CASE WESTERN RESERVE UNIVERSITY · 1826

COMPUTER ENGINEERING

AND

SCIENCE

Case Institute of Technology
Case Western Reserve University
Cleveland, Ohio 44106

Investigation and Evaluation of a Computer Program to Minimize Three-Dimensional Flight Time Tracks

NASA Grant NAG 3-101

FINAL REPORT

FINAL REPORT

Investigation and Evaluation of a Computer Program to Minimize Three-Dimensional Flight Time Tracks

NASA Grant NAG 3-101

15 September 1980 - 12 August 1981

Principal Investigator

Frederic I. Parke

Graduate Research Assistant

J.K. Kim

A Three-Dimensional Flight Planning Model supporting for DC-10 Aircrafts

GENERAL.

The program for the DC8-D3 flight planning has been slightly modified for the three dimensional flight planning for DC10 aircrafts.

Several test runs of the modified program over the North Atlantic and North America have been made for verifying the program in this project.

Referance is made to pages and formulas in 'Optimal Track Selection and 3-Dimensional Flight Planning' written by H. M. De Jong, No. 93.

MODIFICATION

1. Weather Information.

While Dr. De Jong used geopotential height and temperature in his program as a meteorological data, the modified program uses wind (direction and speed) and temperature received from National Weather Service.

A scanning program has been written to collect required weather information from the raw data received in a packed decimal format.

Two sets of weather data, the 12-hour forecast and 24-hour forecast based on 0000 GMT, will be used for dynamic processes in our testruns.

In order to save computing time only the weather data of the North Atlantic and North America is previously stored in a PCF file and then scanned one by one.

The element name (for weather data) of the PCF file is 'NWS' in our test runs.

Wind data is supplied in the form of grid point values of the vector wind

Vector wind is decomposed into x and y components to be applied to interpolation schemes.

Then it prepares spot values by interpolation with respect to the values of the wind components in the surrounding vertices of grid. See Fig. 1, Fig. 2 and Fig. 3.

2. Grid points.

While Dr. De Jong used weather data depicted from the grid points in a Cartesian grid superposed on a polar stereographic chart projection with standard parallel at 60 degree N, the modified program uses weather data depicted from the same grid points that National Weather Service uses.

See 'Aviation Digital Forecasts Program' issued by National Weather Service in Aug, 1978.

A blockette and grid points:

```
*--------* 7.5 degree

*        *

*        *

*--------* 0 degree

5        0
```

Each grid point contains its own weather data.

a. In order to compute a great circle distance on earth between two points the following equation is used:

DIST=60 arccos(sin LS sin LD + cos LS cos LD cos (RD - RS))

where LS=latitude of start,

LD=latitude of destination,

RS=longitude of start,

RD=longitude of destination,

DIST=great cicle distance between start and destination.

(See equation 5.20, P. 64).

b. To compute a true course it uses the following equation:

TC=arccos((sin LD - sin LS cos(DIST/60)) / (sin(DIST/60) cos LS)

{ if sin(RD-RS) > 0 then (360-TC) } is used

where TC= true course in degree.

c. Computation of a drift angle uses the following equation:

DA=arcsin(WV sin(WD - TC) / TAS)

where DA=drift angle,

WD=wind direction (true)

WV=wind velocity,

TAS=true air speed.

d. Heading= TC + DA (in degree).

3. Performance data for DC10.

a. Specific range table as a function of weight and latitude.

This is an important economic index which is the air distance covered by a turbo-jet aircraft per unit of fuel consumption. This data is almost independent of temperature and depends on the aircraft weight only at a prefixed flight level.

See table 1, P.116 and Fig. 4 in this documentation.

b. Climb time table as a function of temperature and weight.

See Fig 5

c. Climb distance table as a function of temperature and weight.

See Fig 6

d. Climb fuel table as a function of tempreature and weight.

See Fig 7

e. Max. weight table as a function of flight and temperature.

The maximaliy allowable weight depends on flight altitude and temperature deviation from standard.

See Fig 4.

f. Parameter r1 and r2 (See table 2, P. 116).

Parameter r1 and r2 are derived from the specific range table. Here r1 and r2 denote the coefficients of a straight line fit in a (weight, 1/entry value) graph at constant pressure altitude. According to the specific range table for DC10 performance data, r1 and r2 are as follow:

flight level	r1	r2
310	2.966	0.0000722
350	0.85	0.0000792
390	-0.83	0.0000890

g. Extra burns for step-up and step-down.

While extra burn parameters for DC8-D3 were 100 kg/ 4000 ft for step-up and -80 kg/ 4000 ft for step-down, the new program uses 700 kg/4000 ft for step-up and -28-kg/4000 ft for step-down for the DC10 flight planning.

h. The equations (6.31) in p. 121 and (6.33) in p.123 will be used in our test runs, since precisely matched parameters for DC10 are not available now.

(See lines 874-879 and 1305-1313 of the new program).

4. Initializing Input Parameters.

FLUR= sequence number.

TAXI= taxi fuel(kg).

GRW= zero fuel weight(kg).

RESERVE= reserve fuel(kg).

TOW= If -1, then computation starting in end point of flight.

MAXTOW= max. takeoff weight(kg).

MAXLW= max. landing weight(kg).

DATE= date/month/year.

IO= If -1. then east bound flight.

TTT= time of departure.

ROUTE= route indicator.

ST= starting graph point number.

ST1= destination number.

Note: These input parameters are read from card images of the PCF file in our testruns (element name = DC10F). But these may be read directly from cards.

5. Output Values.

* See output listings.

NO.=graph point number.

HEAD= heading in flight (degree).

FL= flight level in 100 ft.

TMP= off standard temperature in centigrade degrees.

TAS= true air speed (n.m).

WIND=wind along the track (+ = tail, - = head).

DIST= distance of flight segment.

ACCD= accumulated distance flown(n.m).

TIME=time along the segment flown.

