NASA TECHNICAL NOTE

NASA TN D-6701

pt. 1
c. 1

NASA TN D-6701

FORTRAN PROGRAM FOR CALCULATING VELOCITIES IN THE MERIDIONAL PLANE OF A TURBOMACHINE

I — Centrifugal Compressor

by Michael R. Vanco

Lewis Research Center
Cleveland, Ohio 44135

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MARCH 1972

1. Report No.	2. Government Accession No.	3. Recipie...
NASA TN D-6701		

4. Title and Subtitle FORTRAN PROGRAM FOR CALCULATING VELOCITIES IN THE MERIDIONAL PLANE OF A TURBOMACHINE I - CENTRIFUGAL COMPRESSOR

5. Report Date March 1972

6. Performing Organization Code

7. Author(s)
Michael R. Vanco

8. Performing Organization Report No.
E-6592

9. Performing Organization Name and Address
Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio 44135

10. Work Unit No.
132-15

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, D.C. 20546

13. Type of Report and Period Covered
Technical Note

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract
This program will determine the velocities in the meridional plane of a backward-swept impeller, a radial impeller, and a vaned diffuser. The velocity gradient equation with the assumption of a hub-to-shroud mean stream surface is solved along arbitrary quasi-orthogonals in the meridional plane. These quasi-orthogonals are fixed straight lines.

17. Key Words (Suggested by Author(s))
Compressor
Centrifugal turbomachine
Centrifugal compressor

18. Distribution Statement
Unclassified - unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	57	$3.00

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

FORTRAN PROGRAM FOR CALCULATING VELOCITIES IN THE MERIDIONAL PLANE OF A TURBOMACHINE

I - CENTRIFUGAL COMPRESSOR

by Michael R. Vanco

Lewis Research Center

SUMMARY

A FORTRAN IV computer program which calculates the velocities in the meridional plane of a centrifugal compressor is presented. This program will determine the velocities in the meridional plane of a backward-swept impeller, a radial impeller, and a vaned diffuser. The velocity gradient equation with the assumption of a hub-to-shroud mean stream surface is solved along arbitrary quasi-orthogonals in the meridional plane. These quasi-orthogonals are fixed straight lines.

The input quantities for this program consist essentially of mass flow, rotational speed, number of blades, inlet total conditions, loss in relative total pressure, hub-to-shroud profile, mean blade shape, and a normal thickness table. The output yields meridional velocities, approximate blade surface velocities, streamline coordinates, blade shape coordinates, and stream-channel normal thickness in the meridional plane. Numerical examples are included to indicate the use of the program and the results obtained.

INTRODUCTION

Recently, increased interest has been shown in high-pressure-ratio backward-swept centrifugal impeller blades. Centrifugal compressors with backswept impeller blades have the potential of achieving higher efficiencies than those with radial impeller blades. Several methods are available for designing radial-bladed compressors, but limited work has been done on backward-swept impeller blades. Reference 1 gives the method and numerical techniques used to find the flow distribution in the meridional plane of a radial-flow turbine. This method solves the velocity gradient equations with the assumption of a hub-to-shroud mean stream surface. A set of arbitrary straight lines from hub to shroud is used instead of normals. These arbitrary straight lines are called quasi-orthogonals and they remain fixed regardless of any streamline change. This analysis, which has been used for radial-bladed centrifugal impellers, has now been programmed to include backward-swept centrifugal impeller blades.

This report presents a computer program for calculating the velocities in the meridional plane of a centrifugal compressor. This program will determine the velocities in the meridional plane of a backward-swept impeller, a radial impeller, and a vaned diffuser, as well as approximate blade surface velocities. The output of this program is arranged in a form so that it can be used as input to programs used to calculate the blade-to-blade loadings from references 2, 3, or 4.

In this report, a description of the input and output and a FORTRAN IV computer program are presented. A brief description of the method of analysis and the computer program are given. Numerical examples are included to illustrate the use of the program and the results obtained.

METHOD OF ANALYSIS

Reference 1 presents the method and gives the numerical techniques used to find the flow distribution in the meridional plane of a radial-flow turbine. The general velocity gradient equation is derived along an arbitrary quasi-orthogonal in the meridional plane with the assumption of a hub-to-shroud mean stream surface. The equations derived in appendix B of reference 1 are

$$\frac{dW}{ds} = \left(A \frac{dr}{ds} + B \frac{dz}{ds}\right)W + C \frac{dr}{ds} + D \frac{dz}{ds} + \left(\frac{dh_i'}{ds} - \omega \frac{d\lambda}{ds}\right)\frac{1}{W} \tag{1}$$

$$\left.\begin{aligned}
A &= \frac{\cos \alpha \cos^2\beta}{r_c} - \frac{\sin^2\beta}{r} + \sin \alpha \sin \beta \cos \beta\left(\frac{\partial\theta}{\partial r}\right)_f \\
B &= -\frac{\sin \alpha \cos^2\beta}{r_c} + \sin \alpha \sin \beta \cos \beta\left(\frac{\partial\theta}{\partial z}\right)_f \\
C &= \sin \alpha \cos \beta \frac{dW_m}{dm} - 2\omega \sin \beta + r \cos \beta\left(\frac{dW_\theta}{dm} + 2\omega \sin \alpha\right)\left(\frac{\partial\theta}{\partial r}\right)_f \\
D &= \cos \alpha \cos \beta \frac{dW_m}{dm} + r \cos \beta\left(\frac{dW_\theta}{dm} + 2\omega \sin \alpha\right)\left(\frac{\partial\theta}{\partial z}\right)_f
\end{aligned}\right\} \tag{2}$$

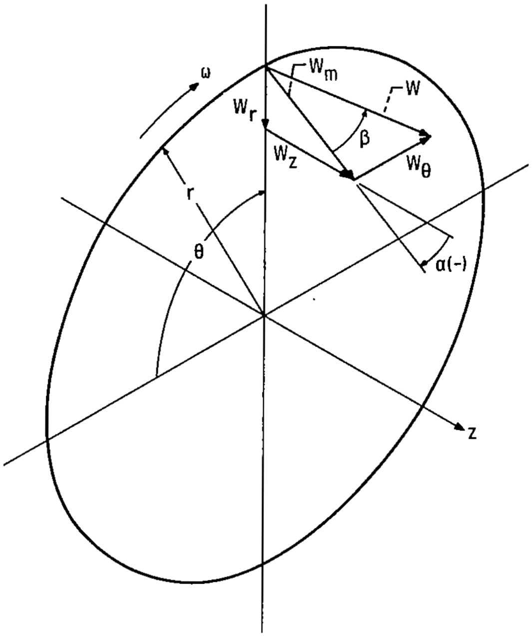

Figure 1. - Coordinate system and velocity components.

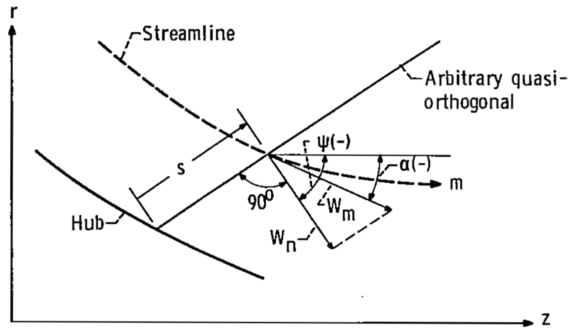

Figure 2. - Component of relative velocity W_n normal to arbitrary quasi-orthogonal.

The coordinate system and nomenclature are shown in figures 1 and 2.

In this analysis, the total enthalpy at the inlet h_i' and the prerotation at the inlet λ, $r_i V_{\theta i}$, are assumed constant. Therefore, equation (1) reduces to

$$\frac{dW}{ds} = \left(A \frac{dr}{ds} + B \frac{dz}{ds}\right) W + C \frac{dr}{ds} + D \frac{dz}{ds} \tag{1a}$$

Continuity must also be satisfied from hub to tip. The calculated mass flow across any fixed line from hub to tip must equal the specified mass flow. The mass flow is computed from

$$w = N \int_0^s \rho W_n r \, \Delta\theta \, ds \tag{3}$$

integrating from hub to tip along a quasi-orthogonal.

The density is calculated from the isentropic flow equation with a correction for loss in total relative pressure. This equation is derived in reference 1. The density equation is

$$\rho = \left(\frac{T}{T'}\right)^{1/(\gamma-1)} \rho_i' - \left[\left(\frac{T}{T_i'}\right)\left(\frac{T_i'}{T''}\right)\right]^{1/(\gamma-1)} \frac{\Delta p''}{RT_i'} \frac{T_i'}{T''} \tag{4}$$

where

$$\frac{T}{T_i'} = 1 - \frac{W^2 + 2\omega\lambda - \omega^2 r^2}{2c_p T_i'} \tag{5}$$

$$\frac{T''}{T_i'} = 1 - \frac{2\omega\lambda - \omega^2 r^2}{2c_p T_i'} \tag{6}$$

and the relative total pressure loss

$$\Delta p'' = p''_{isen} - p'' \tag{7}$$

The change in the angular distance between blades $\Delta\theta$ is

$$\Delta\theta = \frac{2\pi}{N} - \frac{t_\theta}{r} \tag{8}$$

where tangential thickness t_θ is determined from

$$t_\theta^2 = t_n^2 \left[1 + r^2\left(\frac{\partial\theta}{\partial z}\right)^2 + r^2\left(\frac{\partial\theta}{\partial r}\right)^2\right] \tag{9}$$

when the normal thickness t_n is specified.

From figure 2, it can be seen that the velocity normal to the quasi-orthogonal is

$$W_n = W_m \cos(\psi - \alpha) \tag{10}$$

where from figure 1

$$W_m = W \cos \beta \tag{11}$$

The flow angle β is determined from the mean stream surface, $\theta = \theta(m)$, for each streamline, between the blades. Therefore,

$$\tan \beta = r\left(\frac{d\theta}{dm}\right)_f = r\left[\left(\frac{\partial\theta}{\partial r}\right)_f \sin \alpha + \left(\frac{\partial\theta}{\partial z}\right)_f \cos \alpha\right] \tag{12}$$

where $(d\theta/dm)_f$ is the directional derivative along a streamline.

The $\partial\theta/\partial z$ and $\partial\theta/\partial r$ in equation (9) refer to the mean blade shape. The $(\partial\theta/\partial z)_f$ and $(\partial\theta/\partial r)_f$ in equations (1a) and (12) refer to the mean stream surface between the blades. The mean stream surface is assumed to deviate from the mean blade shape at a radius r_b for a centrifugal machine. An approximate equation for determining r_b is given by reference 5,

$$r_b = r_i\, e^{-0.71(\Delta\theta)} \tag{13}$$

The equation for the mean stream surface when $r \geq r_b$ is

$$\theta_f = \frac{\left(\dfrac{\tan \beta_o}{r_o} - \dfrac{\tan \beta_b}{r_b}\right)(m - m_b)^3}{3(m_o - m_b)^2} + \frac{\tan \beta_b}{r_b}(m - m_b) + \theta_b \tag{14}$$

The boundary conditions used to obtain equation (14) were β_o, the outlet flow angle; θ_b, the angular coordinate of the mean blade shape at r_b; and $(d\theta/dm)_b$. Differentiating equation (14), we obtain

$$\left(\frac{d\theta}{dm}\right)_f = \frac{\left(\dfrac{\tan \beta_o}{r_o} - \dfrac{\tan \beta_b}{r_b}\right)(m - m_b)^2}{(m_o - m_b)^2} + \frac{\tan \beta_b}{r_b} \tag{15}$$

It will be noted that equation (1a) is in terms of $(\partial\theta/\partial z)_f$ and $(\partial\theta/\partial r)_f$ and that, on the mean stream surface, θ is a function of the meridional distance m, for each streamline. The relation between them is

$$\left(\frac{d\theta}{dm}\right)_f = \left(\frac{\partial\theta}{\partial r}\right)_f \sin \alpha + \left(\frac{\partial\theta}{\partial z}\right)_f \cos \alpha \tag{16}$$

The preceding equations are solved with the specification of a mean blade shape. The mean blade shape can be specified by two methods. The first method of specifying the mean blade shape is specifying the angular coordinate of the mean blade shape θ constant along a quasi-orthogonal. Since the quasi-orthogonal is a fixed straight line, the mean blade shape is completely specified by specifying θ as a function of the meridional distance m for the hub and shroud streamlines. Therefore, $d\theta/dm$ is known, but the $\partial\theta/\partial r$ and $\partial\theta/\partial z$ have to be determined. If the directional derivative is taken in the m and s direction, then

$$\frac{d\theta}{dm} = \frac{\partial\theta}{\partial r}\frac{dr}{dm} + \frac{\partial\theta}{\partial z}\frac{dz}{dm} = \frac{\partial\theta}{\partial r}\sin\alpha + \frac{\partial\theta}{\partial z}\cos\alpha \tag{17a}$$

and

$$\frac{d\theta}{ds} = \frac{\partial\theta}{\partial r}\frac{dr}{ds} + \frac{\partial\theta}{\partial z}\frac{dz}{ds} = \frac{\partial\theta}{\partial r}\sin(\mu + \alpha) + \frac{\partial\theta}{\partial z}\cos(\mu + \alpha) \tag{17b}$$

With the specification of $d\theta/ds = 0$ and the geometry in figure 3, the following equations are obtained:

$$\frac{\partial\theta}{\partial z} = \frac{\cos\psi}{\cos(\psi - \alpha)}\frac{d\theta}{dm} \tag{18}$$

and

$$\frac{\partial\theta}{\partial r} = \frac{\sin\psi}{\cos(\psi - \alpha)}\frac{d\theta}{dm} \tag{19}$$

This case is used for backswept centrifugal impeller blades. This case is also used for centrifugal diffusers, but equations (13) to (15) are not used because the mean blade shape is the same as the hub-to-shroud mean stream surface.

The second method of specifying the mean blade shape is specifying θ as a function of the axial distance z. This case is used for radial-element centrifugal impellers. Therefore, $\partial\theta/\partial r = 0$ and

$$\frac{d\theta}{dm} = \frac{\partial\theta}{\partial z}\cos\alpha \tag{20}$$

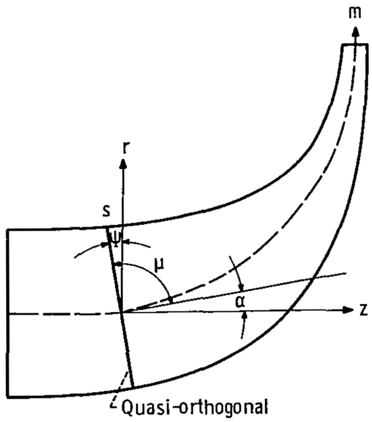

Figure 3. - Relationship between coordinate systems.

However, when slip occurs, that is, when the mean stream surface deviates from the mean blade shape, $(d\theta/dm)_f$ is known from equation (15). It is assumed that the mean stream surface deviates from the mean blade shape only in the radial direction. Therefore, $\partial\theta/\partial z$ is known (mean blade shape), and

$$\left(\frac{\partial\theta}{\partial r}\right)_f = \frac{\left(\frac{d\theta}{dm}\right)_f - \frac{\partial\theta}{\partial z}\cos\alpha}{\sin\alpha} \tag{21}$$

The numerical techniques and procedures used for the solution of equations (1a), (2), and (3) are given in reference 1.

DESCRIPTION OF INPUT

A description of the input for the FORTRAN IV computer program QUAC is given in this section. The input quantities consist essentially of mass flow, rotational speed, number of blades, specific-heat ratio, inlet total temperature and density, gas constant, loss in total relative pressure, hub-to-shroud profile, mean blade shape, and a normal thickness table. Since the program does not use any constants which depend on the system of units being used, any consistent set of units may be used. In the following input, each item has units specified in both the SI and U. S. customary systems.

The input format is shown in table I. The first card is a title card and this card must be put in. The input variables are

MX	number of quasi-orthogonals
KMX	number of streamlines
MR	number of r-values of TN in the thickness table
MZ	number of z-values of TN in the thickness table
W	rotational speed, rad/sec
WT	mass flow, kg/sec; slugs/sec
XN	number of full blades
GAM	specific-heat ratio
AR	gas constant, J/(kg)(K); (ft)(lbf)/(slug)($^{\circ}$R)
TYPE	integer; used as a code to indicate how arrays WA, Z, R, and DN are given initially; the integer values are 0 These quantities will be calculated by the program. 1 Quantities just computed for previous case will be used for next case. (Used only when more than one case is calculated on single computer run.)
MT	number of z-coordinates in ZT array
SRW	integer that will cause the program to print out certain values; used for debugging purposes; the integer values are 0 value when not debugging; usual case 13 SPLINE 16 SPLINT 21 RUUT
MXBL	quasi-orthogonal number where blade starts
TEMP	inlet total temperature, T_i', K; $^{\circ}$R
ALM	inlet prerotation, λ, m^2/sec; ft^2/sec
RHO	inlet total density, ρ_i', kg/m^3; slugs/ft^3
PLOSS	loss in relative total pressure, $\Delta p''$, N/m^2; lb/ft^2

ANGR streamline rotation angle, deg (The streamlines are rotated so that the slope of the program's cubic spline curve is not too large. Good results are obtained from the cubic spline if the absolute value of the slope is not greater than 1. Recommended angles are as follows: for an impeller, 45°; for a diffuser, 90°; and for an axial-flow compressor, 0°.)

