX1(XMAG3,ENFNO)
      ENFNQ=AMAX1(XMAG4,ENFNQ)
   60 CONTINUE
      E2NUMO=SQRT(E2NUMO)
      E2NUMQ=SQRT(E2NUMQ)
      ERLINFO=ENFNO/GATOLIN
      ERLINFQ=ENFNQ/GATQLIN
      ERRL2O=E2NUMO/GATOL2
      ERRL2Q=E2NUMQ/GATQL2
      WRITE(6,669)
  669 FORMAT(///,1H ,46X,37HRELATIVE DIFFERENCES BETWEEN REQUIRED)
      WRITE(6,670)
  670 FORMAT(1H ,53X,23H AND OBTAINED GRADIENTS)
      WRITE(6,666)
  666 FORMAT(//,1H ,58X,9HL-2 ERROR)
      WRITE(6,667) ERRL2O
  667 FORMAT(1H ,42X,4HAT 0.2(9X,F10.5))
      WRITE(6,668)ERRL2Q
  668 FORMAT(1H ,42X,4HAT Q.2(9X,F10.5))
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
      SUBROUTINE MELT
      REAL J1,J1LAM,MMBSJ0
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/C10/ASCRIP(20),BSCRIP(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/C5/R(101),PSI(20,101),SQTI(20)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
      COMMON/C21/THETAB(20,505),THOLD(101),T(10),GRADXN(101),GRADX0(101)
      COMMON/C22/TCASE,TMELT(3)
      COMMON/C23/GRAD2(101),GRAD3(101)
      CHARACTER*17 RIS,ALPHA(6)
      CHARACTER*18 STARS,STAR(6)
      DATA RIS/'R=                '/,STARS/'******************'/
      DO 207 L=1,6
      ALPHA(L)=RIS
      STAR(L)=STARS
  207 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    RLAMD(M)=ROOT OF J0 BESSEL FCN                                 CC
CC    J1(M)=J1(RLAMD(M)) WHERE J1 IS BESSEL FCN                      CC
CC    J1LAM(M)=J1(M)/RLAMD(M)                                        CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RLAMD( 1)=2.4048255577
      RLAMD( 2)=5.5200781103
      RLAMD( 3)=8.6537279129
      RLAMD( 4)=11.7915344391
      RLAMD( 5)=14.9309177086
      RLAMD( 6)=18.0710639679
      RLAMD( 7)=21.2116366299
      RLAMD( 8)=24.3524715308
      RLAMD( 9)=27.4934791320
      RLAMD(10)=30.6346064684
      RLAMD(11)=33.7758202136
      RLAMD(12)=36.9170983537
      RLAMD(13)=40.0584257646
      RLAMD(14)=43.1997917132
      RLAMD(15)=46.3411883717
      RLAMD(16)=49.4826098974
      RLAMD(17)=52.6240518411
      RLAMD(18)=55.7655107550
      RLAMD(19)=58.9069839261
      RLAMD(20)=62.0484691902
      J1( 1)=0.5191474973
      J1( 2)=-0.3402648065
      J1( 3)=0.2714522999
      J1( 4)=-0.2324598314
      J1( 5)=0.2065464331
      J1( 6)=-0.1877288030
      J1( 7)=0.1732658942
      J1( 8)=-0.1617015507
      J1( 9)=0.1521812138
      J1(10)=-0.1441659777
      J1(11)=0.1372969434
      J1(12)=-0.1313246267
      J1(13)=0.1260694971
      J1(14)=-0.1213986248
      J1(15)=0.1172111989
      J1(16)=-0.1134291926
      J1(17)=0.1099911430
      J1(18)=-0.1068478883
      J1(19)=0.1039595729
      J1(20)=-0.1012934989
      DO 566 I=1,20
        J1LAM(I)=J1(I)/RLAMD(I)
```


Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
      SQJ1(I)=J1(I)*J1(I)
  566 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    FIND COEFS FOR BESSEL EXPANSIONS OF A(R)-A(1) AND B(R)-B(1)      CC
CC    SEE EQUATIONS (2.2.17) AND (2.2.18) OF FINAL REPORT              CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL AFC(1.0,AOF1)
      CALL BFC(1.0,BOF1)
      DO 20 I=1,101
        R(I)=(I-1)*0.01
        RHOLD=R(I)
        CALL AFC(RHOLD,ANS)
        A(I)=ANS-AOF1
        CALL BFC(RHOLD,ANS)
        B(I)=ANS-BOF1
   20 CONTINUE
      CALL COEFS(R,A,101,20,ASCRIP)
      CALL COEFS(R,B,101,20,BSCRIP)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    SOLVE FOR THETA BAR OF EQUATIONS (2.2.19) BY SOLVING THE         CC
CC    TRIDIAGONAL SYSTEM (2.2.20) - SEE FINAL REPORT                   CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DX=(XN-X0)/NGRID
      DX2=DX*DX
      L=NGRID-1
      DO 565 M=1,MSUM
        DO 40 I=1,L
          A(I)=1.0+DX*P/2.0
          B(I)=-2.0-DX2*RLAMD(M)*RLAMD(M)