ACCT= accumulated time (hour, min).

BURN= fuel consumed in segment (kg).

WEIGHT= weight (kg).

TOC= top of climb.

TOD= top of descent.

6. Reviews on Test Runs.

Since Dr. DE Jong already proved that a solution in 3-space would reward as compared with a 2-space solution, we will not treat the factorized 2-space optimalization case and the prescribed track case. Thus we will concentrate on test runs about pure 3-dimensional flight planning in free space.

While average CPU time for one flight plan with the modified program is approximately 120 seconds on UNIVAC 1108, the original program takes approximately 110 seconds or more for one flight plan on UNIVAC 1108.

However, for more than one flight plan it would require less than two minutes per one flight plan with the modified program, since they can share precalculated results for the weather information.

Note: In our test runs, two different sets of weather data and performance data will be used.

Weather data. (1) geopotential height and temperature,

(2) wind (speed, direction) and temperature.

Performance data:

(1) for DC8-D3:

standard cruise,

regularity	3 %
Mach	0.8034
taxi fuel	1500 kg
Zero fuel	75000 kg
reserves	10000 kg
max. takeoff wt.	142900 kg
max. landing wt.	93000 kg

(2) for DC10:

ORIGINAL PAGE IS OF POOR QUALITY

standard cruise,	
regularity	3 %
Mach	0.82
taxi fuel	862 kg
zero fuel	177356 kg
reserves	3629 kg
max. takeoff wt.	259458 kg
max. landing wt.	190964 kg

a. Test runs of the original program (for DC8-D3).

(1). Test run #1.

(a). Input parameters.

start-destination: New York - Amsterdam (east bound)

weather data: geopotential height and temperature

performance data: DC8-D3

flight level	r1	r2
31000	4.94	0.0000693
35000	0.54	0.0001043
39000	-0.65	0.000117

extra burn: step up 100 kg/4000 ft

step down -80 kg/4000 ft

(b) Output

* See Fig 8.

criterion	trip fuel	cost	time	distance
min. cost	36398 kg	9914	6h 51m	3705 n.m
min. time	39931 kg	10359	6h 43m	3205 n.m

* routes taken: same route

115-113-108-103-97-91-87-77-66-52-37-21-11-6-2-0

(c). Remarks.

The original program written in Burrough ALGOL has been converted for use in UNIVAC 1108.

Test run results which come from the converted program are just identical to those obtained from the original program, assuming use of same input parameters.

(2). Test run #2.

(a). Input parameters.

start-destination:	New York - Amsterdam (east bound)
weather data:	zero wind and standard temperature
performance data:	DC8-D3
r1 and r2:	same as in test run #1
extra burn:	same as in test run #1

(b). Output.

* See Fig. 9.

criterion	trip fuel	cost	time	distance
min. fuel	38327 kg	10357	7h 05m	3193 n.m

* route taken:

115-113-108-103-97-91-87-76-64-50-36-21-11-6-2-0

(c). Remarks.

To run the program under zero wind and standard temperature is a kind of minimal check.

The optimal track should then consist of the 'optimal operational distance track' in the graph as a good approximation of the great circle track between the end points.

The results of test run #2 shows a slightly different route as compared with that obtained in the test run #1.

The results should be unaffected when the program is run for a flight in opposite direction between the same points (west bound versus east bound). See table 11. P.128.

b. Test runs of the modified program.

(1). Test run #3.

(a). Input parameters.

start-destination: New York - Amsterdam (east bound)

weather data: zero wind and standard temperature

performance data: DC8-D3

flight level	r1	r2
31000	4.94	0.0000693
35000	0.54	0.0001043
39000	-0.65	0.000117

extra burn: step up 100 kg/4000 ft
step down -80 kg/4000 ft

(b). Output.

* See Fig. 10.

criterion	trip fuel	cost	time	distance
min. fuel	38225 kg	10335	7h 05m	3188 n.m

* route taken:

115-113-108-103-97-91-87-76-64-50-36-21-11-6-2-0

(c). Remarks.

Let us compare test run #3 with test run #2.

These test runs with same input parameters should show close results for the optimal operational distance track as shown in Fig. 37, P.110.

Both test run #2 and #3 have same route between New York and Amsterdam.

program	criterion	fuel	cost	time	distance
original	min. fuel	38327	10357	7h 05m	3193 n.m
modified	min. fuel	38225	10335	7h 05m	3188 n.m

Differences in results may be attributed to the use of different formulas for computing segment distance and true course, and round of errors.

Thus we know that the modified program works correctly under this minimal check.

(2). <u>Test run #4</u>.

(a). Input parameters.

start-destination:	New York - Amsterdam (east bound)
weather data:	wind and temperature
performance data:	DC8-D3
r1 and r2:	same as in test run #3
extra burn:	same as in test run #3

(b). Output.

* See Fig. 11.

criterion	trip fuel	cost	time	distance
min. time	34273 kg	9015	5h 58m	3192 n.m

* route taken:

115-113-108-103-97-91-87-77-65-51-37-21-11-6-2-0

(c). Remarks.

Let us compare test run #4 with test run #1.

ORIGINAL PAGE IS OF POOR QUALITY

These test runs were executed under conditions having standard cruise, Mach=0.8034, New York - Amsterdam (east bound), DC8-D3 performance data, and min. time criterion.

test run	weather	fuel	cost	time	distance
testrun#1	geopotential	39931	10359	6h 43m	3205 n.m
testrun#4	wind	34327	9015	5h 58m	3192 n.m

Differences in results are attributed to the use of different weather data set.

The difference in flight time implies the test run #4 used weather data set with almost east bound wind along the track. It is reasonable to have faster flight in the test run #4 than in the test run #1 since as you see in Fig. 11, an aircraft to fly may experience only tail wind along the planned track.

(3). Test run #5.