KSTH determines the number of times the streamlines are smoothed for each iteration (For example, if KSTH = 0, no smoothing occurs. This is the usual case (KSTH = 0).)

NPRT output control that determines which streamlines are printed out (For example, if NPRT = 1, every streamline is printed out; and if NPRT = 5, every fifth streamline is printed out.)

ITER number of iterations to be performed after ERROR is less than TOLER or after ERROR has started to increase (If ITER = 0, data will be printed for every iteration; if ITER > 0, data will be printed only for the final iteration. Normally ITER = 1, but for a first-run set ITER = 0 and check the first few iterations to see if the data were put in properly.)

KD determines compressor type (For a backward-swept impeller, KD = 0; for a diffuser and an axial-flow compressor, KD = 1; for a radial element impeller, KD = 2.)

SFACT blade multiplier to allow for splitter blades (For the case with no splitters, SFACT = 1.0; and for the case with splitters, SFACT = 2.0.)

ZSPLIT z-coordinate where splitter blade begins, m; ft (If there are no splitters, ZSPLIT > ZH(MX).)

BETO outlet flow angle, β_o, deg

CORFAC ratio of streamline correction used to calculated streamline correction (CORFAC affects the stability of the solution. If too large a value is used, the new streamlines are less smooth than the previous ones. If a computation is based on this set of streamlines, the calculated streamline correction becomes erratic. Therefore, it is important that the streamline correction used give a smooth streamline for the next iteration. A value of 0.1 is recommended.)

SSN last quasi-orthogonal where smoothing is desired (For no smoothing, SSN = 0.)

ZS array of z-coordinates on shroud of hub-to-shroud profile located at quasi-orthogonal positions (see fig. 4), m; ft

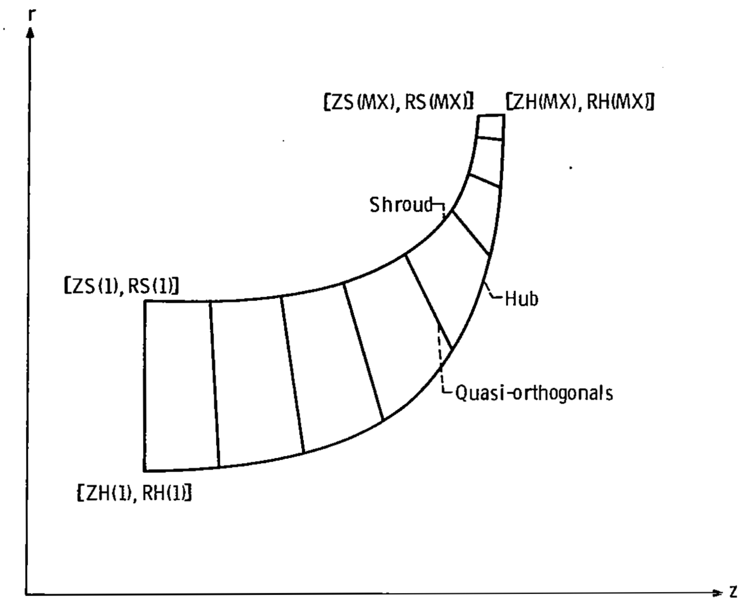

Figure 4. - Input nomenclature.

ZH — array of z-coordinates on hub of hub-to-shroud profile located at quasi-orthogonal positions (see fig. 4), m; ft

RS — array of r-coordinates on shroud corresponding to ZS (see fig. 4), m; ft

RH — array of r-coordinates on hub corresponding to ZH (see fig. 4), m; ft

THTA — array of θ-coordinates (mean blade shape), rad (When KD = 0 and KD = 1, θ is constant along a quasi-orthogonal and must correspond to the ZS, ZH, RS, and RH arrays. When KD = 2, θ is a function of axial distance z and must correspond to the ZT array.)

ZT — array of z-coordinates corresponding to the THTA array, m; ft (Only used when KD = 2.)

TN — array of thicknesses normal to the mean blade shape, t_n, m; ft (This array has z-values of thickness going across and r-values of thickness going down the table. Values of thicknesses and corresponding z- and r-coordinates should extend beyond all boundaries of hub-to-shroud profile so that valid interpolation can be done in the program.)

XZ — array of z-coordinates for thickness table (TN), m; ft (The z-coordinates increase going across the table for a given r-coordinate.)

XR — array of r-coordinates for thickness table (TN), m; ft (The r-coordinates increase going down the table for a given z-coordinate.)

INSTRUCTIONS FOR PREPARING INPUT

Theta Constant Along a Quasi-Orthogonal

After the hub-to-shroud profile has been specified (fig. 5), the mean blade shape is determined. The angular coordinate of the mean blade shape θ is specified as a function of the meridional distance m for the hub and the shroud, as shown in figure 6. Values of θ that are spaced to give good results from a cubic spline used in the program are selected. For a given value of θ, the meridional distances are determined for the hub and shroud from figure 6. These meridional distances are then converted to the proper z- and r-coordinates. Therefore, the z- and r-coordinates for the end points of a quasi-orthogonal have been determined. These are the quantities θ, r_s, z_s, r_h, and z_h that are put in the program. The maximum number of quasi-orthogonal allowed is 21.

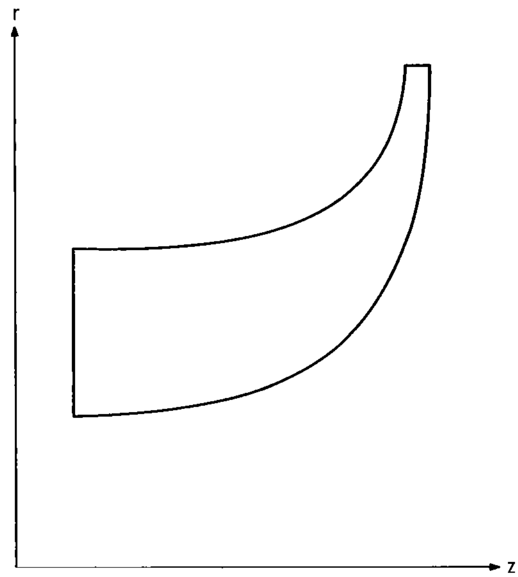

Figure 5. - Hub-to-shroud profile.

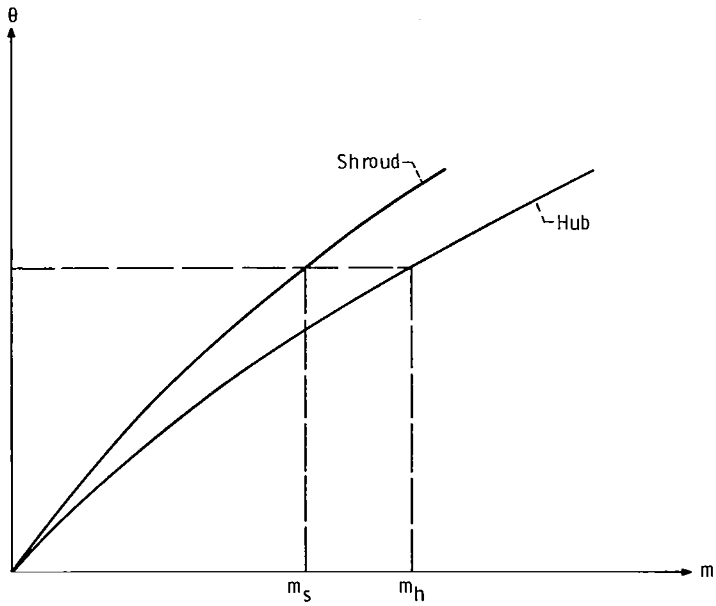

Figure 6. - Hub and shroud mean blade shape. θ = θ(m).

Theta Not Constant Along a Quasi-Orthogonal

This case is used for a radial impeller. The quasi-orthogonals are arbitrarily selected on the hub-to-shroud profile. They should be selected so that the program's cubic spline curve will fit them smoothly. The mean blade shape is determined by specifying θ as a function of the axial distance z, as shown in the third numerical example (p. 18). MT is the number of θ-values used. It should, also, be noted that KD = 2 for this case.

Smoothing of Streamlines

If the streamlines are not smooth, a smoothing routine can be used. KSTH is the number of times the streamlines are smoothed, and SSN is the last quasi-orthogonal where smoothing occurs. For an impeller, the streamline smoothing can take place only in the area shown in figure 7. It cannot take place in the other region because of the methods used. A recommended value for KSTH for smoothing is 4.

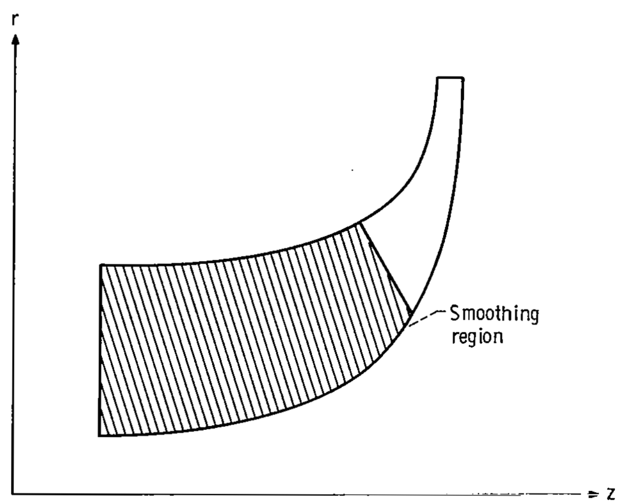

Figure 7. - Streamline smoothing region for centrifugal impeller.

Another method of smoothing the streamlines is to put quasi-orthogonals upstream of the impeller. The mean blade shape is extended into this region with the requirement of a negligible blade loading. These upstream quasi-orthogonals will allow a smoother transition into the impeller. For this case, MXBL is set equal to the quasi-orthogonal number where the blade starts. The first numerical example (p. 14) uses both these techniques.

DESCRIPTION OF OUTPUT

An example of the output from the program is shown in table II. This output is in U. S. customary units. Each section of the output has been numbered to correspond to the following description:

(1) The first output of the program is the input.

(2) Output 2 gives the stagnation speed of sound at the inlet in meters per second (ft/sec); the radius at which the mean stream surface deviates from the mean blade shape (RB) in meters (ft); and a list of the number of iterations required to obtain a solution with the corresponding maximum streamline change in meters (ft).

(3) Output 3 gives some of the important quantities used in the calculation procedure which are also useful for debugging purposes. This output is given for every streamline printed out. Streamline 1 is at the hub and streamline 21 is at the shroud. The number of streamlines printed out is controlled by the input parameter NPRT. Items listed are

ALPHA	angle between meridional streamline and z-axis, deg
RC	curvature of meridional streamline, m^{-1}; ft^{-1}
SM	meridional distance, m; ft
BETA	flow angle, β, deg
TT	tangential blade thickness, m; ft
SA	A, eq. (2)
SB	C, eq. (2)
SC	B, eq. (2)
SD	D, eq. (2)

(4) Output 4 gives the velocities and pressure for every streamline printed out. Items listed are

Z	z-coordinate, m; ft
R	r-coordinate, m; ft
WA	relative velocity on mean stream surface, m/sec; ft/sec
PRESS	static pressure, N/m^2; lb/ft^2
WTR	suction-surface velocity, m/sec; ft/sec
WL	pressure-surface velocity, m/sec; ft/sec
TTREL	total relative temperature, K; ^{o}R

(5) Output 5 gives the stream-channel coordinates and the blade shape coordinates for the hub, mean, and shroud. Only the shroud information is shown here. This information is used to determine the blade-to-blade loading from reference 2, 3, or 4. The M ARRAY, R ARRAY, and the stream-channel normal thicknesses in the meridional plane are in meters (ft); and the THETA ARRAY, the angular coordinates of the blade shape, is in radians.

STGR angular distance from center of trailing-edge circle of blade to center of leading-edge circle of blade, rad

RI leading-edge radius, m; ft

RO trailing-edge radius, m; ft

For the case with splitters, the following additional output is given:

MLER distance from leading edge of blade to leading edge of splitter, m; ft

STGRS angular distance from center of trailing-edge circle of splitter to center of leading-edge circle of splitter, rad

RI leading-edge radius of splitter, m; ft

RO trailing-edge radius of splitter, m; ft

BETAS flow angle at leading edge of splitter, deg

(6) Output 6 gives the inlet flow angle for the hub, mean, and tip, in degrees. These angles are calculated inside the blade passage.

NUMERICAL EXAMPLES

To indicate the use of the program and the results obtained, three numerical examples are given. The first example is a backward-swept centrifugal compressor rotor, the second is a centrifugal compressor diffuser, and the third is the input for a radial compressor. All examples are in U. S. customary units.

Backward-Swept Centrifugal Compressor

This compressor has a 6-to-1 pressure ratio. The hub-to-shroud profile of the impeller is shown in figure 8. The mean blade shape is given in figure 9, where θ is specified as a function of the meridional distance m for the hub and shroud. The quasi-

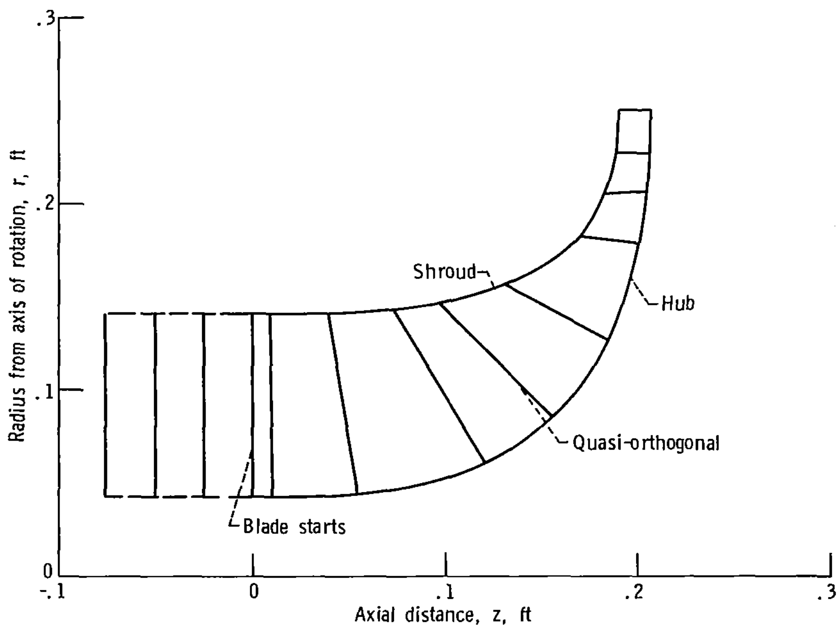

Figure 8. - Hub-to-shroud profile of backswept impeller.