          C(I)=1.0-DX*P/2.0
      X=X0+I*DX
          CALL GBAR(M,X,ANS)
          D(I)=DX2*ANS
   40   CONTINUE
        D(1)=D(1)-(1.0+DX*P/2.0)*ASCRIP(M)*SQJ1(M)*0.5
        D(L)=D(L)-(1.0-DX*P/2.0)*BSCRIP(M)*SQJ1(M)*0.5
      CALL TRIDAG(L)
        DO 50 I=2,NGRID
          II=I-1
          THETAB(M,I)=V(II)
   50   CONTINUE
        NSTOP=NGRID+1
        THETAB(M,1)=ASCRIP(M)*SQJ1(M)/2.0
        THETAB(M,NSTOP)=BSCRIP(M)*SQJ1(M)/2.0
  565 CONTINUE
      DR=1.0/NR
      NRSTOP=NR+1
      DO 60 I=1,NRSTOP
        R(I)=(I-1)*DR
        DO 65 M=1,MSUM
      PSI(M,I)=F(M,R(I))
65      CONTINUE
60    CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC    PRINT TEMPERATURES                                               CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(ICASE.EQ.3) GOTO678
      WRITE(6,30)
   30 FORMAT(1H1,52X,22HL O W E R     S O L I D)
      GOTO679
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
  678 CONTINUE
      WRITE(6,10)
   10 FORMAT(1H1,52X,22HU P P E R     S O L I D)
  679 CONTINUE
      WRITE(6,70)
   70 FORMAT(1H ,45X,49H T E M P E R A T U R E     D I S T R I B U T I O
     1N)
      IFLAG=0
      MRIGHT=6
      MLEFT=1
  180 CONTINUE
      IF(NRSTOP.LE.MRIGHT)IFLAG=1
      MRIGHT=MINO(NRSTOP,MRIGHT)
      WRITE(6,190)(R(J),J=MLEFT,MRIGHT)
  190 FORMAT(/////,1H ,17X,6(F12.8,5X))
      WRITE(6,267) (ALPHA(L),L=1,MRIGHT)
  267 FORMAT(1H+,17X,6A17)
      WRITE(6,268) (STAR(L),L=1,MRIGHT)
  268 FORMAT(1H0,15X,6A17)
      DO 200 I=1,NSTOP
      ISKIP=I-1
      IHOLD=(ISKIP+0.00000001)/10.0
      XHOLD=(ISKIP/10.0)-IHOLD
      IF(XHOLD.GT.0.005) GOTO200
        II=NSTOP+1-I
      X=X0+(II-1)*DX
        DO 202 J=MLEFT,MRIGHT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    DETERMINE TEMPERATURE AT (X,R(J))                               CC
CC    SEE EQUATION (2.2.14) OF FINAL REPORT                           CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          THOLD(J)=0.0
          DO 204 M=1,MSUM
     -      THOLD(J)=THOLD(J)+2.0*PSI(M,J)*THETAB(M,II)/SQJ1(M)
  204     CONTINUE
        CALL HFC(X,ANS)
          THOLD(J)=THOLD(J)+ANS
  202     CONTINUE
          WRITE(6,210)X,(THOLD(J),J=MLEFT,MRIGHT)
  210 FORMAT(3H X=,F10.6,5H    * ,6(E15.8,2X))
  200   CONTINUE
        IF(IFLAG.EQ.1)GO TO 220
        MRIGHT=MRIGHT+6
        MLEFT=MLEFT+6
        GO TO 180
  220   CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    COMPUTE THERMAL GRADIENTS AT X=X0 AND XN                        CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  395 CONTINUE
      IF(ICASE.EQ.3) GOTO681
      WRITE(6,30)
      GOTO682
  681 CONTINUE
      WRITE(6,10)
  682 CONTINUE
      WRITE(6,71)
   71 FORMAT(1H ,47X,35H T H E R M A L     G R A D I E N T S)
      WRITE(6,72)X0,XN
   72 FORMAT(///,44X,1HR,5X,11HGRAD. AT X=,F10.5,14H    GRAD. AT X=,F10.5
     2,//)
      DO 230 I=1,101
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
      R(I)=(I-1)*0.01
      DO 240 M=1,MSUM
        VAR=R(I)*RLAMD(M)
        CALL J0(VAR,Y)
        PSI(M,I)=Y
240   CONTINUE
230   CONTINUE
      DD=-DX
      DO 250 I=1,101
        DO 260 J=1,5
          T(J)=0.0
          DO 270 M=1,MSUM
            T(J)=T(J)+2.0*PSI(M,I)*THETAB(M,J)/SQJ1(M)
270       CONTINUE
          X=X0+(J-1)*DX
          CALL HFC(X,ANS)
          T(J)=T(J)+ANS
260     CONTINUE
        DO 280 J=6,10
          T(J)=0.0
          JHOLD=NSTOP-10+J
            DO 290 M=1,MSUM
            T(J)=T(J)+2.0*PSI(M,I)*THETAB(M,JHOLD)/SQJ1(M)
290       CONTINUE
        X=X0+(JHOLD-1)*DX
        CALL HFC(X,ANS)
        T(J)=T(J)+ANS
280   CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC    APPROXIMATE THERMAL GRADIENTS AT CYLINDER ENDS                      CC