(a). Input parameters.

start-destination:	New York - Amsterdam (east bound)
weather data:	wind and temperature
performance data.	DC10
r1 and r2:	same as in test run #3
extra burn.	same as in test run #3

(b). Output.

See Fig. 12.

criterion	trip fuel	cost	time	distance
min. time	48787 kg	11129	5h 52m	3192 n.m

* route taken:

115-113-108-103-97-91-87-77-65-51-37-21-11-6-2-0

(4). Test run #6.

(a). Input parameters.

start-destination:	Amsterdam - New York (west bound)
weather data:	wind and temperature
performance data:	DC10
r1 and r2:	same as in test run #3
extra burn:	same as in test run #3

(b). Output.

* See fig. 13.

A strange symptom occurs: A negative segment flight time and a negative fuel burn between graph point number 0 and 1. Thus we temporarily add one statement in line #884 to avoid this negative magnitude.

Let us make another test run with this added statement under same conditions (test run #7).

(5). Test run #7.

(a). Input parameters: same as in test run #6.

(b). Output.

* See Fig 14.

criterion	trip fuel	cost	time	distance
min. time	68447 kg	15383	7h 52m	3283 n.m

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

* route taken:

0-1-4-8-17-30-44-60-75-87-92-98-104-109-113-115

(c). Remarks.

Let us compare test run #5 with test run #7.

These test runs are made under following conditions:

DC10 performance data, wind and temperature and same start-destination.

test run	bound	trip fuel	cost	time	distance
#5	east	48789 kg	11129	5h 52m	3192 n.m
#7	west	68447 kg	15383	7h 52m	3283 n.m

The west bound flight takes 2 hours more in flight time as compared with that obtained in east bound flight.

This is caused by the strong east-bound wind trend in weather data and longer flight distance in west bound route taken.

(7). Test run #8 and #9.

(a). Input parameters.

start-destination: New York - Amsterdam (east bound)

weather data: wind and temperature

performance data: DC10

flight level	r1	r2
31000 ft	2,366	0.0000722
35000 ft	0.35	0.0000792
39000 ft	-0.83	0.000089

extra burn: step up 700 kg/4000 ft

step down -28 kg/4000 ft

(b). Output.

* See Fig. 15 and 16.

criterion	trip fuel	cost	time	distance
min. fuel	49952 kg	11320	5h 53m	3192 n.m
min. time	51129 kg	11480	5h 52m	3192 n.m

* routes taken: same route

115-113-108-103-97-91-87-77-65-51-37-21-11-6-2-0

(c). Remarks.

These test runs show that the min. fuel flight plan gains 1177 kg of trip fuel and loses 1 minute of flight time as as compared with the min. time flight plan.

The gains may look inconsiderable.

But this is the result of one test run only.

Here another comment is made for optimal step altitude profiles.

The fuel and cost profiles have both the normal appearance: Approximation of a continuous cruise climb with one or two 4000 ft steps. However, the min. time flight is more irreyular. This is caused by the response of the aircraft's speed to the vertical temperature gradients: The level of highest temperatures is sought.

As you see temperature values in Fig. 3, some higher altitude levels have unexpectedly lower temperature values, for example, around graph point numbers 87,77,37,21,2 etc.

(8). Test run #10 and #11.

(a). Input parameters.

start-destination: New York - Amsterdam (east bound)

weather data: wind and temperature

performance data: DC10

flight level	r1	r2
31000 ft	2.966	0.0000722

35000 ft	0.85	0.0000792
39000 ft	-0.83	0.000089

extra burn: step up 700 kg/4000 ft

step down -28 kg/4000 ft

(b). Output.

* See Fig. 17 and 18.

criterion	trip fuel	cost	time	distance
min. fuel	49948 kg	11319	5h 53m	3192 n.m
min. time	51118 kg	11479	5h 52m	3192 n.m

* route taken:

115-113-108-103-97-91-87-77-65-51-37-21-11-6-2-0

(c). Remarks.

We used slightly different r1, r2-parameter sets to see how they affect flight plannings in this test run.

When we see test runs, #8, #9, #10 and #11 we understand that there is no significant difference in results under slightly different r1 , r2 parameter sets.

(9). Test run #12.

(a). Input parameters.

start-destination:	Amsterdam - New York (west bound)
weather data:	wind and temperature
performance data:	DC10
r1 and r2:	same as in test run #10
extra burn:	same as in test run #10

(b). Output.

*See fig. 19.

criterion	trip fuel	cost	time	distance
min. fuel	69928 kg	15667	7h 58m	3283 n.m

(c). Remarks.

In this test run we understand that there are significant differences in results mainly because of the east bound wind tendency in the weather data set used.

criterion	bound	fuel	cost	time	distance
min. fuel	east	49948	11319	5h 53m	3192
min. fuel	west	69928	15667	7h 58m	3283

(10). Test run #13.

(a). Input parameters: same as in test run #12 except the following:

60 % of vector wind for mean climb vector,

4 minutes for take-off and acceleration.

(b). Output.

See Fig. 20.

criterion	trip fuel	cost	time	distance
min. fuel	70404 kg	15769	8h 01m	3283 n.m

#route taken:

0-1-4-8-17-30-44-60-75-87-92-98-104-109-113-115

(c). Remarks

We made some modifications on climb distance computation to avoid the negative magnitude as described in the test run #6.

We took 60 % of vector wind instead of 75 % for the mean climb vector wind (See line #882-883 of the program listing).

The climb time is usually reduced by a few minutes to account for take-off and acceleration. We took four minutes for it in this test run (two minutes taken in test run #12).

Then let us see how modifications affect the flight plannings.

test run	criterion	fuel	cost	time	distance
#12	min. fuel	69928	15667	7h 58m	3283 n.m
#13	min. fuel	70404	15770	8h 01m	3283 n.m

It shows that the test run #13 requires slightly more cost and time than the test run #12, but not significantly.

Graph point #1 has Dist=10, Time=1, and burn=200 in this test run instead of zeros.