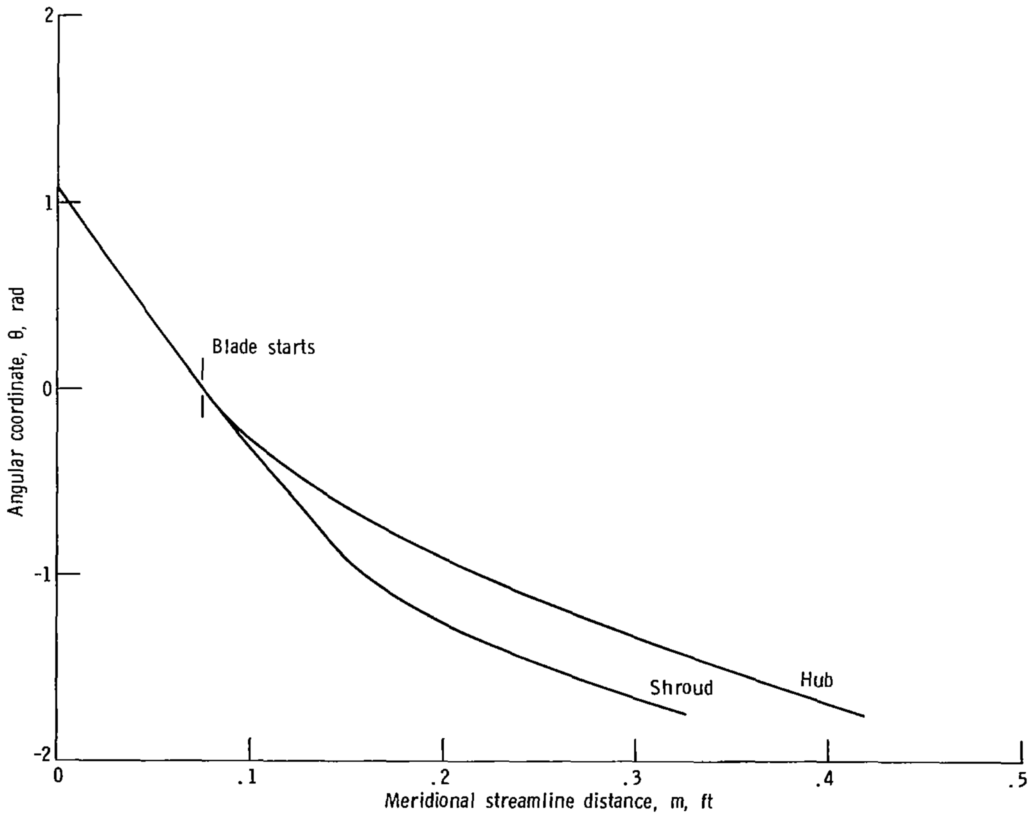

Figure 9. - Mean blade shape of backswept impeller.

orthogonals shown in figure 8 depend on the mean blade shape in figure 9 because θ is constant along a quasi-orthogonal. It will be noted that in this example three quasi-orthogonals were put upstream of the impeller. This was done to allow a smooth flow transition into the impeller because of the low inlet hub-to-tip radius ratio and the high rpm. Streamline smoothing was also used. MXBL was set equal to 4, SSN set equal to 8.0, and KSTH set equal to 4. The input for this case is given in table III. The mean stream surface relative velocities are plotted in figure 10 for the hub, mean, and shroud streamlines. The velocity change near the impeller inlet was due to the blade blockage.

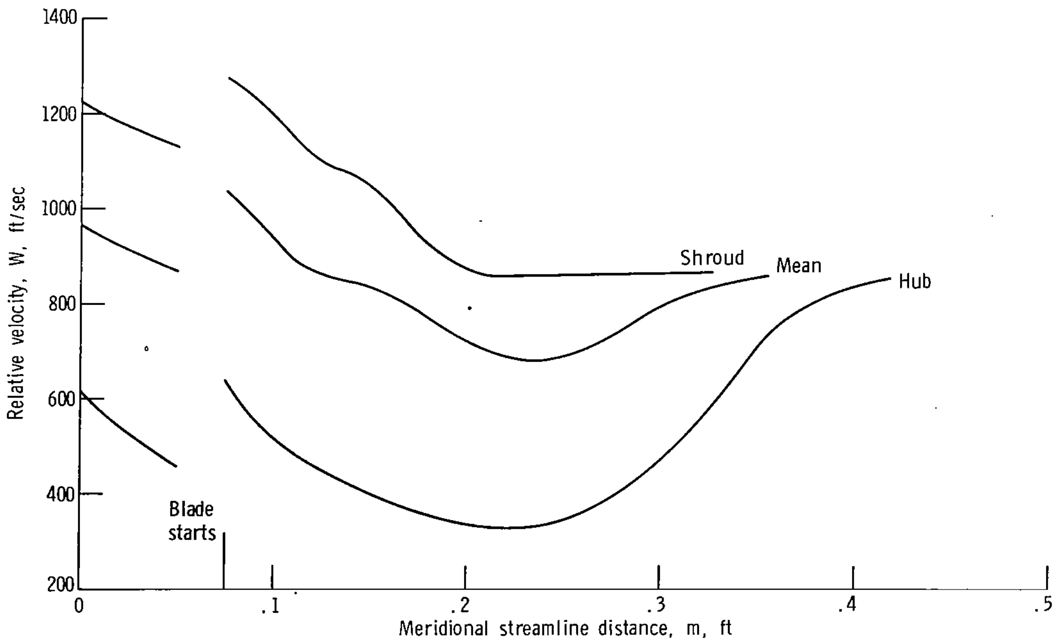

Figure 10. - Relative velocities in meridional plane of backswept impeller.

The blade shape coordinates and the stream-channel normal thickness needed for calculating the blade loading from reference 2, 3, or 4 are given in table IV for the mean streamline. These results are also obtained for the hub and shroud streamlines, but they are not shown here.

Diffuser

A flat-vaned diffuser for a centrifugal compressor was designed to have a linear static-pressure gradient from inlet to outlet. The meridional profile is shown in figure 11. The angular coordinate of mean blade shape θ is given as a function of the me-

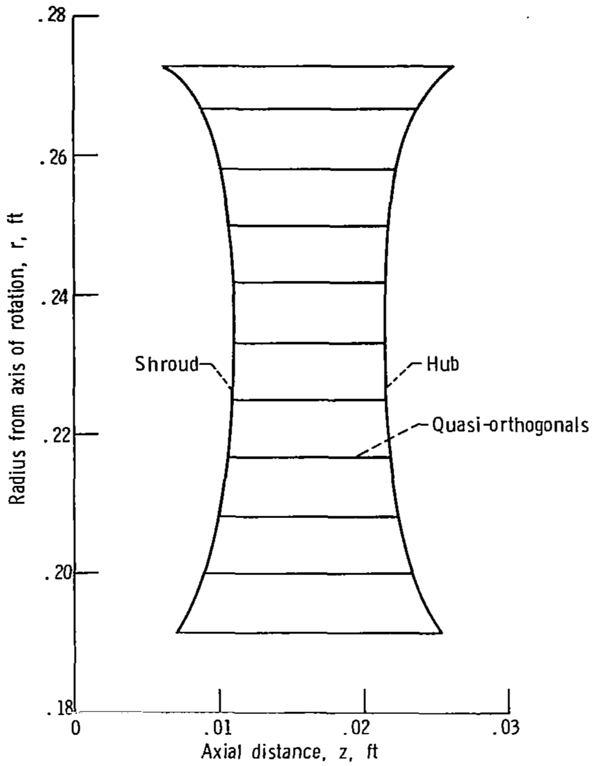

Figure 11. - Hub-to-shroud profile of compressor diffuser.

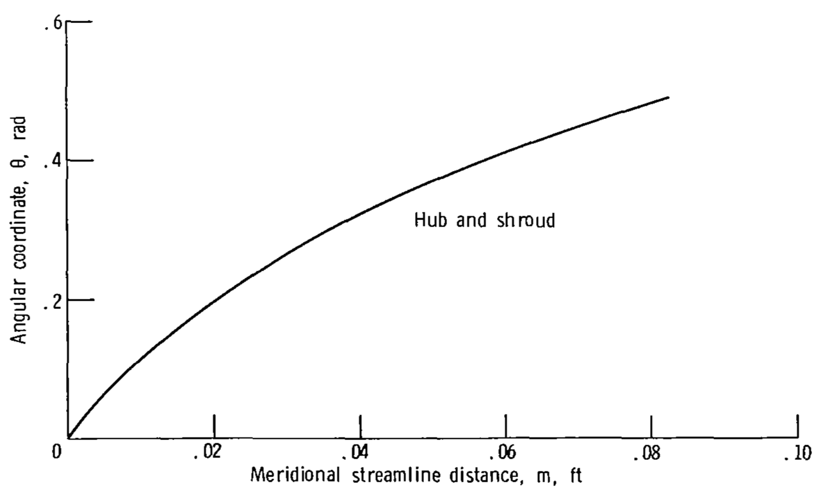

Figure 12. - Mean blade shape for compressor diffuser.

ridional distance m in figure 12. The quasi-orthogonals shown in figure 11 depend on the mean blade shape in figure 12 because θ is constant along a quasi-orthogonal. The input for this case is given in table V. The mean stream surface velocities and the approximate blade surface velocities are plotted in figure 13 for the hub, mean, and shroud streamlines. The blade shape coordinates and the stream-channel normal thickness needed for calculating the blade loading from reference 2, 3, or 4 are given in table VI for the mean streamline.

Radial Impeller

This example is used to indicate the different input required. A hub-to-shroud profile is given in figure 14. The quasi-orthogonals for the profile shown are arbitrary and do not depend on the mean blade shape; that is, θ is not constant along a quasi-

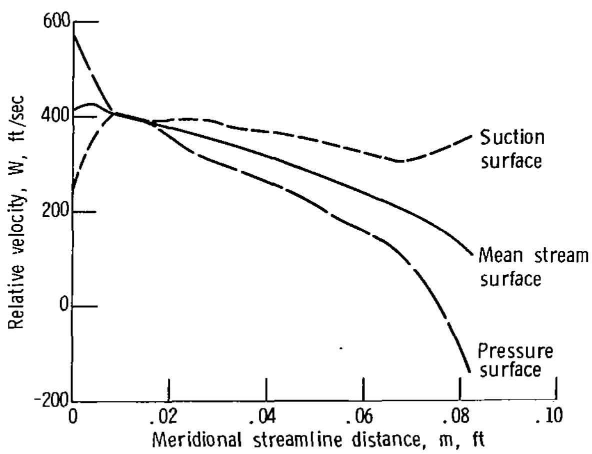

Figure 13. - Blade loading diagram for hub and shroud of compressor diffuser.

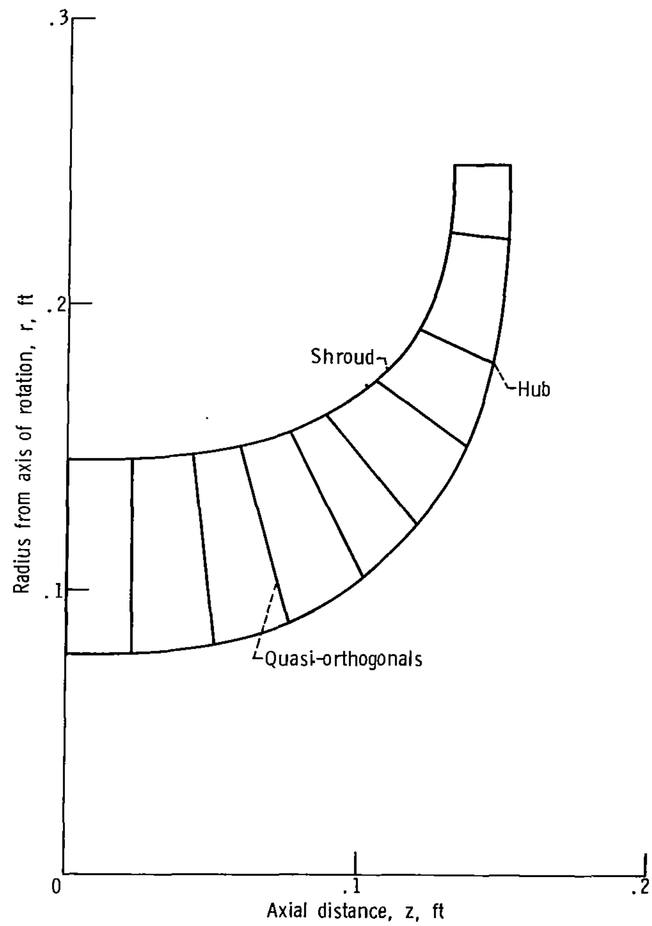

Figure 14. - Hub-to-shroud profile of radial impeller.

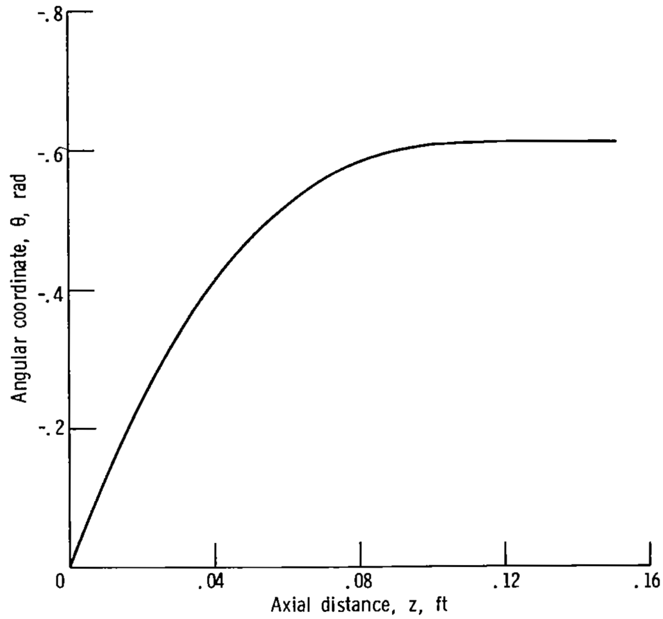

Figure 15. - Mean blade shape for radial impeller.

orthogonal. The mean blade shape is put in as a function of the axial distance z, as shown in figure 15. Sample input is shown in table VII. The output obtained is the same as in the other examples.

PROGRAM DESCRIPTION

Main Program QUAC

The main program QUAC contains all the equations given in the method of analysis and makes the majority of the calculations. It will be noted that K is used for the streamline number and I is used for the quasi-orthogonal number. QUAC calls the subroutines RUUT, SMOOTH, INTGRL, CONTIN, SPLDER, SPLINE, LININT, and SPLINT to perform various functions such as smoothing, finding roots, integration, interpolation, and use of a cubic spline curve to determine derivatives. These subroutines, excluding RUUT and SMOOTH, are described in reference 1. A brief description of each is given herein.

The program variables for QUAC are

A temporary storage

AB temporary storage

AC	temporary storage
AD	temporary storage
AE	meridional length from leading edge
AL	α
ALM	see input
AMLER	MLER (see output)
ANGR	see input
AR	see input
B	temporary storage
BA	total weight flow between hub and K^{th} streamline
BETA	β
BETAD	β_l
BETAS	see output
BETAT	β_t
BETOH	exit blade angle at hub
BETOM	exit blade angle at mean
BETOT	exit blade angle at tip
C	temporary storage
CAL	$\cos \alpha$
CBETA	$\cos \beta$
CI	stagnation speed of sound at inlet
CORFAC	see input
COSBD	$\cos \beta_l$
COSBT	$\cos \beta_t$
CP	c_p
CURV	$1/r_c$
DELBTA	$\beta_t - \beta_l$
DELTA	calculated streamline correction

DENSTY	ρ
DN	distance along quasi-orthogonal from hub
DRDM	$\frac{d}{dm}(r\omega + W \sin \beta)r \; \Delta\theta$
DTDMB	$(d\theta/dm)_b$
DTDMS	$d\theta/dm$ at splitter leading edge
DTDR	$\partial\theta/\partial r$
DTDZ	$\partial\theta/\partial z$
DWMDM	dW_m/dm
DWTDM	dW_θ/dm
E	temporary storage
ERROR	maximum calculated streamline correction for present iteration
ERROR1	ERROR from previous iteration
EXPON	$1/(\gamma - 1)$
G	temporary storage
GAM	γ
HR	increment along quasi-orthogonal in r-direction
HZ	increment along quasi-orthogonal in z-direction
I	subscript to indicate number of quasi-orthogonal
IND	code number for use by subroutine CONTIN
INF	set equal to 1, when (ZH - ZS) = 0
ITER	see output
K	subscript used to indicate streamline number
KD	see input
KMX	see input
KMXM1	KMX - 1
KSTH	see input
MR	see input
MT	see input
MX	see input

MZ	see input
N	MXBL
N1	N + 1
N2	MX - 1
NPRT	see input
PLOSS	see input
PRS	p
PSI	ψ
R	r
RB	r_b
RC	$1/r_c$
REXIT	average exit radius
RH	see input
RHO	ρ_i'
RI	leading-edge radius
RIS	leading-edge radius of splitter
RO	trailing-edge radius
RS	see input
RSPLIT	r-coordinate at leading edge of splitter
RUNO	run number
SA	A, eq. (2)
SAL	sin α
SB	C, eq. (2)
SBETA	sin β
SC	β, eq. (2)
SD	D, eq. (2)
SFACT	see input
SLA	average distance between streamlines on a quasi-orthogonal
SM	distance from inlet along a meridional streamline

SM1	meridional distance from first quasi-orthogonal to quasi-orthogonal that is before point where stream surface deviates from blade surface
SM2	meridional distance from first quasi-orthogonal to a quasi-orthogonal that is after point where stream surface deviates from blade surface
SMF	fractional meridional distance
SMEXIT	meridional distance from first quasi-orthogonal to trailing edge of blade
SMRB	meridional distance from first quasi-orthogonal to point where mean stream surface deviates from mean blade shape
SRW	see input
SSN	see input
STGR	see output
T	t_n (interpolated value)
TANBB	tan β_b
TANS	tan β_s, at leading edge of splitter
TEMP	T_i'
THTAB	θ_b
THTAF	θ_f
THTAS	θ_s
THH	mean blade shape θ-coordinate at hub
THHC	temporary storage
THH1	blade shape, θ-coordinate at hub on surface 1
THH2	θ-coordinate at hub on blade surface 2
THM	θ-coordinate of mean blade shape at mean
THMC	temporary storage
THM1	θ-coordinate at mean on blade surface 1
THM2	θ-coordinate at mean on blade surface 2
THS	θ-coordinate of mean blade shape at shroud
THSC	temporary storage
THSI	θ-coordinate at shroud on blade surface 1
THS2	θ-coordinate at shroud on blade surface 2

TN	see input
TOLER	iteration tolerance
TSPLIT	normal blade thickness at leading edge of splitter
TPP1P	T''/T_i'
TTREL	see output
TT	t_θ
TYPE	see input
T1P	T/T_i'
W	ω
WA	W
WAS	W*, eq. (13) of ref. 1
WASS	W**, eq. (13) of ref. 1
WT	total mass flow
WTFL	calculated total mass between hub and K^{th} streamline
WTHRU	W_n
WTR	W_t, eq. (10) of ref. 1 (suction-surface velocity)
WL	pressure-surface velocity
XN	see input
XR	see input
XZ	see input
YA	average weight flow per unit length crossing a quasi-orthogonal
YH	temporary storage
YM	temporary storage
YS	temporary storage
Z	z
ZEXIT	average z-coordinate at exit
ZH	see input
ZS	see input
ZSPLIT	see input
ZT	see input

Subroutine RUUT

Subroutine RUUT finds the root between two given points. It is used to find the meridional distance where the mean stream surface deviates from the mean blade shape when the radius at which this occurs is given. If the root cannot be found within the tolerance, a message is printed out and the input arguments are listed. If there is trouble in finding a root, set SRW = 21 in the input and all the input to the subroutine will be printed out.