CC    - SEE EQUATIONS (2.2.21) AND (2.2.22) OF FINAL REPORT               CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GRADXN(I)=(-3*T(6)+16*T(7)-36*T(8)+48*T(9)-25*T(10))/(12*DD)
      GRADX0(I)=(-3*T(5)+16*T(4)-36*T(3)+48*T(2)-25*T( 1))/(12*DX)
      ISKIP=I-1
      IHOLD=(ISKIP+0.00000001)/10.0
      XHOLD=(ISKIP/10.0)-IHOLD
      IF(XHOLD.GT.0.005) GOTO250
      WRITE(6,300)R(I),GRADX0(I),GRADXN(I)
  300 FORMAT(1H ,39X,F8.6,3X,E17.9,7X,E17.9)
250   CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                        CC
CC    THIS SUBROUTINE APPROXIMATES (BY FINITE DIFFERENCE) G BAR OF        CC
CC    EQUATION (2.2.16) OF FINAL REPORT                                   CC
CC                                                                        CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE GBAR(M,X,ANS)
      REAL J1,J1LAM
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      EPSLON=0.01
      X1=X-EPSLON
      X2=X+EPSLON
      CALL HFC(X,ANS)
      CALL HFC(X1,ANS1)
      CALL HFC(X2,ANS2)
      G=P*(ANS2-ANS1)/(2.0*EPSLON)
      G=G-(ANS2+ANS1-2.0*ANS)/(EPSLON*EPSLON)
      ANS=G*J1LAM(M)
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC  PURPOSE -                                                         CC
CC     SUPPLY INPUT DATA, SEE APPENDIX A.4 FOR DETAILS                CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE INPUT
      INTEGER DELTERM,DELNSYS
      COMMON/PFAD1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/C22/ICASE,THELT(3)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/C25/GRADATO(101),GRADATG(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(40,20),RHS(44),WWK(1500),IWK(20)
      COMMON/C31/THEC
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      COMMON/C40/SLENGTH
      COMMON/C51/MAXTERM,MINTERM,MAXNSYS,MINNSYS,DELTERM,DELNSYS
      WRITE(6,5)
    5 FORMAT(//,1H ,50X,20HI N P U T    D A T A)
      GO TO(10,20,20),ICASE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC     MELT PARAMETERS                                                CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   10 READ(5,12)P,MSUM,NGRID,NR
   12 FORMAT(E20.10,6I10)
      WRITE(6,25)
   25 FORMAT(1H ,48H          P                 NGRID      NR       MSUM)
      WRITE(6,14) P, NGRID,NR,MSUM
   14 FORMAT(F20.10,3I10)
      READ(5,16)MAXTERM,MINTERM,MAXNSYS,MINNSYS,DELTERM,DELNSYS
   16 FORMAT(10I5)
      WRITE(6,799)
  799 FORMAT(//,1H ,9X,57HMAXTERM   MINTERM   DELTERM    MAXNSYS    MINNSY
     -S    DELNSYS)
      WRITE(6,18)MAXTERM,MINTERM,DELTERM,MAXNSYS,MINNSYS,DELNSYS
   18 FORMAT(1H ,5X,6(5X,I5))
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC     LOWER SOLID/UPPER SOLID PARAMETERS                             CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   20 CONTINUE
      READ(5,12)P,MSUM,NGRID,NR
      WRITE(6,25)
      WRITE(6,14)P,NGRID,NR,MSUM
      READ(5,22)RKS,RKL,RL,SLENGTH
   22 FORMAT(4E20.10)
      IF(ICASE.NE.3) GOTO26
      READ(5,22)Q
      WRITE(6,888)
  888 FORMAT(//,1H ,10X,3HRKS,17X,3HRKL,17X,2HRL,20X,7HSLENGTH,8X,11HMEL
     -T LENGTH)
      WRITE(6,24)RKS,RKL,RL,SLENGTH,Q
   24 FORMAT(5E20.10)
   26 CONTINUE
      IF(ICASE.EQ.3) GOTO48
      WRITE(6,888)
      WRITE(6,24)RKS,RKS,RL,SLENGTH,Q
   48 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC     SPLINE INPUT OPTION                                            CC
CC                                                                    CC
```


Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      READ(5,16)IHFC,M
      WRITE(6,30)M
   30 FORMAT(//////,95H THE SURFACE TEMPERATURE DISTRIBUTION IS APPROXIM
     2ATED BY THE CUBIC SPLINE THROUGH THE FOLLOWING,I4,23H  (X, TEMP) D
     *ATA POINTS,////,37H           X              SURFACE TEMP,)