EXTRA COMMENTS FOR THE MODIFIED PROGRAM.

** 54-334

Procedure WDATA scans weather data received from National Weather Service and obtains the required weather information for North Atlantic and North America area.
(see the weather data format in Operational Manual issued on 9/19/78 and Aviation Digital Forecasts Program issued in Aug. 1978, by National Weather Service).
The weather data received in packed decimal format is supplied in the form of grid point values of the vector wind and temperature.

* 72-85

Procedure CHOP cuts a piece of weather information from the whole weather data for a specified grid point.

* 89-116

Procedure STOR prepares a two dimensional array (120x34) for the weather information and stores the chopped weather data in the array according to the standard pressure levels, i.e. 300 mbar, 250 mbar, 200 mbar etc.

* 119-133

Procedure AREA gets the header information of the weather data, i.e. area number, blockette number etc.

* 136-142

Procedure FINDNEXT checks the end of the weather data block.

* 145-213

Procedure GETDATA stores the weather data of each grid point in the prepared array according to GMT time, standard pressure level, area number, and blockette number.

* 216-225

Procedure SCAN activates the scanning procedures.

* 229-253

Procedure WINDCOMP decomposes a vector wind into x and y components.

* 257-262

Procedure INTP contains a formula for linear interpolation.

* 267-276

This routine prints out the weather data which are stored in the array space.

(One may delete this routine if not needed).

* 281-313

This routine converts the weather data in the standard pressure levels to the weather data in flight levels by using the interpolation formula.

* 315-332

This routine prints out the converted weather data above.

(only x component of vector wind and temperature).

One may delete this routine if not needed.

* 336-340

Initial values:

CG= conversion factor between degree and radian.

MACH= 0.82 for DC10.

P1,P2,P3= flight levels.

Q1,Q2,Q3= pressure altitudes for standard pressure levels.

* 350-353

The graph points are labelled by numbers running from 0 to 115 (note that the gpraph point numbers shown in Fig. 37, P. 110 runs from 1 to 116).

The graph is such that the point sets consist of subsets of points arranged along 'Meridians' whose indices run from 0 to 15.

Array A(index) contains the graph point number assigned to the most northerly point on each meridian.

Array A is used for the computation of zone index (See the procedure ZONEI).

* 358-372

Array V has values assigned for geographic coordinates for graph points and check points on continents: 4-digit latitude 4-digit longitude in degrees and hundreths.

(See Fig. 37, P. 110).

* 376-381

This routine fills Boolean array BK and BKG with 'TRUE'.

* 382-395

This routine denotes blocking in continental airways.

(connectivity in the graph).

Boolean array entries of BK and BKG indicate which graph points are connected or blocked (TRUE=blocked, FALSE=connected).

Array BK is used for the zone index < 3 and array BKG is used for the zone index > 8.

BK(a,b,c)=FALSE means point b on meridian a is conneted with point c on meridian a+1.

Here point b denotes the b-th point from the bottom on the meridian. Example: BK(2,2,2)=FALSE means graph point number 3 on meridian 2 is connected with graph point 5 on meridian 3 as you see Fig. 37.

* 405-414

Procedure ZONEI determines the index of the zone associated with a graph point number.

Parameter G=graph point number.

For example, a graph point number 49 gives ZONEI(49)=6.

* 416-424

Procedure DT dissects latitude and longitude from the compressed coordinates. LALAT and LALONG denote latitude and longitude respectively. Parameter K is an index of array V.

* 426-459

Procedure LIS determines actual latitude and longitude values for a graph point. (See procedure DT above).

The sign of longitude value is changed in the east of Greenwhich.

Parameter U=graph point number.

* 461-480

Procedure CTG prepares a time instant array DD, zonally specified, to be used for the time interpolation later on.

A=0.4 denotes an estimated flight time of 24 minutes per continental zone.

The instant values run from index 0 to index 14.

*484-498

Procedure HH computes grid point values according to SELECT value

If SELECT=1 , HH gets x-componant value of vector wind

If SELECT=2 , HH gets y-component value of vector wind.

If SELECT=3 , HH gets temperature value.

These values are computed under consideration of time factor for dynamic process.

Parameter V and W denote array index numbers for row and column respectively. Parameter F=index for flight levels (31000 ft=1, 35000 ft=2 and 39000 ft=3).

* 503-523

Procedure GEOP computes x and y component values of vector wind and temperature value for an arbitrary point by using procedure HH above. Bi-linear interrpolation scheme is used in this computation. Parameter C=flight level index.

* 528-538

Procedure GEODIST computes a great circle distance between two arbitrary points P1(LAT1,LON1) and P2(LAT2,LON2).

* 543-550

Procedure GEOMGRID converts coordinates of two arbitrary points on sphere to their coordinates on the geometric grid. PP and GG denote two grid point numbers.

* 555-638

Procedure PART2GEOM computes length of segment by summation of contribution from subsegments (LENGTH(I)) and a true course angle of a flight segment.

E1 and E2 denote unit vectors for the true cource.

Normalized coordinates (running from 0 to 1) are stored in array AB.

These values serve also as weightng factors later on.

Array LENGTH(I) contains lengths of subsegments on the sphere.

*640-658

These two procedures are summation formulas.

* 662-684

Procedure DRIFT computes a drift angle by using x,y components of vector wind.

** 687-741

Procedure METPROC processes meteorological data before optimization process later on.

Parameter S=flight level index.

* 695-705

This routine computes average weather parameters for each subsegment.

* 706-708

This routine computes average weather parameters for each segment in order to get single drift angle of the segment.

* 711-716

This routine computes flight time in each subsegment.

* 717-740

This routine computes temperature deviation, flight time, air distance, flight heading, x,y components of vector wind, and x,y components of true air speed in each segment and flight level.

These computed values are all average values along a segment under consideration using previously calculated contributions.

TEM=temperature.

TDEV(S)=temperature deviation.