The calling sequence for RUUT is

```
CALL RUUT(SM1, SM2, RB, SMRB, SM(1, K), R(1, K), MX)
```

where

SM1	meridional distance of quasi-orthogonal before point desired (input)
SM2	meridional distance of quasi-orthogonal after point desired (input)
RB	radius at point desired (input)
SMRB	desired meridional distance (output)
SM(1, K)	array of m-coordinates (input)
R(1, K)	array of r-coordinates (input)
MX	number of r-coordinates (input)

Subroutine SMOOTH

Subroutine SMOOTH smoothes the streamlines to obtain a better numerical solution. It uses the hub streamline as the base streamline for the smoothing operation.

The slopes of the quasi-orthogonals are

$$m_I = \frac{y_{s_I} - y_{h_I}}{x_{s_I} - x_{h_I}} \tag{22}$$

and the streamline slopes are

$$m_K = \frac{y_{I+1} - y_{I-1}}{x_{I+1} - x_{I-1}} \tag{23}$$

where K is the streamline number and I is the quasi-orthogonal number. The x-y coordinates of an intersection can now be determined. From analytical geometry,

$$\left(x_1\right)_I = \frac{(y_{I-1} - m_K x_{I-1}) - \left(y_{h_I} - m_K x_{h_I}\right)}{m_I - m_K} \tag{24}$$

The smoothed x-coordinate is

$$\left(x_1\right)_I = \frac{\left(x_1\right)_I - x_I}{D} + x_I \tag{25}$$

where D is the smoothing factor. The smoothed y-coordinate is

$$y_I = y_{h_I} + m_I\left(x_{1_I} - x_{h_I}\right) \tag{26}$$

When $m_I = 0$, the following equations are used:

$$\left(y_1\right)_I = m_K(x_I - x_{I-1}) + y_{I-1} \tag{27}$$

$$\left(y_1\right)_I = \frac{\left(y_1\right)_I - y_I}{D} + y_I \tag{28}$$

and

$$\left(x_1\right)_I = x_I \tag{29}$$

When $m_K = 0$,

$$\left(y_1\right)_I = m_I\left(x_{I-1} - x_{h_I}\right) + y_{h_I}$$

$$\left(y_1\right)_I = \frac{\left(y_1\right)_I - y_I}{D} + y_I \tag{30}$$

and

$$\left(x_1\right)_I = \frac{\left(x_1\right)_I - x_I}{D} + x_I \tag{31}$$

The value of D is 2 for all quasi-orthogonals except for the last three, where smoothing occurs. The values of D for these three are 2.6667, 4.0, and 8.0, respectively. This was done so that there would not be any discontinuities when only certain sections of the streamlines are smoothed.

The calling sequence for SMOOTH is

CALL SMOOTH(Z(1, K), R(1, K), ZH, RH, AB, SSN, INF)

where

Z(1, K)	z-coordinate of streamline
R(1, K)	r-coordinate of streamline
ZH	z-coordinate of hub streamline
RH	r-coordinate of hub streamline
AB	slope of quasi-orthogonals, m_I
SSN	last quasi-orthogonal where smoothing is desired
INF	indicator for quasi-orthogonals with a slope of infinity

The program variables are

D	smoothing factor
SLOPE	slope of quasi-orthogonals, m_I
SLOPE1	streamline slope, m_K
X	z-coordinate of streamlines
XH	z-coordinate of hub streamline
X1	z-coordinate of smoothed streamline
Y	r-coordinate of streamlines
YH	r-coordinate of hub streamline
Y1	r-coordinate of smoothed streamline

Other Subroutines

Subroutines INTGRL, CONTIN, SPLDER, SPLINE, LININT, and SPLINT are described in reference 1. INTGRL is used for numerical integration. CONTIN is used to determine the hub velocity for the next continuity iteration. SPLDER is used to determine the values of the derivatives at the specified interpolated points. SPLINE is used to determine the first and second derivatives. If there is a problem with the SPLINE subroutine, set SRW = 13 in the input and the input and output of the SPLINE subroutine will be printed out. LININT is used to determine the interpolated values of the normal blade thickness from the given thickness table. SPLINT is used for interpolation. The input and output data for SPLINT will be printed out if SRW = 16.