      IF(IHFC.EQ.0) RETURN
      DO 32 I=1,M
      READ(5,22)XD(I),YD(I)
      WRITE(6,34)XD(I),YD(I)
   34 FORMAT(2E20.10)
   32 CONTINUE
      CALL COFGEN
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    THIS SUBROUTINE PROVIDES FOR DATA INPUT                        CC
CC    THIS SUBROUTINE SUPPLIES LATERAL SURFACE TEMPERATURE           CC
CC    - SEE EQUATION (2.2.4) OF FINAL REPORT                         CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE HFC(X,ANS)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/C31/IHFC
      COMMON/C26/CPOLY(20)
      COMMON/C22/ICASE,TMELT(3)
      GO TO(10,20,30),ICASE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    MELT SURFACE CONTROL TEMPERATURE, SEE EQUATION 2.0.18          CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10    ANS=0.0
      DO 12 K=1,NSYS
      CALL BASIS(K,X,ANS1)
      ANS=ANS+CPOLY(K)*ANS1
12    CONTINUE
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    LOWER SOLID SURFACE TEMPERATURE COMPUTED NEXT                  CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   20 IF(IHFC.EQ.1) GOTO22
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    USER SUPPLIED LOWER SOLID SURFACE TEMP. DISTRIBUTION           CC
CC    PLACED HERE                                                    CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RETURN
   22 CALL SPLINE(X,ANS)
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
CC    UPPER SOLID SURFACE  EMPERATURE COMPUTED NEXT                  CC
CC                                                                   CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   30 IF(IHFC.EQ.1) GOTO22
CC    USER SUPPLIED UPPER SOLID SURFACE TEMP PLACED HERE IF IN       CC
CC    FUNCTIONAL FORM                                                CC
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                   CC
```

ORIGINAL PAGE IS OF POOR QUALITY

Figure C-3. Computer Code List For Problem P1-3 (Cont)

```
CC     THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION          CC
CC     ON LOWER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL          CC
CC     REPORT                                                            CC
CC                                                                       CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE AFC(R,ANS)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      CALL HFC(X0,ANS)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                       CC
CC     THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION          CC
CC     ON UPPER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL REPORT  CC
CC                                                                       CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE BFC(R,ANS)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      CALL HFC(XN,ANS)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                       CC
CC     THIS SUBROUTINE FITS BESSEL SERIES TO DATA BY LEAST SQUARES      CC
CC     METHOD - SEE EQUATIONS (2.2.17) , (2.2.18) AND (2.2.23)          CC
CC     OF FINAL REPORT                                                   CC
CC                                                                       CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE COEFS(R,Y,NR,NCOFF,COEF)
      INTEGER NR,NCOEF
      REAL F,R(101),Y(101),COEF(20),WK(460)
      EXTERNAL F
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                       CC
CC     USER SUPPLIED LEAST SQUARES METHOD FOLLOWS HERE TO DETERMINE     CC
CC     THE COEFFICIENTS OF EQUATIONS (2.2.17) AND 2.2.18). THE          CC
CC     SUBROUTINE IFLSQ BELOW IS THE IMSL LEAST SQUARES FUNCTION       CC