TIME(S)=flight time in flight level index S.

AIRDIST(S)=air distance.

LE=distance of flight segment.

WIX(S)=x component of vector wind.

TAX(S)=true air speed component.

* 745-785

Procedure TABLE and CLIMBCORT are table lookup procedures for picking up performance data for DC10.

* 787-830

Procedure READQ reads in performance data and procedure WRITEQ writes out those inputed data.

One may delete procedure WRITEQ and lines 813-830 if not needed.

* 833-852

This routine reads in input parameters for a flight planning. In case of prescribed track flight (ROUTE >= 1111 case), the routine reads in Q.

In another case, it reads in ST(starting point number) and ST1(end point number) and decides Q value.

Q= # of graph points on the prescribed route or (# of zones between source and destination) - 1.

861-864

Procedure SQ and JQ are conversion formulas for indices used in subsegment algorithms by using array A(index) and array F(index) respectively.

* 866-889

Procedure DESCLIMB computes performance by using the table lookup procedure in climb(if MG=0) or descent(if MG not= 0). The calculation is somewhat complicated by the fact that the location of TOC is not fixed. The position where the flight levels off may vary due to the influence of the winds in the climb phase.

For mean climb vector wind we may take 60 or 75 % of the vector wind and the climb time is usually reduced by a few minutes to account for take-off and acceleration and thus we may take 2 minutes for it.

Array DECLTIME contains climb time or descent time.

Array DECLDIST contains climb distance or descent distance.

Array DECLFUEL contains climb fuel or descent fuel.

Array AIRDIST contains climb air distance or descent air distance.

It is noted that in the descent zone, the performance table is not used and thus specific expressions are chosen according to company's planning policy.

(See formula 6.31 in P. 120-121).

* 892-933

Procedure LINE prints out a flight plan.

** 935-1242

Procedure SPACEOPT is a key procedure for optimalization.

In this procedure, crucial is that some parameters may not surpass upper bounds, for example, aircraft weight.

This requiresthe inclusion of several protective statements in order to ensure proper functioning of the operational scheme.

FBG determines whether the flight plan computation will be performed backward(FBG=FALSE) or forward(FBG=TRUE).

FB determines whether cost(FB=1), fuel(FB=0) or flight time (FB=-1) will be optimized.

BI determines whether the navigation regime is free in the horizontal(BI=FALSE) or bounded by one position(BI=TRUE).

BG determines whether the cruising alitude is free(BG=FALSE) or bounded(BG=TRUE).

RR istake-off weight or landing weight.

* 953-961

Procedure PREP computes distance and flight time, and prints out a line of flight plan.

A1=segment distance.

B1=segment flight time.

AA=TRUE denotes a climb phase and AA=FALSE a descent phase.

* 963-1010

Procedure EDITING prints out proper heading lines for a flight plan.

** 1012-1084

Procedure QG computes segment contribution, fuel, etc.

If D=-1, computation is for in-flight direction and if D=+1, then computation is for opposite flight direction.

* 1018-1038

Procedure CL computes the aircraft weight(GRWG) and total fuel.

flight time or costs at end point of a flown segment.

(the contribution of fuel in steps included).

Parameter M1 denotes segment fuel(kg) and M2 segment time(min).

In the climb phase no step contribution is taken into account as required(BCLIMB=TRUE).

GG denotes flight level index at the graph point when the computation starts. G denotes the flight level index to be investigated.

Step consumption: UP=700 kg/4000ft

DOWN=-28 kg/4000ft.

(surplus of segment fuel)

ITERATION becomes TRUE when (1) back-tracking (2) first zone is reached.

(3) TOW is unknown and Wb is known (see P. 120).

In array ROW the aircraft weight is stored.

GRWQ denotes aircraft weight at the end point of segment.

QUANT denotes cost, fuel or time at the end point of flown segment.

*1045-1052

computation in climb or descent by procedure CL.

If ED is TRUE then it prints out the results.

*1055-1057

safeguarding statements.

*1060-1068

computes segment contribution of changes in aircraft weight.

Note that fuel consumption is expressed as loss of aircraft weight.

The table for specific range (P. 116) is used.

Formula 6.27 (P. 115) requires an estimate of weight halfway a flight segment. For that purpose coefficients as shown in table 2, P. 116 are required (r1, r2).

*1069

jumping back to label JM1 is for safeguarding that the actually derived aircraft weight does not surpass the maximally allowed weight.

*1071-1078

computes the contribution for the last segment.

If ED is TRUE then it prints out the results.

*1081

QRWQ and QUANT attain absurd values for safeguarding operational performance.

*1087

ED becomes TRUE when all degrees of freedom in the horizontal and vertical are lost, for example when a call of procedure SPACEOPT takes place in the final mode for flight p~~lan~~ computation along the (optimal) track found in previous calls.

(when navigation regimes is bounded by one point and cruising altitude is bounded).

*1088-1113

Initialization and preparation of all parameters needed for the algorithmic process of optimization.

FBQ=TRUE means flight plan computation will be performed forwards, while FBQ=FALSE backwards.

Array F is filled in line 1244-1248 (see procedure FF).

The process can be activated for arbitray begin- and end-points.

In order to reduce storage space a renumbering is made for all subset points. So array F is defined and functions completely analogous to array A for whole graph point set.

Array STOREI and STOREIQ contain the assigned numbers of labelled end-points on meridians in the subgraph.

Zone cycle runs from line 1115 to 1220.

The zone cycle repeats as many times as the number of zones.

A cycle for graph points along meridian runs from line 1122 to 1216.

This cycle repeats as many times as the number of graph points in the current meridian.

A cycle for graph points along next meridian runs from line 1127 to 1214.

This cycle repeats as many times as the number of graph points in the next meridian.

A cycle for flight levels runs from line 1158 to 1212.

This cycle repeats as many times as the number of flight levels used.

* 1126

conversion for indices used in subsequent algorithms.