PROGRAM LISTING

```
$IBFTC QUAC     DECK

C  CALCULATION OF VELOCITY AND PRESSURE DISTRIBUTION IN A CENTRIFUGAL COMPRESSOR
C              BY USE OF QUASI-ORTHOGONALS
C
C
      COMMON SRW
      DIMENSION AL(21,21),BETA(21,21),CAL(21,21),CBETA(21,21),INF(21),
     1CURV(21,22),DN(21,21),PRS(21,21),R(21,21),Z(21,21),SM(21,21),
     2SA(21,21),SB(21,21),SC(21,21),SD(21,21),SAL(21,21),SBETA(21,21),
     3TN(21,21),TT(21,21),WA(21,21),WTR(21,21),TTREL(21,21),WL(21,21)
      DIMENSION AB(22),AC(22),AD(22),BA(21),DELBTA(21),DRDM(21), AE(22),
     1YM(21),  DTDM(21),DWMDM(21),DWTDM(21),RH(21),RS(21),ZH(21),ZS(21),
     2THTA(21),WTFL(21),XR(21),XT(21),XZ(21),BETAI(3),AA(3),THTAF(21)
      DIMENSION THH(21),THM(21),THS(21),THH1(21),THH2(21),THM1(21),
     1THM2(21),THS1(21),THS2(21),DTDZ(21),DTDR(21),ZT(21),
     2YA(21),YH(21),YS(21),TI(3),TO(3)
      INTEGER RUNO,TYPE,      SRW,HUB,SHROUD
      RUNO=0
10    READ (5,1001)
      WRITE(6,1049)
      WRITE(6,1001)
      READ  (5,1010)MX,KMX,MR,MZ,W,WT,XN,GAM,AR
      ITNC = 1
      RUNO=RUNO+1
      WRITE (6,1020) RUNO
      WRITE(6,1007)
      WRITE (6,1011)MX,KMX,MR,MZ,W,WT,XN,GAM,AR
      READ (5,1010)TYPE,  MT,SRW,MXBL,TEMP,ALM,RHO,PLOSS ,ANGR
      WRITE(6,1008)
      WRITE(6,1011)TYPE,  MT,SRW,MXBL,TEMP,ALM,RHO,PLOSS,ANGR
      READ (5,1010) KSTH,NPRT,ITER,KD   ,SFACT,ZSPLIT,BETO ,CORFAC,SSN
      WRITE(6,1009)
      WRITE(6,1011) KSTH,NPRT,ITER,KD   ,SFACT,ZSPLIT,BETO ,CORFAC,SSN
      ITER1 = ITER
      READ(5,1030)(ZS(I),I=1,MX)
      WRITE(6,1029)
      WRITE(6,1028)(ZS(I),I=1,MX)
      READ(5,1030)(ZH(I),I=1,MX)
```

```
      WRITE(6,1031)
      WRITE(6,1028)(ZH(I),I=1,MX)
      READ(5,1030)(RS(I),I=1,MX)
      WRITE(6,1032)
      WRITE(6,1028)(RS(I),I=1,MX)
      READ(5,1030)(RH(I),I=1,MX)
      WRITE(6,1033)
      WRITE(6,1028)(RH(I),I=1,MX)
      IF(TYPE.NE.0) GO TO 145
      IF(RS(1).EQ.RH(1)) GO TO 20
      WA(1,1) = WT/RHO/((RS(1)**2-RH(1)**2)*3.14)
      GO TO 21
 20   WA(1,1)=WT/RHO/(ZH(1)-ZS(1))/3.14/(RS(1)+RH(1))
 21   DO 30 I=1,MX
      DN(I,KMX)=SQRT((ZS(I)-ZH(I))**2+(RS(I)-RH(I))**2)
      DO 30 K=1,KMX
      DN(I,K)=FLOAT(K-1)/FLOAT(KMX-1)*DN(I,KMX)
      WA(I,K)=WA(1,1)
      Z(I,K)=DN(I,K)/DN(I,KMX)*(ZS(I)-ZH(I))+ZH(I)
   30 R(I,K)=DN(I,K)/DN(I,KMX)*(RS(I)-RH(I))+RH(I)
      IF ( KD.EQ.2 ) GO TO 50
      READ  (5,1030)(THTA(I),I=1,MX    )
      WRITE(6,1034)
      WRITE (6,1028)(THTA(I),I=1,MX    )
      WRITE(6,1035)
      GO TO 51
   50 READ  (5,1030)(THTA(I),I=1,MT    )
      WRITE(6,1034)
      WRITE (6,1028)(THTA(I),I=1,MT    )
      READ(5,1030)(ZT(I),I=1,MT)
      WRITE(6,1039)
      WRITE(6,1028)(ZT(I),I=1,MT)
   51 WRITE(6,1036)
      DO 60 K=1,MR
      READ  (5,1030)(TN(I,K),I=1,MZ)
   60 WRITE (6,1028)(TN(I,K),I=1,MZ)
      READ  (5,1030)(XZ(I),I=1,MZ)
      WRITE(6,1037)
      WRITE (6,1028)(XZ(I),I=1,MZ)
      READ  (5,1030)(XR(I),I=1,MR)
      WRITE(6,1038)
      WRITE (6,1028)(XR(I),I=1,MR)
C
C   END OF INPUT STATEMENTS
C
C   INITIALIZE,CALCULATE CONSTANTS
C
      WTOLER = WT/100000.
      TOLER = (RS(1)-RH(1))/5000.
      IF(RS(1).EQ.RH(1)) TOLER= (ZH(1)-ZS(1))/5000.
      DO 110 K=1,KMX
  110 SM(1,K)=0.
      BA(1)=0.
      DO 120 K=2,KMX
  120 BA(K) = FLOAT(K-1)*WT/FLOAT(KMX -1)
      DO 130 I=1,MX
  130 DN(I,1)=0.
      ANGR = ANGR/57.29577
145   CONTINUE
      CI = SQRT(GAM*AR*TEMP)
      WRITE(6,1049)
      WRITE (6,1050) CI
```

```
      KMXM1 = KMX-1
      CP=AR*GAM/(GAM-1.)
      EXPCN = 1./(GAM-1.)
      BETC  =  BETO /57.29577
      ZEXIT = (ZS(MX)+ZH(MX))/2.
      REXIT = (RS(MX)+RH(MX))/2.
      IF ( KD.EQ. 1 ) GO TO 149
      CALL LININT (ZEXIT ,REXIT ,XZ,XR,TN,21,21,T)
      RB = REXIT *EXP(-.71*(2.*3.14159/(XN*SFACT)-T/REXIT ))
      WRITE (6,1027) RB
149   ERRCR=100000.
C
C  BEGINNING OF LOOP FOR ITERATIONS
  150 IF(ITER.EQ.0) WRITE (6,1060) ITNO
C
      IF(ITER.EQ.0) WRITE (6,1070)
      ERRCR1=ERRCR
      ERRCR=0.
C
C  START CALCULATION OF PARAMETERS
C
      DO 180 K=1,KMX
      DO 180 I=2,MX
      SM(I,K) = SM(I-1,K)+SQRT((Z(I,K)-Z(I-1,K))**2+(R(I,K)-R(I-1,K))**
     1 2)
180   CONTINUE
      DO 230 K=1,KMX
      DO 160 I=1,MX
      AB(I) = Z(I,K)*COS(ANGR)  + R(I,K)*SIN(ANGR)
160   AC(I) = R(I,K)*COS(ANGR)  - Z(I,K)*SIN(ANGR)
      CALL SPLINE (AB,AC,MX,AL(1,K),CURV(1,K))
      DO 170 I=1,MX
      CURV(I,K)=CURV(I,K)/(1.+AL(I,K)**2)**1.5
      AL(I,K) = ATAN(AL(I,K))+ANGR
      CAL(I,K) = COS(AL(I,K))
  170 SAL(I,K) = SIN(AL(I,K))
      IF ( KD.EQ. 2) GO TO 171
      CALL SPLINE (SM(1,K),THTA   ,MX,DTDM,AC)
      GO TO 172
  171 CALL SPLDER(ZT,THTA,MT,Z(1,K),MX,DTDZ)
172   DO 204 I =1,MX
      T = 0.
      THTAF(I) = THTA(I)
      IF(I.GE.MXBL) CALL LININT(Z(I,K),R(I,K),XZ,XR,TN,21,21,T)
      IF (ZS(I).GE.ZH(I)) GO TO 202
      PSI = ATAN((RS(I)-RH(I))/(ZS(I)-ZH(I)))+1.5708
      GO TO 203
202   PSI = ATAN((ZH(I)-ZS(I))/(RS(I)-RH(I)))
203   IF ( KD.EQ. 2 ) DTDM(I) = DTDZ(I)*CAL(I,K)
      IF ( KD.EQ. 2 ) DTDR(I) = 0.0
      IF (KD.NE. 2 ) DTDZ(I) = COS(PSI)/COS(PSI -AL(I,K))*DTDM(I)
      IF (KD.NE. 2 ) DTDR(I) = SIN(PSI)/COS(PSI -AL(I,K))*DTDM(I)
204   TT(I,K) =T*SQRT(1.0+R(I,K)**2*(DTDR(I)**2+DTDZ(I)**2))
      IF (KD.EQ.1) GO TO 207
      DO 205 I =1,MX
      IF (R(I,K).GT. RB) GO TO 206
205   CONTINUE
206   SM1 = SM(I-1,K)
      SM2 = SM(I,K)
      CALL RUUT (SM1,SM2,RB,SMRB,SM(1,K),R(1,K),MX)
      IF (KD .EQ. 2 ) CALL SPLINT (ZT,THTA,MT,Z(1,K),MX,THTAF)
      CALL SPLINT (SM(1,K),THTAF,MX,SMRB,1,THTAB)
```

```
      CALL SPLCER (SM(1,K),THTAF,MX,SMRB,1,DTDMB)
      TANRB = RB*DTDMB
      SMEXIT = SM(MX,K)
      DC 201 I =1,MX
      IF (R(I,K).LT.RB) GO TO 201
      THTAF(I) = THTAB + (SM(I,K)-SMRB)**3*(TAN(BETO)/REXIT-TANBB/RB)/
     1(3.0*(SMEXIT-SMRB)**2) + (SM(I,K)-SMRB)* TANBB/RB
      DTCM(I)  =         + (SM(I,K)-SMRB)**2*(TAN(BETO)/REXIT-TANBB/RB)/
     1((SMEXIT-SMRB)**2)+TANBB/RB
      IF ( SAL(I,K) .EQ. 0.0 ) GO TO 200
      DTCR(I) = (DTDM(I)-DTDZ(I)*CAL(I,K))/SAL(I,K)
      GC TC 201
200   DTCR(I) =0.0
201   CCNTINUE
207   DC 220 I=1,MX
      BETA(I,K) = ATAN(R(I,K)*DTDM(I))
      SBETA(I,K) = SIN(BETA(I,K))
      CBETA(I,K) = COS(BETA(I,K))
      AB(I)=WA(I,K)*CBETA(I,K)
  22C AC(I)=WA(I,K)*SBETA(I,K)
      CALL SPLINE(SM(1,K),AB,MX,DWMDM,AD)
      CALL SPLINE(SM(1,K),AC,MX,DWTDM,AD)
      IF((ITER.LE.0).AND.(MOD(K-1,NPRT).EQ.0)) WRITE (6,1C80) K
      DC 23C I=1,MX
      SA(I,K) = CBETA(I,K)**2*CAL(I,K)*CURV(I,K)-SBETA(I,K)**2/R(I,K)
     1+SAL(I,K)*CBETA(I,K)*SBETA(I,K)*DTDR(I)
      SB(I,K) = SAL(I,K)*CBETA(I,K)*DWMDM(I) -2.0*W*SBETA(I,K) +DTDR(I)
     1*R(I,K)*CBETA(I,K)*(DWTDM(I)+2.*W*SAL(I,K))
      SC(I,K) = -CBETA(I,K)**2*SAL(I,K)*CURV(I,K)
     1+SAL(I,K)*CBETA(I,K)*SBETA(I,K)*DTDZ(I)
      SD(I,K) = CAL(I,K)*CBETA(I,K)*DWMDM(I) +DTDZ(I)
     1*R(I,K)*CBETA(I,K)*(DWTDM(I)+2.*W*SAL(I,K))
      IF((ITER.GT.0).OR.(MOD(K-1,NPRT).NE.0))GO TO 230
      A=  AL(I,K)*57.29577
      B=  SM(I,K)
      E=  TT(I,K)
      G=BETA(I,K)*57.29577
      WRITE (6,1090) A,CURV(I,K),B,G,E,  SA(I,K),SB(I,K),SC(I,K),SD(I,K)
  23C CCNTINUE
C
C  END CF LCOP - PARAMETER CALCULATION
C  CALCULATE BLADE SURFACE VELOCITIES (AFTER CONVERGENCE)
C
      IF(ITER.NE.0) GO TO 260
      DC 25C K=1,KMX
      CALL SPLINE (SM(1,K),TT(1,K),MX,DELBTA,AC)
      A=XN
      DC 240 I=1,MX
  24C AB(I)=(R(I,K)*W+WA(I,K)*SBETA(I,K))*(6.283186*R(I,K)/ A-TT(I,K))
      CALL SPLINE (SM(1,K),AB,MX,DRDM,AC)
      IF (SFACT.LE. 1.0) GC TO 245
      A = SFACT*XN
      DC 244 I=1,MX
  244 AB(I)=(R(I,K)*W+WA(I,K)*SBETA(I,K))*(6.283186*R(I,K)/ A-TT(I,K))
      CALL SPLINE (SM(1,K),AB,MX,AD   ,AC)
  245 DC 250       I=1,MX
      BETAC = BETA(I,K)-DELBTA(I)/2.
      BETAT = BETAD+DELBTA(I)
      CCSBC = COS(BETAC)
      CCSBT = COS(BETAT)
      IF(Z(I,K).GT.ZSPLIT) DRDM(I) = AD(I)
```

```
      WTR(I,K)=COSBD*COSBT/(COSBD+COSBT)*(2.*WA(I,K)/COSBD+R(I,K)*W*
     1(BETAD-BETAT)/CBETA(I,K)**2+CRDM(I))
      WL(I,K) = 2.0*WA(I,K)-WTR(I,K)
  250 CONTINUE
C
C       END  OF  BLADE  SURFACE  VELOCITY  CALCULATIONS
C  START CALCULATION OF WEIGHT FLOW VS. DISTANCE FROM HUB
C
  260 DO 370 I=1,MX
      IND=1
      DO 270 K=1,KMX
  270 AC(K)=DN(I,K)
      GO TO 290
  280 WA(I,1)=.5*WA(I,1)
  290 DO 300 K=2,KMX
      J=K-1
      HR=R(I,K)-R(I,J)
      HZ=Z(I,K)-Z(I,J)
      WAS = WA(I,J)*(1.0+SA(I,J)*HR+SC(I,J)*HZ) +SB(I,J)*HR+SD(I,J)*HZ
      WASS = WA(I,J)+WAS*(SA(I,K)*HR+SC(I,K)*HZ)+SB(I,K)*HR+SD(I,K)*HZ
  300 WA(I,K)=(WAS+WASS)/2.
  310 DO 340 K=1,KMX
      TIP= 1.-(WA(I,K)**2+2.*W*ALM-(W*R(I,K))**2)/2./CP/TEMP
      IF(TIP.LT..0) GO TO 280
      TPPIP= 1.-         (2.*W*ALM-(W*R(I,K))**2)/2./CP/TEMP
      TTREL(I,K) = TPPIP*TEMP
      SMF = 0.0
      IF(I.GE.MXBL)   SMF= (SM(I,K)-SM(MXBL,K))/(SM(MX,K)-SM(MXBL,K))
      DENSTY=TIP**EXPON*RHO-(TIP/TPPIP)**EXPON*PLOSS/AR/TPPIP/TEMP*SMF
      PRS(I,K)=DENSTY*AR*TIP*TEMP
      IF(ZS(I).GE.ZH(I)) GO TO 320
      PSI = ATAN((RS(I)-RH(I))/(ZS(I)-ZH(I)))+1.5708
      GO TO 330
  320 PSI=ATAN((ZH(I)-ZS(I))/(RS(I)-RH(I)))
  330 WTHRU=WA(I,K)*CBETA(I,K)*COS(PSI-AL(I,K))
      A=XN
      IF(Z(I,K).GT.ZSPLIT) A=SFACT*XN
      C = 6.283186*R(I,K)-A*TT(I,K)
  340 AD(K)=DENSTY*WTHRU*C
      CALL INTGRL(AC(1),AD(1),KMX,WTFL(1))
      YA(I) = WTFL(KMX)/DN(I,KMX)
      YH(I) = AD(1)
      KM = (KMX+1)/2
      YM(I) = AD(KM)
      YS(I) = AD(KMX)
      IF (ABS(WT-WTFL(KMX)).LE.WTOLER) GO TO 350
      CALL CONTIN (WA(I,1),WTFL(KMX),IND,I,WT)
      IF (IND.NE.6) GO TO 290
  350 CALL SPLINT (WTFL,AC,KMX,BA,KMX,AB)
      DO 360 K=1,KMX
      DELTA=ABS(AB(K)-DN(I,K))
      DN(I,K)=(1.-CORFAC)*DN(I,K)+CORFAC*AB(K)
  360 IF(DELTA.GT.ERROR)ERROR=DELTA
  370 CONTINUE
C
C  END OF LOOP -  WEIGHT FLOW CALCULATION
C  CALCULATE STREAMLINE COORDINATES FOR NEXT ITERATION
C
      DO 380 K=2,KMXM1
      DO 390 I=1,MX
      Z(I,K)=DN(I,K)/DN(I,KMX)*(ZS(I)-ZH(I))+ZH(I)
```

```
  380 R(I,K)=DN(I,K)/DN(I,KMX)*(RS(I)-RH(I))+RH(I)
      IF (KSTH.EQ.0) GO TO 383
      DO 381 I=1,MX
      INF(I) = 0
      IF(ZS(I).EQ.ZH(I)) GO TO 3805
      AB(I) = (RS(I)-RH(I))/(ZS(I)-ZH(I))
      GO TO 381
3805   INF(I) = 1
381   CONTINUE
      DO 382 K=2,KMXM1
      DO 382 J=1,KSTH
382   CALL SMOOTH (Z(1,K),R(1,K),ZH,RH,AB,SSN,INF)
383   IF((ERROR.GE.ERROR1).OR.(ERROR.LE.TOLER)) ITER=ITER-1
      IF(ITER.GT.0) GO TO 410
      WRITE (6,1100)
      DO 400 K=1,KMX,NPRT
      WRITE (6,1080) K
      DO 390 I=1,MX
      AB(I) = Z(I,K)*COS(ANGR)  + R(I,K)*SIN(ANGR)
390   AC(I) = R(I,K)*COS(ANGR)  - Z(I,K)*SIN(ANGR)
      CALL SPLINE (AB,AC,MX,AD,CURV(1,K))
      DO 400 I=1,MX
      CURV(I,K)=CURV(I,K)/(1.+AD(I)  **2)**1.5
      B= Z(I,K)
      D= R(I,K)
  400 WRITE (6,1110)   B,D,WA(I,K),PRS(I,K),WTR(I,K),WL(I,K),TTREL(I,K)
      WRITE (6,1130)
  410 A=ERROR
      WRITE (6,1120) ITNO,A
      ITNO=ITNO+1
      IF (ITER.GE.0) GO TO 150
      N = MXBL
      DO 419 J=1,3
      K = 1
      IF (J.EQ.2) K = (KMX+1)/2
      IF (J.EQ.3) K = KMX
      IF (KD .EQ.2 ) GO TO 417
      CALL SPLINE (SM(1,K),THTA   ,MX,DTDM,AC)
      GO TO 418
417   CALL SPLDER(ZT,THTA,MT,Z(1,K),MX,DTDZ)
      DTDM(MX)= CAL(MX,K)*DTDZ(MX)
418   IF (J.EQ.1 ) BETCH = ATAN(R(MX,K)*DTDM(MX))
      IF (J.EQ.2 ) BETCM = ATAN(R(MX,K)*DTDM(MX))
      IF (J.EQ.3 ) BETCT = ATAN(R(MX,K)*DTDM(MX))
      CALL LININT (Z(MX,K),R(MX,K),XZ,XR,TN,21,21,TO(J))
419   CALL LININT (Z( N,K),R( N,K),XZ,XR,TN,21,21,TI(J))
      K = (KMX+1)/2
      DO 440 I=1,MX
      SLA = DN(I,KMX)/FLOAT(KMX-1)
      IF(ZS(I).GE.ZH(I)) GO TO 420
      PSI = ATAN((RS(I)-RH(I))/(ZS(I)-ZH(I)))+1.5708
      GO TO 430
  420 PSI = ATAN((ZH(I)-ZS(I))/(RS(I)-RH(I)))
430   AB(I) = YA(I)*SLA*COS(PSI-AL(I,1))/YH(I)
      AC(I) = YA(I)*SLA*COS(PSI-AL(I,K))/YM(I)
440   AD(I) = YA(I)*SLA*COS(PSI-AL(I,KMX))/YS(I)
      IF ( KD .EQ. 2 ) GO TO 442
      DO 441  I=1,MX
      THH(I) = THTA(I)
      THM(I) = THTA(I)
441   THS(I) = THTA(I)
      GO TO 443
```

```
442   CALL SPLINT (ZT,THTA,MT,Z(1,1   ),MX,THH)
      CALL SPLINT (ZT,THTA,MT,Z(1,K   ),MX,THM)
      CALL SPLINT (ZT,THTA,MT,Z(1,KMX),MX,THS)
443   RI = TI(1)/2.
      THHC=  THH(N)+RI*TAN(BETA(N,1))/R(N,1)
      RO = TO(1)/2.
      THH1(MX)=THH(MX)-RO*TAN(BETOH)/R(MX,1   ) -THHC
      THH2(MX)=THH(MX)-RO*TAN(BETOH)/R(MX,1   ) -THHC
      RI = TI(2)/2.
      THMC=  THM(N)+RI*TAN(BETA(N,K))/R(N,K)
      RO = TO(2)/2.
      THM1(MX)=THM(MX)-RO*TAN(BETOM)/R(MX,K   ) -THMC
      THM2(MX)=THM(MX)-RO*TAN(BETOM)/R(MX,K   ) -THMC
      RI = TI(3)/2.
      THSC=  THS(N)+RI*TAN(BETA(N,KMX))/R(N,KMX)
      RO = TO(3)/2.
      THS1(MX)=THS(MX)-RO*TAN(BETOT)/R(MX,KMX) -THSC
      THS2(MX)=THS(MX)-RO*TAN(BETOT)/R(MX,KMX) -THSC
      DO 449 I=1,MXBL
      THH1(I) = 0.0
      THM1(I) = 0.0
      THS1(I) = 0.0
      THH2(I) = 0.0
      THM2(I) = 0.0
449   THS2(I) = 0.0
      N1 = N+1
      N2 = MX-1
      DO 450 I=N1,N2
      THH1(I)  = THH(I)+      TT(I,1)/2./R(I,1)-THHC
      THM1(I)  = THM(I)+      TT(I,K)/2./R(I,K) -THMC
      THS1(I)  = THS(I)+      TT(I,KMX   )/2./R(I,KMX   )  -THSC
      THH2(I)  = THH(I)-      TT(I,1)/2./R(I,1)-THHC
      THM2(I)  = THM(I)-      TT(I,K)/2./R(I,K) -THMC
450   THS2(I)  = THS(I)-      TT(I,KMX   )/2./R(I,KMX   )  -THSC
      WRITE(6,1200)
      WRITE(6,1239)
      WRITE(6,1251)
      DO 451 I=1,MX
451   AE(I) = SM(I,1   )-SM(MXBL,1   )
      WRITE(6,1230)(AE(I),I=1,MX)
      WRITE(6,1249)
      WRITE(6,1230)( R(I,1   ),I=1,MX)
      WRITE(6,1240)
      WRITE(6,1230)( AB(I),I=1,MX)
      WRITE(6,1250)
      WRITE(6,1251)
      WRITE(6,1230)(AE(I),I=1,MX)
      WRITE(6,1252)
      WRITE(6,1230)(THH1(I),I=1,MX)
      WRITE(6,1253)
      WRITE(6,1230)(THH2(I),I=1,MX)
      RI = TI(1)/2.
      RO = TO(1)/2.
      STGR = THH1(MX)
      WRITE(6,1254) STGR,RI,RO
      IF (Z(MX,1 ) .LT. ZSPLIT) GO TO 453
      CALL SPLINT (Z(1,1   ),SM(1,1   ),MX,ZSPLIT,1,AMLER)
      CALL SPLINT (SM(1,1   ),R(1,1   ),MX,AMLER,1,RSPLIT)
      CALL SPLINT (SM(1,1   ),THH,MX,AMLER,1,THTAS)
      CALL SPLDER(SM(1,1   ),THH,MX,AMLER,1,DTDMS)
      CALL LININT(ZSPLIT,RSPLIT,XZ,XR,TN,21,21,TSPLIT)
      TANS = RSPLIT*DTDMS
```

```
      RIS = TSPLIT/2.0
      STGRS = THM1(MX)-RIS*TANS/RSPLIT-THTAS +THHC
      AMLER=AMLER-SM(MXBL,1)
      BETAS = ATAN(TANS)
      BETAS = BETAS*57.29577
      WRITE(6,1255) AMLER,STGRS,RIS,RO,BETAS
453   WRITE(6,1201)
      WRITE(6,1239)
      WRITE(6,1251)
      DO 454 I=1,MX
454   AE(I) = SM(I,K  )-SM(MXBL,K  )
      WRITE(6,1230)(AE(I),I=1,MX)
      WRITE(6,1249)
      WRITE(6,1230)( R(I,K  ),I=1,MX)
      WRITE(6,1240)
      WRITE(6,1230)( AC(I),I=1,MX)
      WRITE(6,1250)
      WRITE(6,1251)
      WRITE(6,1230)(AE(I),I=1,MX)