CC     FIT ROUTINE                                                       CC
CC                                                                       CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL IFLSQ(F,R,Y,NR,COEF,NCOEF,WK,IER)
      IF(IER.EQ.129.OR.IER.EQ.130)WRITE(6,10)
   10 FORMAT(56H TERMINAL ERROR IN LEAST SQUARES METHOD,SUBROUTINE COEFS
     1)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                       CC
CC     THIS FUNCTION EVALUATES THE ZERO ORDER BESSEL FUNCTION           CC
CC     DENOTED IN NOTATION N2=1 (III)) - SEE FINAL REPORT               CC
CC                                                                       CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL FUNCTION F(N,R)
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      X=RLAMD(N)*R
      CALL J0(X,Y)
      F=Y
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                       CC
CC     THIS SUBROUTINE COMPUTES THE J0 BESSEL FUNCTION Y=J0(X)         CC
CC                                                                       CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE J0(X,Y)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```


Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
CC                                                                      CC
CC     USER SUPPLIED J0 FUNCTION PLACED HERE, IN THIS EXAMPLE, THE      CC
CC     IMSL BESSEL FUNCTION MMBSJ0 IS ILLUSTRATED                       CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL MMBSJ0
      Y=MMBSJ0(X,IER)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC     SUBROUTINE FOR SOLVING A SYSTEM OF LINEAR SIMULTANEOUS           CC
CC     EQUATIONS HAVING A TRIDIAGONAL COEFFICIENT MATRIX, DIAGONALS     CC
CC     ARE STORED IN THE ARRAYS A, B, AND C, THE COMPUTED               CC
CC     SOLUTION VECTOR V(1) . . . V(L) IS STORED IN THE ARRAY V,        CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE TRIDAG(L)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC     COMPUTE INTERMEDIATE ARRAYS BETA AND GAMMA                       CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      BETA(1)=B(1)
      GAMMA(1)=D(1)/BETA(1)
      IFP1=2
      DO 1 I=IFP1,L
        BETA(I)=B(I)-A(I)*C(I-1)/BETA(I-1)
        GAMMA(I)=(D(I)-A(I)*GAMMA(I-1))/BETA(I)
    1 CONTINUE
C
C       COMPUTE FINAL SOLN, VECTOR V
      V(L)=GAMMA(L)
      LAST=L-1
      DO 2 K=1,LAST
        I=L-K
        V(I)=GAMMA(I)-C(I)*V(I+1)/BETA(I)
    2 CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC   PURPOSE -                                                          CC
CC     1, GENERATE MELT ZONE SURFACE CONTROL FUNCTION                   CC
CC     2. GENERATE THERMAL DISTRIBUTION IN MELT ZONE                    CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE MELT1
      INTEGER DELTERM,DELNSYS
      REAL J1,J1LAM,MMBSJ0
      COMMON/C1/RLAMD(20),J1(20),J1LAM(20)
      COMMON/C5/R(101),PSI(20,101),SQJ1(20)
      COMMON/C9/COEF(20),RH
      COMMON/C10/ASCRIP(20),BSCRIP(20)
      COMMON/C20/A(500),B(500),C(500),D(500),V(500),BETA(505),GAMMA(505)
      COMMON/C21/THETAB(20,505),THCLD(101),T(10),GRADXN(101),GRADX0(101)
      COMMON/C22/ICASE,TMELT(3)
      COMMON/C23/GRAD2(101),GRAD3(101)
      COMMON/C24/RKS,RKL,RL,NSYS
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      COMMON/FIXPT/IOPTION,XMIN,GMIN,CLIP
      COMMON/C25/GRADAT0(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(40,20),RHS(40),WNK(1500),IINK(20)
      COMMON/C26/CPOLY(20)
      COMMON/C27/KERNEL,NKERNEL,KKERNEL
```


Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
      COMMON/C50/F(4),AL6(44,20),RH6(44)
      COMMON/C51/MAXTERM,MINTERM,MAXNSYS,MINNSYS,DELTERM,DELNSYS