* 1131-1151

If the navigation is free (BI=FALSE) in the horizontal, it checks whether a segment (II, JJ) is blocked or not.

BK or BKQ = TRUE means airway is blocked.

If blocked, it skips computation for the segment.

* 1152-1156

conversion for indices (point G4).

determines geographic elements for a segment between points G3 and G4.

See procedures GEOMGRID and PART2GEOM.

* 1162-1191

In case of backtracking in a flight plan computation an iteration process is put into action in the climb zone

This is done in order to determine the (unknown) take-off weight by iteration in such a way that the climb parameters do match with the parameters found during backtracking when arriving in the climb zone. The iteration starts with a take-off weight 20000 kg below the (known) maximum take-off weight.

Array ROW contains the (optimal) value of aircraft weight found during optimalization, using a zone-cycle.

The iteration finishes when either the weight difference at the point of matching is less than 10 kg or the number of iteration steps exceeds 10 (poor convergence).

In each step take-off weight is adjusted, see line 1181-1191.

* 1195-1203

similar computations but not backtracking case or not take-off weight adjustment case.

* 1204-1210

These statements are crucial opti~~ma~~lization criteria.

* 1225-1241

Preparation of elements which are required for a subsequent call for procedure SPACEOPT with lessened number of degrees of freedom including a final call for computation and presentation of flight plan data. In this final stage no degrees of freedom are left. Procedure SPACEOPT then merely operates along a predefined track (the solution of previous calls).

* 1244-1248

Procedure FF determines a zone index array F for a subset of graph points analogous to array A. (array F is a sub-zone array).

** 1250-1351

A call for procedure PROCES results in the production of a flight plan along whatever route is desired.

DETAILS

*1275-1289

Specification of limits in horizontal of graph points along meridians.

Array STOREI and array STOREIQ store lower limits and upper limits respectively for the number of graph points on each meridian.

If route >= 1111, then production of flight plan along a presribed route.

In this case the procedure SPACEOPT works in a degenerate mode.

* 1291-1296

Specification of limits in the vertical.

Array STOREQ and array STOREQQ store lower limits and upper limits respectively for the number of flight levels on each meridian.

*1298-1303

Procedure EP indicates that the flight planning computation blows up.

* 1305-1316

See P. 123-124 for procedures SS, SSS and SSSS.

These procedures compute landing weights.

NN contains landing weight or take-off weight.

* 1318-1324

Procedure TW is for safeguarding against overloading.

* 1327-1351

activates process as follow:

If FACTORIZAION is TRUE, then the optimalization takes place first in the horizontal and is followed by an optimalizaion in the vertical.

The following steps occur:

a) optimalization in the horizontal based on time.

ORIGINAL PAGE IS
OF POOR QUALITY

(b) optimalization in the vertical, using the track solution found in (a) and based on fuel, time or costs.

(c) computation of flight plan along solution found in (b).

If the optimalization takes place in free space, the sequence of calls results in: (three procedure calls)

(i) optimalization in space based on time, fuel or costs.

(ii) optimalization in the vertical through horizonal track found in (i), based on time, fuel or costs.

(iii) compilation flight plan along track solution found in (ii).

Note: (ii) could be bypassed as the solution is found already in (i). But (ii) can be generated with a slightly different landing weight. The compilation of a flight plan along a prescribed route passes through all three procedure calls.

This means that in fact the computation is repeated threefold, however with properly tuned landing or take-off weights.

In order to protect against subsequent calls of procedure PROCES, FOUTE is assigned by 1000.

* 1353-1364

Read statement in case that a flight plan compilation is desired along a prescribed track.

This track is specified by graph points indicated by their numbers.

* 1365-1369

Computation of an estimate of flight time to be used for the estimation of other parameters.

The same with distance.

Procedure CIQ makes time instant array DD for each zone(meridian).

Array E is for storing the number of graph points used on a meridian.

* 1371-1389

performs a process for the production of a flight plan.

Actual parameters when calling procedure ;

use landing weight (BB=TRUE).

cost optimalization (FFBB=1).

FFBB=0 means fuel opt, FFBB=-1 means time opt.

regularity percentage (PR=0.03).

fligh levels used between S1=1 for 31000 and S2=3 for 39000.

S3=1; the value is immaterial if FACTORIZATION is FALSE.

For required flight plans ;

1. when a flight plan for the optimal cost track is needed,
   call 'proces (1,3,1,0.03,1,TRUE,FALSE)'.
2. when a flight plan for the optimal fuel track is needed,
   call 'proces (1,3,1,0.03,0,TRUE,FALSE)'.
3. when a flight plan for the least ~~time~~ track is needed,
   call 'proces (1,3,1,0.03,-1,TRUE,FALSE)'.

For the flight plan 1, 2 or 3 above, FACTORIZATION=FALSE.

For other flight planning simulations, FACTORIZATION=TRUE.