      WRITE(6,1252)
      WRITE(6,1230)(THM1(I),I=1,MX)
      WRITE(6,1253)
      WRITE(6,1230)(THM2(I),I=1,MX)
      RI = TI(2)/2.
      RO = TO(2)/2.
      STGR = THM1(MX)
      WRITE(6,1254) STGR,RI,RO
      IF (Z(MX,K ) .LT. ZSPLIT) GO TO 456
      CALL SPLINT (Z(1,K  ),SM(1,K  ),MX,ZSPLIT,1,AMLER)
      CALL SPLINT (SM(1,K  ),R(1,K  ),MX,AMLER,1,RSPLIT)
      CALL SPLINT (SM(1,K  ),THM,MX,AMLER,1,THTAS)
      CALL SPLDER(SM(1,K  ),THM,MX,AMLER,1,DTDMS)
      CALL LININT(ZSPLIT,RSPLIT,XZ,XR,TN,21,21,TSPLIT)
      TANS = RSPLIT*DTDMS
      RIS = TSPLIT/2.0
      STGRS = THM1(MX)-RIS*TANS/RSPLIT-THTAS +THMC
      AMLER=AMLER-SM(MXBL,K)
      BETAS = ATAN(TANS)
      BETAS = BETAS*57.29577
      WRITE(6,1255) AMLER,STGRS,RIS,RO,BETAS
456   WRITE(6,1202)
      WRITE(6,1239)
      WRITE(6,1251)
      DO 457 I=1,MX
457   AE(I) = SM(I,KMX)-SM(MXBL,KMX)
      WRITE(6,1230)(AE(I),I=1,MX)
      WRITE(6,1249)
      WRITE(6,1230)( R(I,KMX),I=1,MX)
      WRITE(6,1240)
      WRITE(6,1230)( AC(I),I=1,MX)
      WRITE(6,1250)
      WRITE(6,1251)
      WRITE(6,1230)(AE(I),I=1,MX)
      WRITE(6,1252)
      WRITE(6,1230)(THS1(I),I=1,MX)
      WRITE(6,1253)
      WRITE(6,1230)(THS2(I),I=1,MX)
      RI = TI(3)/2.
      RO = TO(3)/2.
      STGR = THS1(MX)
      WRITE(6,1254) STGR,RI,RO
      IF (Z(MX,KMX).LT. ZSPLIT) GO TO 459
```

```
      CALL SPLINT (Z(1,KMX),SM(1,KMX),MX,ZSPLIT,1,AMLER)
      CALL SPLINT (SM(1,KMX),R(1,KMX),MX,AMLER,1,RSPLIT)
      CALL SPLINT (SM(1,KMX),THS,MX,AMLER,1,THTAS)
      CALL SPLDER(SM(1,KMX),THS,MX,AMLER,1,DTDMS)
      CALL LININT(ZSPLIT,RSPLIT,XZ,XR,TN,21,21,TSPLIT)
      TANS = RSPLIT*DTDMS
      RIS = TSPLIT/2.0
      STGRS = THS1(MX)-RIS*TANS/RSPLIT-THTAS +THSC
      AMLER=AMLER-SM(MXBL,KMX)
      BETAS = ATAN(TANS)
      BETAS = BETAS*57.29577
      WRITE(6,1255) AMLER,STGRS,RIS,RO,BETAS
459   DO 460 J=1,3
      I = MXBL
      K=1
      IF(J.EQ.2) K=(KMX+1)/2
      IF(J.EQ.3) K=KMX
      T1P= 1.-(WA(I,K)**2+2.*W*ALM-(W*R(I,K))**2)/2./CP/TEMP
      DENSTY = T1P**EXPON*RHO
      C = 6.283186*R(I,K)-XN*TT(I,K)
      WIDTH = AB(MXBL)
      IF(J.EQ.2) WIDTH = AC(MXBL)
      IF(J.EQ.3) WIDTH = AD(MXBL)
      WM = BA(2)/DENSTY/C/WIDTH
      WTHETA = ALM/R(I,K)-W*R(I,K)
      BETAI(J) = ATAN(WTHETA/WM)
      AA(J) = BETAI(J)*57.29577
  460 CONTINUE
      WRITE (6,1170) AA
      GO TO 10
 1010 FORMAT (4I5,6F10.4)
 1020 FORMAT (8HORUN NO.I3,10X,25HINPUT DATA CARD LISTING    )
 1030 FORMAT (7F10.4)
 1040 FORMAT (10X24HBCD CARDS FOR DN,WA,Z,R   )
 1050 FORMAT (36HK     STAG. SPEED OF SOUND AT INLET = ,F9.2)
 1060 FORMAT (///5X13HITERATION NO.I3)
 1070 FORMAT (1H 6X5HALPHA9X5HRC    9X5HSM    9X5HBETA 9X5HTT    9X5HSA    9
     1X5HSB    9X5HSC    9X5HSD    )
 1080 FORMAT (2X10HSTREAMLINEI3)
 1090 FORMAT (9F14.6)
 1100 FORMAT (1HL9X5H    Z15X5H   R   15X5HWA    15X5HPRESS14X3HWTR14X3HWL
     114X6HTTREL )
 1110 FORMAT (6F19.6,F18.6)
 1120 FORMAT (18H      ITERATION NO. I3,10X,24HMAX. STREAMLINE CHANGE = ,
     1F10.6)
 1130 FORMAT (1HJ)
 1160 FORMAT (12F11.4)
 1170 FORMAT (///1HL,10X,20HINLET ANGLES   -   HUB,F7.2,8H,    MEANF7.2,10H
     1,    SHROUDF7.2)
1001  FORMAT(80H
     1                                       )
1007  FORMAT(1H0,3X,2HMX,2X,3HKMX,3X,2HMR,3X,2HMZ,6X,1HW,14X,2HWT,
     113X,2HXN,12X,3HGAM,12X,2HAR)
1008  FORMAT(1H0,1X,4HTYPE,1X,4H MT ,2X,3HSRW,1X,4HMXBL,5X,4HTEMP,
     111X,3HALM,12X,3HRHO,12X,5HPLOSS,9X,4HANGR)
1009  FORMAT(1H0,1X,4HKSTH,1X,4HNPRT,1X,4HITER,1X,4H KD ,4X,5HSFACT,
     19X,6HZSPLIT,10X,4HBETO,11X,6HCORFAC,9X,3HSSN)
1011  FORMAT (4I5,6G15.5)
1027  FORMAT(1H0,4HRB =,F8.5)
1028  FORMAT (7G15.5)
1029  FORMAT(1H0,5X,8HZS ARRAY)
1031  FORMAT(1H0,5X,8HZH ARRAY)
```

```
1032  FORMAT(1H0,5X,8HRS ARRAY)
1033  FORMAT(1H0,5X,8HRH ARRAY)
1034  FORMAT(1H0,5X,10HTHTA ARRAY)
1035  FORMAT(1H0,15X,21HBLADE THICKNESS TABLE)
1036  FORMAT(1H0,5X,8HTN ARRAY)
1037  FORMAT(1H0,5X,7HZ ARRAY)
1038  FORMAT(1H0,5X,7HR ARRAY)
1039  FORMAT(1H0,5X,8HZT ARRAY)
1049  FORMAT(1H1)
1200  FORMAT(1H1,20X,3HHUB)
1201  FORMAT(1H1,20X,4HMEAN)
1202  FORMAT(1H1,20X,6HSHROUD)
1230  FORMAT(1H ,8G16.7)
1239  FORMAT(1H0,26HSTREAM-CHANNEL COORDINATES)
1240  FORMAT(1H0,31HSTREAM-CHANNEL NORMAL THICKNESS)
1249  FORMAT(1H0,7HR ARRAY)
1250  FORMAT(1H0,17HBLADE COORDINATES)
1251  FORMAT(1H0,7HM ARRAY)
1252  FORMAT(1H0,27HTHETA ARRAY BLADE SURFACE 1)
1253  FORMAT(1H0,27HTHETA ARRAY BLADE SURFACE 2)
1254  FORMAT(1H0,6HSTGR =,G13.5,3X,4HRI =,G13.5,3X,4HRO =,G13.5)
1255  FORMAT(1H0,9HSPLITTERS,4X,6HMLER =,G13.5,4X,7HSTGRS =,G13.5,4X,
     14HRI =,G13.5,4X,4HRO =,G13.5,4X,7HBETAS =,G13.5)
      END



$IBFTC RUUT    DECK

      SUBROUTINE RUUT(A,B,Y,X,SM,R,MX)
C
C  ROOT FINDS A ROOT FOR (FX-Y) IN THE INTERVAL (A,B)
C
      COMMON SRW
      INTEGER SRW
      DIMENSION SM(21),R(21)
      TOLERY = Y/50000.
      IF (SRW.EQ.21) WRITE(6,1000) A,B,Y,TOLERY
       X1 = A
      CALL SPLINT (SM,R,MX,X1,1,FX1)
      IF (SRW.EQ.21) WRITE(6,1010) X1,FX1
      X2 = B
10     DO 30 I=1,15
      X = (X1+X2)/2.
      CALL SPLINT (SM,R,MX,X,1,FX)
      IF (SRW.EQ.21) WRITE(6,1010) X,FX
      IF ((FX1-Y)*(FX-Y).GT.0.) GO TO 20
      X2 = X
      GO TO 30
20    X1 = X
      FX1 = FX
30    CONTINUE
      IF(ABS(Y-FX).LT.TOLERY) RETURN
      WRITE (6,1020)A,B,Y,FX,X
      RETURN
1000  FORMAT (32H1INPUT ARGUMENTS FOR ROOT -- A =G13.5,3X,3HB =,G13.5,
     1   3X,3HY =,G13.5,3X,8HTOLERY =, G13.5/16X,1HX,17X,2HFX)
1010  FORMAT(8X,G16.5,4G18.5)
1020   FORMAT(21HROOT OUT OF TOLERANCE,2X,3HA =,G16.5,2X,3HB =,G16.5,2X,
     13HY =,G16.5,2X,4HFX =,G16.5,2X,3HX =,G16.5)
      END
```

```
$IBFTC SMOOTH  DECK

      SUBROUTINE SMOOTH (X,Y,XH,YH,SLOPE,SSN,INF)
      DIMENSION X(21),Y(21),XH(21),YH(21),X1(21),Y1(21),INF(21),
     1SLOPE(21)
      NS = SSN
       N1=NS-1
      DO 10 I =2,N1
      D=2.0
      IF(I.EQ.(NS-1)) D=8.0
      IF(I.EQ.(NS-2))D=4.0
      IF(I.EQ.(NS-3)) D=2.6667
      IF(X(I+1).EQ.X(I-1)) GO TO 5
      SLOPE1 = (Y(I+1)-Y(I-1))/(X(I+1)-X(I-1))
      IF (INF(I).EQ. 1 ) GO TO 6
      X1(I) = ((Y(I-1)-SLOPE1*X(I-1))-(YH(I)-SLOPE(I)*XH(I)))/(SLOPE(I)
     1-SLOPE1)
      X1(I)=((X1(I)-X(I))/D)+X(I)
      Y1(I) = YH(I)+SLOPE(I)*(X1(I)-XH(I))
      GO TO 10
C
C    SLOPE1 = INFINITY
5     Y1(I) =SLOPE(I)*(X(I-1)-XH(I))+YH(I)
      Y1(I)=((Y1(I)-Y(I))/D)+Y(I)
      X1(I)=((X(I-1)-X(I))/D)+X(I)
      GO TO 10
C
C    SLOPE = INFINITY
6     Y1(I) = SLOPE1*(X(I)-X(I-1))+Y(I-1)
      Y1(I)=((Y1(I)-Y(I))/D)+Y(I)
      X1(I) = X(I)
10    CONTINUE
      DO 20 I=2,N1
      X(I) = X1(I)
20    Y(I) = Y1(I)
      RETURN
      END
```

```
$IBFTC INTGRL  DECK

      SUBROUTINE INTGRL (X,Y,N,SUM)
      DIMENSION X(50),Y(50),S(50),A(50),B(50),C(50),F(50),W(50),SB(50),
     1G(50),EM(50),SUM(50)
      COMMON SRW
      INTEGER SRW
      DO 10 I=2,N
   10 S(I)=X(I)-X(I-1)
      NO=N-1
      DO 20 I=2,NO
      A(I)=S(I)/6.0
      B(I)=(S(I)+S(I+1))/3.0
      C(I)=S(I+1)/6.0
   20 F(I)=(Y(I+1)-Y(I))/S(I+1)-(Y(I)-Y(I-1))/S(I)
      A(N)=-.5
      B(1)=1.0
      B(N)=1.0
      C(1)=-.5
```

```
      F(1)=0.0
      F(N)=0.0
      W(1)=B(1)
      SB(1)=C(1)/W(1)
      G(1)=0.0
      DO 30 I=2,N
      W(I)=B(I)-A(I)*SB(I-1)
      SB(I)=C(I)/W(I)
   30 G(I)=(F(I)-A(I)*G(I-1))/W(I)
      EM(N)=G(N)
      DO 40 I=2,N
      K=N+1-I
   40 EM(K)=G(K)-SB(K)*EM(K+1)
      SUM(1) =0.0
      DO 50 K=2,N
   50 SUM(K) = SUM(K-1)+S(K)*(Y(K)+Y(K-1))/2.0-S(K)**3*(EM(K)+EM(K-1))/2
     14.0
      IF(SRW.EQ.17) WRITE(6,1000) N,(X(I),Y(I),SUM(I),EM(I),I=1,N)
      RETURN
 1000 FORMAT (17HK NO. OF POINTS =I3/10X5HX     15X5HY     15X5HSUM
     1    13X10H2ND DERIV./(4E20.8))
      END



$IBFTC CONTIN  DECK

      SUBROUTINE CONTIN (WA,WTFL,IND,I,WT)
      DIMENSION SPEED(3),WEIGHT(3)
  135 GO TO (140,150,210,270,370),IND
  140 SPEED(1) = WA
      WEIGHT(1) = WTFL
      DELTA = WT/WTFL*WA-WA
      IF(ABS(DELTA).GT.100.) DELTA = SIGN(100.,DELTA)
      WA = DELTA+WA
      IND = 2
      RETURN
  150 IF ((WTFL-WEIGHT(1))/(WA-SPEED(1))) 180,180,160
  160 SPEED(2) = WA
      DELTA = (WT-WTFL)/(WTFL-WEIGHT(1))*(WA-SPEED(1))
      IF(ABS(DELTA).GT.100.) DELTA = SIGN(100.,DELTA)
      WA = DELTA+WA
  166 SPEED(1) = SPEED(2)
      WEIGHT(1) = WTFL
      RETURN
  170 WRITE (6,1000) I,WTFL
      IND = 6
      RETURN
  180 IND = 3
      IF (WTFL.GE.WT) GO TO 140
      IF (SPEED(1)-WA) 190,200,200
  190 SPEED(2) = SPEED(1)
      SPEED(1) = 2.0*SPEED(1)-WA
      SPEED(3) = WA
      WEIGHT(2) = WEIGHT(1)
      WEIGHT(3) = WTFL
      WA = SPEED(1)
      RETURN
```

```
  200 SPEED(2) = WA
      SPEED(3) = SPEED(1)
      SPEED(1) = 2.0*WA-SPEED(1)
      WEIGHT(2) = WTFL
      WEIGHT(3) = WEIGHT(1)
      WA = SPEED(1)
      RETURN
  210 WEIGHT(1) = WTFL
      IF (WTFL.GE.WT) GO TO 140
      IF (WEIGHT(1)-WEIGHT(2)) 230,380,220
  220 WEIGHT(3) = WEIGHT(2)
      WEIGHT(2) = WEIGHT(1)
      SPEED(3) = SPEED(2)
      SPEED(2) = SPEED(1)
      SPEED(1) = 2.0*SPEED(2)-SPEED(3)
      WA = SPEED(1)
      RETURN
  230 IF (SPEED(3)-SPEED(1)-10.0) 170,170,240
  240 INC = 4
  245 IF (WEIGHT(3)-WEIGHT(1)) 260,260,250
  250 WA = (SPEED(1)+SPEED(2))/2.0
      RETURN
  260 WA = (SPEED(3)+SPEED(2))/2.0
      RETURN
  270 IF (SPEED(3)-SPEED(1)-10.0) 170,170,280
  280 IF (WTFL-WEIGHT(2)) 320,350,290
  290 IF (WA-SPEED(2)) 310,300,300
  300 SPEED(1) = SPEED(2)
      SPEED(2) = WA
      WEIGHT(1) = WEIGHT(2)
      WEIGHT(2) = WTFL
      GO TO 245
  310 SPEED(3) = SPEED(2)
      SPEED(2) = WA
      WEIGHT(3) = WEIGHT(2)
      WEIGHT(2) = WTFL
      GO TO 245
  320 IF (WA-SPEED(2)) 340,330,330
  330 WEIGHT(3) = WTFL
      SPEED(3) = WA
      GO TO 245
  340 WEIGHT(1) = WTFL
      SPEED(1) = WA
      GO TO 245
  350 INC = 5
      IF (WA-SPEED(2)) 380,360,360
  360 SPEED(1) = SPEED(2)
      WEIGHT(1) = WEIGHT(2)
      SPEED(2) = (SPEED(1)+SPEED(3))/2.0
      WA = SPEED(2)
      RETURN
  370 INC = 4
      WEIGHT(2) = WTFL
      WA = (SPEED(1)+SPEED(2))/2.0
      RETURN
  380 INC = 5
  390 WEIGHT(3) = WEIGHT(2)
      SPEED(3) = SPEED(2)
      SPEED(2) = (SPEED(1)+SPEED(3))/2.
      WA = SPEED(2)
      RETURN
 1000 FORMAT (/12H FIXED LINE I2,12H,  MAX WT = F10.6)
      END
```

```
$IBFTC SPLDER   DECK

      SUBROUTINE SPLDER(X,Y,N,Z,MAX,DYDX)
      DIMENSION X(50),Y(50),S(50),A(50),B(50),C(50),F(50),W(50),SB(50),
     1G(50),EM(50),Z(50),DYDX(50)
      DO 10 I=2,N
   10 S(I)=X(I)-X(I-1)
      NO=N-1
      DO 20 I=2,NO
      A(I)=S(I)/6.0
      B(I)=(S(I)+S(I+1))/3.0
      C(I)=S(I+1)/6.0
   20 F(I)=(Y(I+1)-Y(I))/S(I+1)-(Y(I)-Y(I-1))/S(I)
      A(N)=-.5
      B(1)=1.0
      B(N)=1.0
      C(1)=-.5
      F(1)=0.0
      F(N)=0.0
      W(1)=B(1)
      SB(1)=C(1)/W(1)
      G(1)=0.0
      DO 30 I=2,N
      W(I)=B(I)-A(I)*SB(I-1)
      SB(I)=C(I)/W(I)
   30 G(I)=(F(I)-A(I)*G(I-1))/W(I)
      EM(N)=G(N)
      DO 40 I=2,N
      K=N+1-I
   40 EM(K)=G(K)-SB(K)*EM(K+1)
      DO 90 I=1,MAX
      K=2
      IF(Z(I)-X(1)) 60,70,70
   60 WRITE (6,1000)Z(I)
 1000 FORMAT (17H OUT OF BLADE Z =F10.6)
      GO TO 85
   65 WRITE (6,1000)Z(I)
      K=N
      GO TO 85
   70 IF(Z(I)-X(K)) 85,85,80
   80 K=K+1
      IF(K-N) 70,70,65
   85 DYDX(I)=-EM(K-1)*(X(K)-Z(I))**2/2.0/S(K)+EM(K)*(X(K-1)-Z(I))**2/2.
     10/S(K)+(Y(K)-Y(K-1))/S(K)-(EM(K)-EM(K-1))*S(K)/6.0
   90 CONTINUE
  100 RETURN
      END
```

```
$IBFTC SPLINE   DECK

      SUBROUTINE SPLINE (X,Y,N,SLOPE,EM)
      DIMENSION X(50),Y(50),S(50),A(50),B(50),C(50),F(50),W(50),SB(50),
     1G(50),EM(50),SLOPE(50)
      COMMON Q
      INTEGER Q
      DO 10 I=2,N
```

```
   10 S(I)=X(I)-X(I-1)
      NO=N-1
      DO 20 I=2,NO
      A(I)=S(I)/6.
      B(I)=(S(I)+S(I+1))/3.
      C(I)=S(I+1)/6.
   20 F(I)=(Y(I+1)-Y(I))/S(I+1)-(Y(I)-Y(I-1))/S(I)
      A(N)=-.5
      B(1)=1.
      B(N)=1.
      C(1)=-.5
      F(1)=0.
      F(N)=0.
      W(1)=B(1)
      SB(1)=C(1)/W(1)
      G(1)=0.
      DO 30 I=2,N
      W(I)=B(I)-A(I)*SB(I-1)
      SB(I)=C(I)/W(I)
   30 G(I)=(F(I)-A(I)*G(I-1))/W(I)
      EM(N)=G(N)
      DO 40 I=2,N
      K=N+1-I
   40 EM(K)=G(K)-SB(K)*EM(K+1)
      SLOPE(1)=-S(2)/6.*(2.*EM(1)+EM(2))+(Y(2)-Y(1))/S(2)
      DO50 I=2,N
   50 SLOPE(I)=S(I)/6.*(2.*EM(I)+EM(I-1))+(Y(I)-Y(I-1))/S(I)
      IF (Q.EQ.13) WRITE (6,100) N,(X(I),Y(I),SLOPE(I),EM(I),I=1,N)
  100 FORMAT (2X15HNO. OF POINTS =I3/10X5HX     15X5HY     15X5HSLOPE15X5H
     1EM    /(4F20.8))
      RETURN
      END
```

```
$IBFTC LININT  DECK
      SUBROUTINE LININT(X1,Y1,X,Y,TN,MX,MY,F)
      COMMON K
      DIMENSION X(MX),Y(MY),TN(MX,MY)
      DO 10 J3=1,MX
   10 IF(X1.LE.X(J3))GO TO 20
      J3=MX
   20 DO 30 J4=1,MY
   30 IF(Y1.LE.Y(J4))GO TO 40
      J4=MY
   40 J1=J3-1
      J2=J4-1
      EPS1=(X1-X(J1))/(X(J3)-X(J1))
      EPS2=(Y1-Y(J2))/(Y(J4)-Y(J2))
      EPS3=1.-EPS1
      EPS4=1.-EPS2
      F=TN(J1,J2)*EPS3*EPS4+TN(J3,J2)*EPS1*EPS4+TN(J1,J4)*EPS2*EPS3+
     1TN(J3,J4)*EPS1*EPS2
      IF(K.EQ.14) WRITE(6,1)X1,Y1,F,J1,J2,EPS1,EPS2
    1 FORMAT (8H  LININT3F10.5,2I3,2F10.5)
      K=0
      RETURN
      END
```

```
$IBFTC SPLINT  DECK
      SUBROUTINE SPLINT (X,Y,N,Z,MAX,YINT)
      DIMENSION X(50),Y(50),S(50),A(50),B(50),C(50),F(50),W(50),SB(50),
     1G(50),EM(50),Z(50),YINT(50)
      COMMON G
      INTEGER G
      DO 10 I=2,N
   10 S(I)=X(I)-X(I-1)
      NO=N-1
      DO 20 I=2,NO
      A(I)=S(I)/6.0
      B(I)=(S(I)+S(I+1))/3.0
      C(I)=S(I+1)/6.0
   20 F(I)=(Y(I+1)-Y(I))/S(I+1)-(Y(I)-Y(I-1))/S(I)
      A(N)=-.5
      B(1)=1.0
      B(N)=1.0
      C(1)=-.5
      F(1)=0.0
      F(N)=0.0
      W(1)=B(1)
      SB(1)=C(1)/W(1)
      G(1)=0.0
      DO 30 I=2,N
      W(I)=B(I)-A(I)*SB(I-1)
      SB(I)=C(I)/W(I)
   30 G(I)=(F(I)-A(I)*G(I-1))/W(I)
      DO 40 I=2,N
      K=N+1-I
   40 EM(K)=G(K)-SB(K)*EM(K+1)
      DO 90 I=1,MAX
      K=2
      IF(Z(I)-X(1)) 60,50,70
   50 YINT(I)=Y(1)
      GO TO 90
   60 IF(Z(I).LT.(1.1*X(1)-.1*X(2)))WRITE (6,1000)Z(I)
      GO TO 85
 1000 FORMAT (17H OUT OF RANGE Z =F10.6)
   65 IF(Z(I).GT.(1.1*X(N)-.1*X(N-1))) WRITE (6,1000)Z(I)
      K=N
      GO TO 85
   70 IF(Z(I)-X(K)) 85,75,80
   75 YINT(I)=Y(K)
      GO TO 90
   80 K=K+1
      IF(K-N) 70,70,65
   85 YINT(I) = EM(K-1)*(X(K)-Z(I))**3/6./S(K)+EM(K)*(Z(I)-X(K-1))**3/6.
     1/S(K)+(Y(K)/S(K)-EM(K)*S(K)/6.)*(Z(I)-X(K-1))+(Y(K-1)/S(K)-EM(K-1)
     2*S(K)/6.)*(X(K)-Z(I))
   90 CONTINUE
      EM(N)=G(N)
```

```
      MXA = MAXO(N,MAX)
      IF(C.EQ.16) WRITE(6,1010) N,MAX,(X(I),Y(I),Z(I),YINT(I),I=1,MXA)
 1010 FORMAT (2X21HNO. OF POINTS GIVEN =,I3,30H, NO. OF INTERPOLATED POI
     1NTS =,I3,/10X5HX     15X5HY     12X11HX-INTERPOL.9X11HY-INTERPOL./(4
     2E20.8))
  100 RETURN
      END
```