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                           CC
CC    GENERATE BESSEL EXPANSION COEFFICIENTS OF EQUATIONS (4.07) AND         CC
CC    (4.0.8) OF FINAL REPORT                                                CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 20 I=1,101
      A(I)=GRADATO(I)-GRADATO(101)
      B(I)=GRADATQ(I)-GRADATQ(101)
   20 CONTINUE
      CALL COEFS(R,A,101,20,ASCRIP)
      CALL COEFS(R,B,101,20,BSCRIP)
      MTERM=MAXTERM
      NSYS=MAXNSYS
      DO 30 I=1,MTERM
      II=I+4
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                           CC
CC    GENERATE RIGHT HAND SIDES OF EQUATION (4.0.23) OF FINAL REPORT         CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RHS(II)=-0.5*ASCRIP(I)*J1(I)*RLAMD(I)-GRADATO(101)
      II=II+MTERM
      RHS(II)= 0.5*BSCRIP(I)*J1(I)*RLAMD(I)+GRADATQ(101)
      S(I)=P*P+4.0*RLAMD(I)*RLAMD(I)
      S(I)=SQRT(S(I))
   30 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                           CC
CC    GENERATE RIGHT HAND SIDES OF EQUATIONS (4.0.19) - (4.0.22) OF          CC
CC    FINAL REPORT                                                           CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RHS(1)=0.0
      RHS(2)=0.0
      RHS(3)=GRADATO(101)
      RHS(4)=GRADATQ(101)
      DO 40 N=1,MTERM
      DO 50 K=1,NSYS
      NN=N+4
      CALL INTEGL1(N,K,ANS)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                           CC
CC    GENERATE COEFFICIENTS IN LEFT HAND SIDE OF EQUATION (4.0.23)           CC
CC    OF FINAL REPORT                                                        CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      AL2(NN,K)=ANS
      CALL INTEGL2(N,K,ANS)
      NN=NN+MTERM
      AL2(NN,K)=ANS
   50 CONTINUE
   40 CONTINUE
      DO 60 K=1,NSYS
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                           CC
CC    GENERATE COEFFICIENTS IN LEFT HAND SIDE OF                             CC
CC    EQUATIONS (4.0.19) - (4.0.22)                                          CC
CC                                                                           CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL BASIS(K,0,ANS)
      AL2(1,K)=ANS
      CALL BASIS(K,0,ANS)
      AL2(2,K)=ANS
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-3. Computer Code List For
Problem P1-3 (Cont)

```
      CALL DBASIS(K,0,ANS)
      AL2(3,K)=ANS
      CALL DBASIS(K,G,ANS)
      AL2(4,K)=ANS
 60   CONTINUE
      DO 510 NSYS=MINNSYS,MAXNSYS,DELNSYS
      DO 500 MTERM=MINTERM,MAXTERM,DELTERM
      NN=4+2*MTERM
      IF(NN.LE.NSYS) GOTO500
      DO 540 I=1,44
      RH6(I)=RHS(I)
      DO 550 J=1,20
      AL6(I,J)=AL2(I,J)
  550 CONTINUE
  540 CONTINUE
      DO 570 I=1,MTERM
      I2=MAXTERM+4+I
      I6=MTERM+4+I
      DO 580 J=1,20
      AL6(I6,J)=AL2(I2,J)
  580 CONTINUE
      RH6(I6)=RHS(I2)
  570 CONTINUE
      E(1)=0.0
      E(2)=0.0
      E(3)=0.0
      E(4)=0.0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                  CC
CC    SOLVE FOR LEAST SQUARES SOLUTION OF EQUATIONS (4.3.19) -      CC
CC    - (4.0.23). THE IMSL ROUTINE LLRQF IS ILLUSTRATED HERE        CC
CC                                                                  CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL LLRQF(AL6,44,NN,NSYS,RH6,44,1,0,E,CPOLY,20,IWK,WK,IER)
CC    DISPLAY THE COEFFICIENTS OF EQUATION (4.0.18), THE MELT ZONE  CC
CC    SURFACE CONTROL TEMPERATURE - SEE FINAL REPORT                CC
      WRITE(6,89)
   89 FORMAT(1H1,21X,79HM E L T   Z O N E   S U R F A C E   C O N T R
     1 O L   C O E F F I C I E N T S)
      WRITE(6,789) MTERM,NSYS
  789 FORMAT(//,1H ,50X,12HFOR MTERM = ,I2,12H AND NSYS = ,I2)
      WRITE(6,90)