Fig. 1-1.

```
* WEATHER DATA IN 3 STANDARD PRESSURE LEVELS *
* 34X16 DATA OUT OF 34X20 GRIDPOINTS *
* TEMPERATURE, DIRECTION, SPEED *

3127023 3327027 3426034 3526046 3525056 3624060 3723066 3822070 4022073 4222076 4321075 4421071 4520068 4621063 4821062 4822055
3231027 3330026 3430031 3529037 3627040 3626041 3725044 3824046 3924048 4123052 4223055 4322056 4422056 4522057 4623061 4724061
3433035 3333037 3432044 3532048 3632049 3732048 3731042 3830034 3928029 4026031 4224036 4224038 4324041 4424045 4525054 4625061
3432043 3532051 3632057 3832061 3933064 3934054 3934037 4035019 4029006 4124020 4125032 4225040 4326042 4326046 4427056 4527067
3329049 3630044 3930041 4130041 4230032 4234015 4206018 4107025 4008017 4031003 4127024 4227042 4227049 4328054 4428063 4528076
3426046 3626035 3926033 4226033 4325022 4319005 4209017 4107024 4005024 4001023 4133029 4130044 4229056 4329065 4429075 4529085
3423049 3624050 3825043 4025037 4224027 4324009 4205008 4203018 4101028 4136037 4134047 4233060 4331072 4431081 4531089 4630092
3324040 3525063 3726064 3926051 4126037 4230022 4334026 4335037 4335047 4335058 4334070 4333061 4433090 4532096 4632098 4831094
3325029 3525060 3726076 3926072 4128056 4330044 4333048 4434060 4435073 4535085 4534092 4534098 4633102 4733102 4832098 5032088
3225028 3326054 3526078 3827086 4228077 4429065 4430061 4532064 4633072 4634082 4635089 4634093 4734094 4833092 4933085 5033074
3027031 3227054 3426077 3727089 4127084 4327076 4527071 4628063 4729053 4730045 4732044 4733048 4833052 4833054 5033052 5133048
3029029 3227042 3426064 3726081 3926081 4225077 4525073 4625068 4825063 4924058 5024052 5024043 5125033 5127026 5229024 5330023
3230028 3330033 3428047 3627063 3825069 4125072 4324071 4624066 4823063 4923064 5123068 5223070 5323065 5324051 5424036 5525027
3231036 3331036 3530035 3527039 3625051 3924065 4223072 4523073 4822071 4922073 5022078 5223084 5223083 5224071 5324050 5524037
3330030 3531022 3531014 3528014 3625027 3723044 3923062 4222078 4421091 4521099 4722105 4922110 4923108 4923091 4924065 5223048
3426013 3522007 3617005 3621005 3622015 3622027 3722040 3822056 3921074 4021092 4222109 4322124 4423129 4423116 4523092 4822071
3332006 3430007 3631009 3731007 3825007 3823013 3722021 3722032 3722046 3722064 3822084 4022105 4123123 4123128 4223114 4423097
3434016 3634013 3833002 3915011 3917018 4018024 3920025 3822027 3723032 3723041 3723053 3723072 3823094 3923112 4023116 4223110
3605005 3812013 3913025 3914028 3815024 3817024 3820025 3822030 3824034 3724037 3624042 3625052 3725069 3725090 3925105 4024111
3711022 3712029 3712029 3611021 3610015 3611011 3721006 3725020 3726033 3726038 3726041 3726047 3726059 3726074 3826091 4026106
3808025 3808033 3708026 3506019 3503023 3602028 3736031 3734037 3732042 3731044 3730044 3729050 3729058 3728070 3828086 4027103
3803009 3806032 3706032 3603020 3601021 3701027 3736039 3834052 3834058 3833059 3833059 3732063 3731071 3830081 3930095 4129110
3728024 3703019 3704035 3702032 3701032 3736042 3735053 3734065 3834075 3834082 3834089 3833093 3833100 3932108 4131115 4331117
3626052 3729026 3736033 3736050 3735072 3735101 3735116 3835123 3835127 3835129 3935131 3934131 4034130 4133126 4433117 4632104
3526070 3627054 3730040 3932056 4133078 4134100 4135118 4235127 4335128 4335126 4335123 4235118 4335111 4534098 4734082 4933064
3326072 3526071 3727063 3928070 4328079 4529071 4631058 4833053 4834054 4835056 4835055 4836051 4836046 4935039 5035027 5232016
3326067 3426076 3726072 3926075 4225094 4625098 5025081 5225054 5324031 5323016 5224008 5224003 5109002 5212007 5316013 5318021
3425058 3525075 3625075 3924073 4224083 4524096 4824097 5023084 5222065 5421050 5420041 5319036 5318031 5317031 5316035 5318040
3524038 3625063 3625074 3724071 4024073 4324080 4523087 4723089 4922082 5121075 5220073 5219073 5319065 5318057 5217054 5218054
3726022 3726045 3725061 3724066 3924070 4124072 4324070 4523070 4722072 4821076 4920082 4920085 5019079 5019070 5018063 5019059
3729026 3728039 3627048 3727059 3826065 3926061 4125050 4224042 4422045 4521052 4621060 4721064 4821064 4820061 4820056 4921053
3531033 3530044 3530055 3630059 3729052 3828041 3926032 4125025 4224023 4424025 4524030 4624035 4723037 4822038 4822038 4924043
3632043 3631055 3631063 3731059 3831050 3930041 4030035 4130032 4331031 4431030 4531028 4730025 4828023 4928022 5028028 5028043
3732045 3831054 3930058 4031055 4032053 4132048 4232047 4332044 4533039 4733033 4833027 4933023 5032021 5131023 5230031 5229045
```