Lewis Research Center,

National Aeronautics and Space Administration,

Cleveland, Ohio, November 16, 1971,

132-15.

APPENDIX - SYMBOLS

A	coefficient, eq. (2)
B	coefficient, eq. (2)
C	coefficient, eq. (2)
c_p	specific heat, J/(kg)(K); (ft)(lbf)/(slug)($^{\circ}$R)
D	coefficient, eq. (2)
h	enthalpy, J/kg; (ft)(lbf)/slug
m	meridional streamline distance, m; ft
N	number of blades
$\Delta p''$	loss in relative total pressure, N/m^2; lb/ft^2
R	gas constant, J/(kg)(K); (ft)(lbf)/(slug)($^{\circ}$R)
r	radius from axis of rotation, m; ft
r_c	radius of curvature of a meridional streamline, m; ft
s	distance along a quasi-orthogonal, m; ft
T	temperature, K, $^{\circ}$R
t_n	blade thickness normal to mean blade shape, m; ft
t_θ	blade thickness in tangential direction, m; ft
V	absolute velocity, m/sec; ft/sec
W	relative velocity, m/sec; ft/sec
w	mass flow, kg/sec; slugs/sec
z	axial distance, m; ft
α	angle between meridional streamline and z-axis, rad
β	angle between relative velocity and meridional plane, rad
γ	ratio of specific heats
θ	angular coordinate, rad
λ	inlet prerotation, m^2/sec; ft^2/sec
ρ	density, kg/m^3; slugs/ft^3
ψ	angle between quasi-orthogonal and radial direction, rad
ω	rotational speed, rad/sec

Subscripts:

b	point at which mean stream surface deviates from mean blade shape
f	flow
h	hub
i	inlet
isen	isentropic
l	leading surface
m	direction of meridional streamline
n	normal direction
o	outlet
s	shroud
t	trailing surface
θ	tangential direction

Superscripts:

$'$	absolute total conditions
$''$	relative total conditions

REFERENCES

1. Katsanis, Theodore: Use of Arbitrary Quasi-Orthogonals For Calculating Flow Distribution In the Meridional Plane Of A Turbomachine. NASA TN D-2546, 1964.
2. Katsanis, Theodore; and McNally, William D.: Fortran Program For Calculating Velocities And Streamlines On A Blade-To-Blade Stream Surface Of A Tandem Blade Turbomachine. NASA TN D-5044, 1969.
3. Katsanis, Theodore; and McNally, William D.: Revised Fortran Program For Calculating Velocities And Streamlines On A Blade-To-Blade Stream Surface Of A Turbomachine. NASA TM X-1764, 1969.
4. Katsanis, Theodore: Fortran Program For Calculating Transonic Velocities On A Blade-To-Blade Stream Surface Of A Turbomachine. NASA TN D-5427, 1969.
5. Stanitz, John D.; and Prian, Vasily D.: A Rapid Approximate Method For Determining Velocity Distribution On Impeller Blades Of Centrifugal Compressors. NACA TN 2421, 1951.

TABLE I. - INPUT FORM FOR QUAC

5	10	15	20	21 30	31 40	41 50	51 60	61 70
TITLE								
MX	KMX	MR	MZ	W	WT	XN	GAM	AR
TYPE	MT	SRW	MXBL	TEMP	ALM	RHO	PLOSS	ANGR
KSTH	NPRT	ITER	KD	SFACT	ZSPLIT	BETO	CORFAC	SSN
ZS ARRAY								
ZH ARRAY								
RS ARRAY								
RH ARRAY								
THTA ARRAY								
ZT ARRAY (IF KD = 2, ZT ARRAY USED. THTA = f(ZT). MT = No. of.)								
TN ARRAY								
XZ ARRAY								
XR ARRAY								

TABLE II. - SAMPLE OUTPUT

1

```
      IMPELLER 4ME S-IA3 B

 RUN NO.  1           INPUT DATA CARD LISTING

   MX  KMX   MR   MZ       W                WT            XN            GAM            AR
   13   21    5    6    7853.98         0.62160E-01    15.0000        1.40000        1716.20

  TYPE  MT   SRW MXBL     TEMP            ALM            RHO           PLOSS          ANGR
   -0   -0   -0    4    518.700        -0             0.23770E-02    1622.80        45.0000

  KSTH NPRT ITER  KD     SFACT           ZSPLIT          BETO          CORFAC          SSN
    4    5    1   -0    2.00000         0.12500       -53.9500        0.10000        8.00000

      ZS ARRAY
  -0.75430E-01   -0.50000E-01   -0.25000E-01   0              0.99208E-02    0.39682E-01    0.72917E-01
   0.97583E-01    0.13142        0.16958       0.18258        0.18892        0.19167

      ZH ARRAY
  -0.75430E-01   -0.50000E-01   -0.25000E-01   0              0.10167E-01    0.54000E-01    0.12083
   0.15575        0.18433        0.20100       0.20475        0.20589        0.20617

      RS ARRAY
   0.14125        0.14125        0.14125       0.14125        0.14125        0.14135        0.14320
   0.14692        0.15712        0.18291       0.20487        0.22757        0.25004

      RH ARRAY
   0.42375E-01    0.42375E-01    0.42375E-01   0.42375E-01    0.42375E-01    0.43229E-01    0.60442E-01
   0.85655E-01    0.12620        0.17845       0.20711        0.22771        0.25004

      THTA ARRAY
   1.08290        0.69770        0.33800       0             -0.12500       -0.50000       -0.90000
  -1.10000       -1.30000       -1.50000      -1.60000       -1.67500       -1.75000

                    BLADE THICKNESS TABLE

      TN ARRAY
   0.41670E-02    0.50000E-02    0.58330E-02   0.75000E-02    0.75000E-02    0.75000E-02
   0.31250E-02    0.37500E-02    0.43750E-02   0.58330E-02    0.75000E-02    0.75000E-02
   0.20830E-02    0.20830E-02    0.22500E-02   0.33330E-02    0.50000E-02    0.58330E-02
  -0             -0             -0            -0              0.20830E-02    0.54170E-02
  -0             -0             -0            -0             -0              0.50000E-02

      Z ARRAY
  -0.83300E-03    0.47250E-01    0.94450E-01   0.14170        0.18333        0.20833

      R ARRAY
   0.41670E-01    0.91580E-01    0.14125       0.20833        0.25417
```

2

```
     STAG. SPEED OF SOUND AT INLET =   1116.36

 RR = 0.21747
     ITERATION NO.  1          MAX. STREAMLINE CHANGE =   0.018577
     ITERATION NO.  2          MAX. STREAMLINE CHANGE =   0.015987
     ITERATION NO.  3          MAX. STREAMLINE CHANGE =   0.014245
     ITERATION NO.  4          MAX. STREAMLINE CHANGE =   0.012898
     ITERATION NO.  5          MAX. STREAMLINE CHANGE =   0.011650
     ITERATION NO.  6          MAX. STREAMLINE CHANGE =   0.010522
     ITERATION NO.  7          MAX. STREAMLINE CHANGE =   0.009499
     ITERATION NO.  8          MAX. STREAMLINE CHANGE =   0.008590
     ITERATION NO.  9          MAX. STREAMLINE CHANGE =   0.007747
     ITERATION NO. 10          MAX. STREAMLINE CHANGE =   0.006983
     ITERATION NO. 11          MAX. STREAMLINE CHANGE =   0.006293
     ITERATION NO. 12          MAX. STREAMLINE CHANGE =   0.005668
     ITERATION NO. 13          MAX. STREAMLINE CHANGE =   0.005104
     ITERATION NO. 14          MAX. STREAMLINE CHANGE =   0.004611
     ITERATION NO. 15          MAX. STREAMLINE CHANGE =   0.004148
     ITERATION NO. 16          MAX. STREAMLINE CHANGE =   0.003731
     ITERATION NO. 17          MAX. STREAMLINE CHANGE =   0.003354
     ITERATION NO. 18          MAX. STREAMLINE CHANGE =   0.003015
     ITERATION NO. 19          MAX. STREAMLINE CHANGE =   0.002709
     ITERATION NO. 20          MAX. STREAMLINE CHANGE =   0.002434
```