   90 FORMAT(///,1H ,49X,1HK,22X,4HC(K))
      DO 85 I=1,NSYS
      WRITE(6,686) I, CPOLY(I)
  686 FORMAT(/,1H ,66X,I2,10X,E20.10)
   85 CONTINUE
      CALL FOSTER
 500  CONTINUE
 510  CONTINUE
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                  CC
CC  PURPOSE -                                                       CC
CC    - COMPUTE MATRIX TAT ELEMENTS OF EQUATION (4.0.53)            CC
CC                                                                  CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE INTEGL1(N,K,ANS)
      EXTERNAL G
      COMMON/C25/GRADAT0(101),GRADAT0?(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(44,20),RHS(44),WK(1500),IWK(20)
      COMMON/C27/KERNEL,NKERNEL,KKERNEL
      KERNEL=1
      NKERNEL=N
      KKERNEL=K
```

ORIGINAL PAGE IS
OF POOR QUALITY

Figure C-3. Computer Code List For Problem P1-3 (Cont)

ORIGINAL PAGE IS
OF POOR QUALITY

```
      AERR=0.0
      RERR=1.0E-10
      RERR=1.0E-12
      ANS=DCADRE(G,0,Q,AERR,RERR,ERROR,IER)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC  PURPOSE -                                                           CC
CC    - COMPUTE MATRIX ?A? ELEMENTS OF EQUATION (4.0.33)                CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE INTEGL2(N,K,ANS)
      COMMON/C25/GRADAT0(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(40,20),RHS(40),WWK(1500),IIWK(20)
      COMMON/C27/KERNEL,NKERNEL,KKERNEL
      EXTERNAL G
      KERNEL=2
      NKERNEL=N
      KKERNEL=K
      AERR=0.0
      RERR=1.0E-10
      RERR=1.0E-12
      ANS=DCADRE(G,0,Q,AERR,RERR,ERROR,IER)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC  PURPOSE -                                                           CC
CC    - EVALUATE THE KERNAL FUNCTION DEFINED BY EQUATION (4.0.16)       CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE KERNEL1(N,T,ANS)
      COMMON/C25/GRADAT0(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(40,20),RHS(40),WWK(1500),IIWK(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      Z=-Q*S(N)
      TERM=(P*P-S(N)*S(N))/4.0
      RHOLD=0.0
      IF(Z.GT.-250.0)RHOLD=EXP(Z)
      TERM=TERM*(1.0-RHOLD)
      Z=-(P+S(N))*T/2.0
      Z1=-S(N)*Q+(S(N)-P)*T/2.0
      R1=0.0
      R2=0.0
      IF(Z.GT.-250.0)R1=EXP(Z)
      IF(Z1.GT.-250.0)R2=EXP(Z1)
      ANS=TERM*(R1-R2)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                      CC
CC  PURPOSE -                                                           CC
CC    - EVALUATE THE KERNAL FUNCTION DEFINED BY EQUATION (4.0.17)       CC
CC                                                                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE KERNEL2(N,T,ANS)
      COMMON/C25/GRADAT0(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(40,20),RHS(40),WWK(1500),IIWK(20)
      COMMON/READ1/P,MTERM,MSUM,X0,XN,NGRID,NR
      T1=(P*P-S(N)*S(N))/4.0
      Z=-S(N)*Q
      RHOLD=0.0
      IF(Z.GT.-250.0)RHOLD=EXP(Z)
      T1=T1/(1.0-RHOLD)
      Z=(P-S(N))*0.5*(Q-T)
```

Figure C-3. Computer Code List For Problem P1-3 (Cont)

```
      T2=0.0
      IF(Z.GT.-250.0)T2=EXP(Z)
      Z=-S(N)*T
      T3=1.0
      IF(Z.GT.-250.0)T3=1.0-EXP(Z)
      ANS=T1*T2*T3
      RETURN
      END
```

ORIGINAL PAGE IS OF POOR QUALITY

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC  PURPOSE -                                                          CC
CC    - COMPUTE INTEGRAND USED TO COMPUTE MATRIX ?A? ELEMENTS OF       CC
CC    EQUATION (4.0.23)                                                CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL FUNCTION G(X)
      COMMON/C27/KERNEL,NKERNEL,KKERNEL
      REAL X
      IF(KERNEL.EQ.1) CALL KERNEL1(NKERNEL,X,ANS)
      IF(KERNEL.EQ.2) CALL KERNEL2(NKERNEL,X,ANS)
      CALL BASIS(KKERNEL,X,ANS1)