TABLE II. - Continued. SAMPLE OUTPUT

3

ITERATION NO. 67

ALPHA	RC	SM	BETA	TT	SA	SB	SC	SD
STREAMLINE 1								
-0.001619	0.001666	0.	-33.192096	0.	-7.071380	8599.316772	-0.000200	-1731.773972
0.002022	0.003332	0.025430	-32.066519	0.	-6.649137	8339.272705	0.000235	-5723.242554
-0.006425	-0.015122	0.050430	-30.721272	0.	-6.169999	8024.158325	-0.000692	5073.781799
0.023697	0.057244	0.075430	-28.558731	0.004744	-5.349131	7509.221313	0.002212	-1434.670074
-0.063958	-0.357136	0.085597	-26.058385	0.004834	-4.842164	6899.353760	-0.005405	-7692.848938
4.145020	4.201212	0.129438	-16.202538	0.005290	2.081720	4206.509583	-0.150864	-1777.882187
26.741523	7.103331	0.198449	-16.851673	0.006449	4.734078	3789.024353	-2.385294	-1276.558319
44.955225	7.187098	0.241520	-20.260381	0.006957	3.757438	4238.034607	-3.751571	-1323.671707
64.044206	5.886395	0.291126	-25.751180	0.006422	1.758786	4482.935120	-3.613119	-1574.753906
79.549380	3.979333	0.345971	-31.632465	0.005817	0.483932	4048.283081	-2.623682	-518.020050
85.266557	2.981542	0.374875	-36.617331	0.006209	0.172996	-2045.489014	-2.089269	743.802719
88.218536	2.065947	0.395507	-41.230254	0.006235	0.036778	686.672424	-1.182478	-18.773860
90.105220	0.938462	0.417838	-53.949998	0.005918	-0.000597	12901.881836	-0.325015	-0.086923
STREAMLINE 6								
3.004852	-0.147154	0.	-50.771102	0.	-7.610848	12159.041016	0.399514	-1552.281219
2.685177	-0.289882	0.025462	-50.012261	0.	-7.391540	11959.742432	0.346660	-4975.400024
1.986931	-0.676773	0.050483	-48.883551	0.	-7.233922	11922.131714	0.250962	4686.165161
2.757111	1.802283	0.075500	-46.720125	0.004849	-5.564736	11485.923950	0.267986	-996.291580
4.062589	2.746191	0.085582	-44.589050	0.004847	-4.528446	10912.951172	0.321631	-6988.275818
8.484717	1.156943	0.123293	-37.185014	0.004838	-3.356020	8995.205444	0.500646	-3405.392944
17.853872	5.723180	0.175706	-32.627673	0.005029	1.376789	7146.185120	-0.443467	-2030.765625
27.212084	3.301493	0.210372	-29.878273	0.005245	0.718950	5459.497681	-0.369682	-2232.488037
44.228121	10.548003	0.253909	-29.933708	0.005329	4.991963	3801.527527	-4.859242	-2174.733856
71.731218	7.527054	0.304334	-33.680576	0.005068	1.557246	3016.578857	-4.717300	-593.339760
81.412612	4.883134	0.331991	-36.389583	0.005310	0.498906	-2376.510742	-3.303784	686.496490
85.929255	2.713154	0.353242	-40.335552	0.005248	0.112912	574.096863	-1.586566	-120.496057
88.387595	1.207301	0.375634	-53.949998	0.005031	0.011765	12817.429688	-0.417955	-12.193480

4

Z	R	WA	PRESS	WTR	WL	TTREL
STREAMLINE 1						
-0.075430	0.042375	617.987427	1810.882050	628.454063	607.520790	527.920029
-0.050000	0.042375	528.864166	1921.923386	555.311363	502.416969	527.920029
-0.025000	0.042375	453.919613	2004.897369	430.213726	477.625500	527.920029
0.	0.042375	638.129807	1784.119232	646.908478	629.351135	527.920029
0.010167	0.042375	575.012436	1826.039398	612.065453	537.959419	527.920029
0.054000	0.043229	440.273067	1794.442337	449.905170	430.640965	528.295410
0.120830	0.060442	335.542988	1704.843262	389.437489	281.640614	537.458199
0.155750	0.085655	419.089806	1746.701675	477.649139	360.520981	556.372025
0.184330	0.126200	521.131454	2194.479919	643.966003	398.296906	600.477211
0.201000	0.178450	764.574997	3271.660889	892.956024	636.193970	682.210899
0.204750	0.207110	810.338303	4495.681458	1009.844093	610.832512	738.949913
0.205890	0.227710	762.966583	6003.736694	916.265022	609.668144	784.942772
0.206170	0.250040	923.185478	7195.518982	928.774849	917.596107	839.720375
STREAMLINE 6						
-0.075430	0.079456	811.558098	1812.940201	826.095238	797.005196	551.115196
-0.050000	0.080727	748.752609	1933.357193	794.702576	702.784332	552.160957
-0.025000	0.081767	698.583702	2027.476196	653.749702	743.395409	553.028755
0.	0.082675	878.032661	1729.091812	899.370468	856.674393	553.794922
0.010065	0.083266	830.890221	1782.698273	913.205231	748.554077	554.299088
0.047529	0.087576	704.661819	1890.823746	755.231674	654.069443	558.079300
0.098786	0.098518	631.463585	1924.358246	700.472382	562.426483	568.534744
0.130711	0.112028	664.214363	1958.107941	713.508934	614.890160	583.140831
0.166728	0.136486	666.233994	2303.128235	763.726509	568.733528	614.351204
0.192875	0.179603	797.847511	3276.374908	935.761078	659.932220	684.331329
0.199133	0.206542	819.723518	4457.356262	1039.918396	599.528473	737.744568
0.201395	0.227673	767.949860	5994.851135	924.623489	611.276291	784.856064
0.202443	0.250040	924.466454	7189.067261	909.392128	939.540794	[illegible]

TABLE II. - Concluded. SAMPLE OUTPUT

5
```
                     SHROUD

STREAM-CHANNEL COORDINATES

M ARRAY
  -0.7543000E-01  -0.5000000E-01  -0.2500000E-01  0               0.9920799E-02   0.3968217E-01   0.7296862E-01   0.9791355E-01
   0.1332545       0.1793122       0.2048316      0.2284003       0.2510380

R ARRAY
   0.1412500       0.1412500       0.1412500      0.1412500       0.1412500       0.1413500       0.1432000       0.1469200
   0.1571200       0.1829100       0.2048700      0.2275700       0.2500400

STREAM-CHANNEL NORMAL THICKNESS
   0.3221791E-02   0.3084384E-02   0.2948232E-02  0.3039101E-02   0.2999851E-02   0.3170479E-02   0.2734923E-02   0.2184033E-02
   0.1884080E-02   0.1333241E-02   0.9827734E-03  0.7701051E-03   0.6961409E-03

BLADE COORDINATES

M ARRAY
  -0.7543000E-01  -0.5000000E-01  -0.2500000E-01  0               0.9920799E-02   0.3968217E-01   0.7296862E-01   0.9791355E-01
   0.1332545       0.1793122       0.2048316      0.2284003       0.2510380

THETA ARRAY BLADE SURFACE 1
   0               0               0              0              -0.9669029E-01  -0.4712923      -0.8729693      -1.0756145
  -1.2762059      -1.4772978      -1.5797404     -1.6559060      -1.7335161

THETA ARRAY BLADE SURFACE 2
   0               0               0              0              -0.1265826      -0.5019806      -0.9003035      -1.0976584
  -1.2970669      -1.4959751      -1.5935324     -1.6673668      -1.7335161

STGR = -1.73352       RI =  0.10415E-02   RO =  0.90280E-03

SPLITTERS    MLER =  0.12674       STGRS = -0.47327       RI =  0.11815E-02    RO =  0.90280E-03    BETAS = -38.2841
```

6
```
          INLET ANGLES  -  HUB -30.70,    MEAN -53.79,    SHROUD -60.95
```

TABLE III. - INPUT FOR QUAC FOR BACKSWEPT IMPELLER

5	10	15	20 | 21	30 | 31	40 | 41	50 | 51	60 | 61	70
TITLE								
BACKSWEPT IMPELLER								
MX	KMX	MR	MZ	W	WT	XN	GAM	AR
13	21	5	6	7853.98	.06216	15.0	1.40	1716.2
TYPE	MT	SRW	MXBL	TEMP	ALM	RHO	PLOSS	ANGR
0	0	0	4	518.7	0	.002377	1622.8	45.0
KSTR	NPRT	ITER	KD	SFACT	ZSPLIT	BETO	CORFAC	SSN
4	5	1	0	1.0	.208	-53.95	.1	8.0

ZS ARRAY						
-.07543 .097583	-.05 .13142	-.025 .16958	0 .18258	.0099208 .18892	.039682 .19167	.072917
ZH ARRAY						
-.07543 .15575	-.05 .18433	-.025 .20100	0 .20475	.010167 .20589	.05400 .20617	.12083
RS ARRAY						
.14125 .14692	.14125 .15712	.14125 .18291	.14125 .20487	.14125 .22757	.14135 .25004	.14320
RH ARRAY						
.042375 .085655	.042375 .12620	.042375 .17845	.042375 .20711	.042375 .22771	.043229 .25004	.060442
THTA ARRAY						
1.0829 -1.100	.6977 -1.300	.338 -1.500	0 -1.600	-.125 -1.675	-.500 -1.750	-.900
ZT ARRAY (IF KD = 2, ZT ARRAY USED. THTA = f(ZT). MT = No. of.)						
TN ARRAY						
.004167 .003125 .002083 0 0	.005000 .003750 .002083 0 0	.005833 .004375 .002250 0 0	.007500 .005833 .003333 0 0	.007500 .007500 .005000 .002083 0	.007500 .007500 .005833 .005417 .005000	
XZ ARRAY						
-.000833	.04725	.09445	.1417	.18333	.20833	
XR ARRAY						
.04167	.09158	.14125	.20833	.25417		

TABLE IV. - STREAM-CHANNEL AND BLADE COORDINATES FOR BACKSWEPT IMPELLER

MEAN

STREAM-CHANNEL COORDINATES

M ARRAY

-0.7546577E-01	-0.5001705E-01	-0.2500616E-01	0	0.1001300E-01	0.4444378E-01	0.8904854E-01	0.1198214
0.1600369	0.2083283	0.2351293	0.2570320	0.2794789			

R ARRAY

0.1041787	0.1051547	0.1058929	0.1064478	0.1068066	0.1094699	0.1164862	0.1256780
0.1442684	0.1807066	0.2060157	0.2276400	0.2500400			

STREAM-CHANNEL NORMAL THICKNESS

0.4358762E-02	0.4259891E-02	0.4151687E-02	0.4181409E-02	0.4095016E-02	0.3960968E-02	0.3570975E-02	0.3193651E-02
0.2587770E-02	0.1492610E-02	0.1053555E-02	0.8437943E-03	0.7244833E-03			

BLADE COORDINATES

M ARRAY

-0.7546577E-01	-0.5001705E-01	-0.2500616E-01	0	0.1001300E-01	0.4444378E-01	0.8904854E-01	0.1198214
0.1600369	0.2083283	0.2351293	0.2570320	0.2794789			

THETA ARRAY BLADE SURFACE 1

0	0	0	0	-0.8467653E-01	-0.4606674	-0.8628947	-1.0642809
-1.2663478	-1.4700192	-1.5709620	-1.6472431	-1.7265479			

THETA ARRAY BLADE SURFACE 2

0	0	0	0	-0.1290727	-0.5030818	-0.9008545	-1.0994683
-1.2974013	-1.4937300	-1.5927871	-1.6665061	-1.7265479			

STGR = -1.72655 RI = 0.14103E-02 RO = 0.15993E-02

TABLE V. - INPUT FOR QUAC FOR FLAT-VANE DIFFUSER

5	10	15	20	21 30	31 40	41 50	51 60	61 70
TITLF								
FLAT VANE DIFFUSER								
MX	KMX	MR	MZ	W	WT	XN	GAM	AR
12	21	7	7	0	.0235	18.0	1.667	593.3
TYPE	MT	SRW	MXBL	TEMP	ALM	RHO	PLOSS	ANGR
0	0	0	1	737.6	77.35	.0089470	171.4	90.0
KSTH	NPRT	ITER	KD	SFACT	ZSPLIT	BETO	CORFAC	SSN
0	5	2	1	1.0	.03	41.03	.1	0

ZS ARRAY						
.00698 .01086	.00819 .01062	.00900 .01007	.00996 .00882	.0105 .00607	.01078	.01089
ZH ARRAY						
.02527 .02138	.02405 .02163	.02325 .02216	.02228 .02343	.02175 .02617	.02146	02135
RS ARRAY						
.1917 .2417	.1958 .2500	.2000 .2583	.2083 .2667	.2167 2729	.2250	.2333
RH ARRAY						
.1917 .2417	.1958 .2500	.2000 .2583	.2083 .2667	.2167 .2729	.2250	.2333
THTA ARRAY						
0 .37228	.05166 .40823	.09686 .44069	.17174 .47054	.23387 .49096	.28606	.33161
ZT ARRAY (IF KD = 2, ZT ARRAY USED. THTA = f(ZT). MT = No. of.)						
TN ARRAY						
.0025 .0025 .0025 .0025 .0025 0025 0025	.0025 .0025 .0025 .0025 0025 0025 0025	.0025 .0025 .0025 .0025 .0025 .0025 0025	.0025 .0025 .0025 .0025 .0025 .0025 0025	.0025 .0025 .0025 .0025 .0025 0025 .0025	.0025 .0025 0025 0025 0025 0025 0025	.0025 0025 0025 0025 0025 .0025 0025
XZ ARRAY						
0	0050	.01	.015	.020	.025	030
XR ARRAY						
.180	200	.220	.240	.260	.280	300

TABLE VI. - STREAM-CHANNEL AND BLADE COORDINATES FOR DIFFUSER

```
                         MEAN

STREAM-CHANNEL COORDINATES

M ARRAY
   0             0.4100001E-02  0.8300000E-02  0.1660000E-01  0.2500000E-01  0.3330000E-01  0.4160000E-01  0.5000000E-01
   0.5830000E-01 0.6660000E-01  0.7500000E-01  0.8120000E-01

R ARRAY
   0.1917000     0.1958000      0.2000000      0.2083000      0.2167000      0.2250000      0.2333000      0.2417000
   0.2500000     0.2583000      0.2667000      0.2729000

STREAM-CHANNEL NORMAL THICKNESS
   0.9114576E-03 0.7949003E-03  0.7129374E-03  0.6164340E-03  0.5628559E-03  0.5343616E-03  0.5232948E-03  0.5265550E-03
   0.5513153E-03 0.6348968E-03  0.7394151E-03  0.9687544E-03

BLADE COORDINATES

M ARRAY
   0             0.4100001E-02  0.8300000E-02  0.1660000E-01  0.2500000E-01  0.3330000E-01  0.4160000E-01  0.5000000E-01
   0.5830000E-01 0.6660000E-01  0.7500000E-01  0.8120000E-01

THETA ARRAY BLADE SURFACE 1
   0             0.5094888E-01  0.9419942E-01  0.1668712      0.2274512      0.2785887      0.3233202      0.3633232
   0.3987394     0.4307461      0.4601997      0.4702807

THETA ARRAY BLADE SURFACE 2
   0             0.1905362E-01  0.6620308E-01  0.1432913      0.2069713      0.2602138      0.3065823      0.3479192
   0.3844031     0.4173164      0.4475628      0.4702807

STGR =  0.47028        RI =  0.12500E-02   RO =  0.12500E-02
```

TABLE VII. - INPUT FOR QUAC FOR RADIAL IMPELLER

5	10	15	20	21 30	31 40	41 50	51 60	61 70
TITLE								
RADIAL IMPELLER								
MX	KMX	MR	MZ	W	WT	XN	GAM	AR
10	21	6	10	4031.7	.018975	15.0	1.667	1246.1
TYPE	MT	SRW	MXBL	TEMP	ALM	RHO	PLOSS	ANGR
0	11	0	1	536.0	0	.001293	164.0	45.0
KSTH	NPRT	ITER	KD	SFACT	ZSPLIT	BETO	CORFAC	SSN
0	5	1	2	1.0	.16	-21.0	.1	0

ZS ARRAY

0 .11967	.02225 .13108	.043083 .13317	.05975	.076416	.088916	.10558

ZH ARRAY

0 .14642	.02225 .15142	.051416 .15142	.076416	.10142	.11967	.13642

RS ARRAY

.14592 .19167	.14600 .2250	.14700 .2490	.14975	.15475	.16067	.17325

RH ARRAY

.077416 .17892	.0775 .22283	.0805 .2490	.088166	.1035	.12217	.15033

THTA ARRAY

0 -.5924	-.078 -.6069	-.1785 -.6120	-.2659 -.6120	-.3745	-.4802	-.5519

ZT ARRAY (IF KD 2, ZT ARRAY USED. THTA = f(ZT). MT = No. of.)

0 .08475	.0055833 .09725	.013917 .11433	.02225 .15142	.03475	.051416	.068083

TN ARRAY

.002667 .0125 .002083 .01083 .00125 .0080 0.0 .00575 0.0 .00350 0.0 .001167	.003417 .0125 .002500 .01146 .001417 .008708 0.0 .00650 0.0 .00425 0.0 .001958	.00475 .0125 .003458 .01146 .001833 .01012 0.0 .007833 0.0 .005667 0.0 .003208	.009083 .00675 .003792 .001417 0.0 0.0	.01075 .00825 .005167 .002708 0.0 0.0	.01167 .00925 .006208 .003792 .001375 0.0	.0125 .01017 .007167 .004833 .00250 0.0

XZ ARRAY

-.000833 .12375	.005583 .13317	.018083 .15167	.05975	.08475	.10142	.11433

XR ARRAY

.075	.10833	.1500	.18333	.21667	.25166	

NASA-Langley, 1972 —— 1 E-6592

NATIONAL AERONAUTICS AND SPACE ADMISTRATION
WASHINGTON, D.C. 20546

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300

FIRST CLASS MAIL

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION

U.S.MAIL

019 001 C1 U 01 720204 S00903DS
DEPT OF THE AIR FORCE
AF WEAPONS LAB (AFSC)
TECH LIBRARY/WLOL/
ATTN: E LOU BOWMAN, CHIEF
KIRTLAND AFB NM 87117

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Retur

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

— NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546