      G=ANS*ANS1
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC  PURPOSE -                                                          CC
CC    - PROVIDE USER SUPPLIED SET OF FUNCTIONS USED IN EXPANSION       CC
CC    OF MELT ZONE SURFACE CONTROL FUNCTION. SEE EQUATION (4.0.18)     CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE BASIS(K,T,ANS)
      COMMON/C25/GRADATO(101),GRADATR(101),RHS1(20),RHS2(20),S(20),Q,
     1AMAT1(20,10),AMAT2(20,10),AL2(44,20),RHS(44),WWK(1500),IWK(20)
      IF(K.NE.1)GO TO 10
      ANS=1.0
      RETURN
10    ANS=(T-Q/2.0)**(K-1)
      RETURN
      END
      SUBROUTINE DBASIS(K,T,ANS)
      DELTAT=0.001
      X=T+DELTAT
      CALL BASIS(K,X,ANS1)
      X=T-DELTAT
      CALL BASIS(K,X,ANS2)
      ANS=(ANS1-ANS2)/(2.0*DELTAT)
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                     CC
CC    THIS SUBROUTINE ESTIMATES LATERAL SURFACE TEMPERATURE            CC
CC    BY USE OF CUBIC SPLINE IF THE USER SUPPLIES A DISCRETE SET OF    CC
CC    LATERAL SURFACE TEMPERATURES.                                    CC
CC                                                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SPLINE(XINT,YINT)
      COMMON/C32/XD(100),YD(100),C(4,100),M
      IF(XINT-XD(1))2,1,2
    1 YINT=YD(1)
      RETURN
    2 K=1
    3 IF(XINT-XD(K+1))6,4,5
    4 YINT=YD(K+1)
      RETURN
    5 K=K+1
```

Figure C-3. Computer Code List For Problem P1-3 (Cont)

```
      IF((M-K).GT.0) GOTO3
      IF((M-K).LE.0) K=M-1
    6 YINT=(XD(K+1)-XINT)*(C(1,K)*(XD(K+1)-XINT)**2+C(3,K))
      YINT=YINT+(XINT-XD(K))*(C(2,K)*(XINT-XD(K))**2+C(4,K))
      RETURN
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                                                    CC
CC    FIND THE SPLINE CURVE FIT COEFFICIENTS, FOR USE IN CONJUNCTION  CC
CC    WITH SUBROUTINE SPLINE.                                         CC
CC    INPUTS -                                                        CC
CC    M    = NO. OF DATA PAIRS                                        CC
CC    XD   = ARRAY OF X (ABCISSA) VALUES                              CC
CC    YD   = ARRAY OF Y (ORDINATES) VALUES                            CC
CC    OUTPUTS -                                                       CC
CC    C    = 2-D ARRAY OF SPLINE FIT COEFFICIENTS (4 COEFFICIENTS     CC
CC              PER TRIPLET OF DATA POINTS).                          CC
CC                                                                    CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE COFGEN
C
      COMMON/C32/XD(100),YD(100),C1(4,100),M
      DIMENSION C(4,100)
      DIMENSION P(100),E(100),A(100,3),B(100),Z(100),D(100)
      EQUIVALENCE(C1(1,1),C(1,1))
C
      ND=M
      M=M-1
       DO 2 K=1,M
       D(K)=XD(K+1)-XD(K)
       P(K)=D(K)/6.
    2  E(K)=(YD(K+1)-YD(K))/D(K)
       DO 3 K=2,M
    3  B(K)=E(K)-E(K-1)
       A(1,2)=-1.-D(1)/D(2)
       A(1,3)=D(1)/D(2)
       A(2,2)=2.*(P(1)+P(2))-P(1)*A(1,2)
       A(2,3)=(P(2)-P(1)*A(1,3))/A(2,2)
       B(2)=B(2)/A(2,2)
       DO 4 K=3,M
       A(K,2)=2.*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
       B(K)=B(K)-P(K-1)*B(K-1)
       A(K,3)=P(K)/A(K,2)
    4  B(K)=B(K)/A(K,2)
       Q=D(M-1)/D(M)
       A(ND,1)=1.+Q+A(M-1,3)
       A(ND,2)=-Q-A(ND,1)*A(M,3)
       B(ND)=B(M-1)-A(ND,1)*B(M)
       Z(ND)=B(ND)/A(ND,2)
       DO 6 I=1,ND-2
       K=ND-I
    6  Z(K)=B(K)-A(K,3)*Z(K+1)
       Z(1)=-A(1,2)*Z(2)-A(1,3)*Z(3)
       DO 7 K=1,M
       Q=1./(6.*D(K))
       C(1,K)=Z(K)*Q
       C(2,K)=Z(K+1)*Q
       C(3,K)=YD(K)/D(K)-Z(K)*P(K)
    7  C(4,K)=YD(K+1)/D(K)-Z(K+1)*P(K)
      M=M+1
       RETURN
       END
```

Figure C-3. Computer Code List For
Problem P1-3 (Cont)

U.S. GOVERNMENT PRINTING OFFICE 1984-746-070/113