ORIGINAL PAGE IS
OF POOR QUALITY

Fig. 1-2.

```
* WEATHER DATA IN 3 STANDARD PRESSURE LEVELS *
* 34X16 DATA OUT OF 34X20 GRIDPOINTS *
* TEMPERATURE, DIRECTION, SPEED *

4227033 4327035 4426040 4426051 4425062 4424071 4523080 4623086 4822088 4922086 5022081 5121073 5121069 5221065 5221063 5122053
4331031 4330029 4429036 4528042 4527044 4526048 4626053 4725056 4824057 4923059 5023060 5122060 5221060 5322062 5322062 5223059
4434039 4433041 4432049 4532055 4532057 4632057 4731053 4830044 4929036 5026035 5125039 5124040 5224042 5324047 5325055 5225059
4331044 4532057 4632068 4732074 4833076 4734064 4834045 4935024 5030011 5025022 5125035 5225041 5226044 5326049 5327059 5327068
4229054 4529054 4830053 5030052 5031041 5034020 4906024 4907030 4907019 5030006 5127029 5127045 5227052 5328059 5428070 5429080
4326053 4525047 4826045 5026042 5125026 5216002 5108025 4907035 4905033 5001029 5132037 5230051 5229063 5329072 5429083 5429091
4324057 4424060 4725055 4924048 5124033 5224009 5206015 5004028 5002037 5136046 5234057 5232068 5331079 5431089 5530096 5530096
4324052 4525074 4625074 4825062 5026046 5229025 5235026 5236040 5235053 5335067 5334080 5333092 5433101 5432105 5432103 5431095
4425041 4525071 4625088 4726086 5127068 5330047 5333047 5334060 5334075 5435086 5434094 5434102 5433108 5433107 5332098 5332085
4225037 4326063 4426089 4727101 5027091 5229070 5330061 5232063 5233069 5234075 5234080 5234085 5234088 5233086 5233078 5233068
4027036 4127061 4327086 4527101 4827096 5127084 5228073 5228063 5129053 5130044 5032041 5033043 5033046 5033048 5133048 5133044
4129032 4227047 4326074 4426093 4626094 4925088 5225080 5325071 5325063 5224055 5224048 5224039 5125029 5127024 5129023 5230024
4230023 4229033 4328055 4326077 4525087 4825087 5124082 5224074 5323067 5323066 5323067 5423066 5323059 5324047 5324034 5325027
4032037 4231042 4330044 4327047 4325059 4624074 5023081 5222081 5222077 5222078 5322082 5322085 5323081 5223069 5224052 5323039
4031040 4232036 4332028 4330023 4326029 4523046 4722067 4922084 5021099 5121109 5122116 5222117 5123111 4923094 5023071 5223055
4126023 4324014 4522005 4525005 4625015 4523027 4622042 4622060 4721081 4821103 4922123 4922137 4822137 4722122 4723099 4922079
4228013 4428015 4530015 4632012 4628010 4625012 4623020 4522032 4522048 4622068 4622091 4722113 4622129 4623130 4623116 4823101
4533030 4732026 4731014 4817004 4818016 4819023 4720024 4722025 4623031 4523041 4523055 4523076 4523097 4623112 4623116 4723114
4732010 4813006 4814024 4814031 4715030 4717030 4719030 4722031 4624032 4524035 4425042 4425055 4525073 4525092 4725108 4824119
4612018 4712030 4712031 4612023 4611015 4612010 4622009 4625021 4626031 4626035 4526039 4526048 4526062 4526078 4726098 4826119
4608019 4708032 4708027 4606019 4503024 4502029 4636031 4634037 4632042 4631044 4630044 4529050 4529060 4528075 4628097 4827122
4732006 4806028 4706032 4503020 4501020 4501024 4535035 4534050 4534060 4533061 4532060 4532063 4531074 4630089 4730110 4829131
4627032 4601014 4704033 4602031 4601028 4635035 4534049 4533065 4534076 4534085 4534091 4633096 4633104 4732115 4831127 4931133
4426060 4529032 4636035 4736057 4735082 4735105 4635117 4735125 4735133 4635137 4635138 4735137 4734134 4833130 4933124 5032112
4326079 4426064 4629046 4832061 4934092 4834113 4735125 4835135 4935138 4835135 4835129 4835121 4835113 4934101 5034087 5133069
4125080 4226084 4427073 4728072 4928077 4929073 4931063 4933057 5034058 5035059 5035057 4936054 4936049 5035042 5135032 5233021
4126072 4126088 4326087 4526083 4825094 5125096 5225079 5225053 5224030 5224014 5124005 5101002 5104004 5109004 5216008 5319017
4526060 4425080 4425088 4625088 4824096 5124106 5324103 5423088 5322068 5321051 5220040 5219032 5218027 5217027 5417032 5518038
4626043 4625065 4525077 4624082 4824087 5024092 5323097 5423096 5522086 5421076 5420071 5319068 5319061 5318054 5518054 5718056
4526033 4526049 4525062 4525071 4724080 4924082 5124078 5423075 5522075 5521079 5620084 5620086 5620079 5519070 5619066 5819064
4527028 4427040 4427052 4527063 4626068 4726063 4924054 5023050 5222053 5421059 5521065 5621068 5721067 5720064 5820060 5921058
4632032 4430046 4430062 4429066 4429058 4528051 4627043 4825037 5025034 5224033 5424036 5524040 5724043 5823042 5923041 6024046
4533056 4532067 4531076 4631073 4631063 4731053 4831044 4931040 5031040 5131041 5331039 5530035 5729032 5928030 6029034 6028046
4532056 4531064 4631069 4831067 4932065 5033059 5033056 5133052 5233047 5333040 5433032 5633027 5832025 5931027 5931035 5930047
```

Fig. 1-3.

```
* WEATHER DATA IN 3 STANDARD PRESSURE LEVELS *
* 34X16 DATA OUT OF 34X20 GRIDPOINTS *
* TEMPERATURE, DIRECTION, SPEED *

5427050 5527051 5626048 5626052 5425070 5524085 5524098 5623104 5723102 5722093 5622080 5421066 5321058 5321054 5222052 5123046
5431039 5529040 5528043 5628042 5528044 5527055 5626066 5625071 5725071 5824070 5823065 5722060 5622056 5522054 5423053 5324051
5434043 5533046 5631054 5631058 5632060 5732063 5731060 5830051 5929043 6027040 6125041 6124041 5924042 5824044 5725049 5525052
5331049 5532063 5632075 5732078 5733077 5633065 5734047 5834029 5931018 6127024 6226036 6226042 6126043 6027047 5827054 5627059
5228065 5529067 5630063 5730054 5631037 5634019 5605022 5707026 5806016 6030011 6228033 6327048 6228054 6128059 5928066 5629069
5325064 5425065 5626058 5726047 5726026 5805001 5807023 5807035 5804034 6136035 6332046 6430059 6329068 6229073 5929077 5630078
5424063 5425073 5625072 5824061 6023040 6025011 6005015 5904031 6002042 6235056 6533069 6532079 6431084 6231087 5931086 5631082
5525059 5525079 5525083 5725075 6025057 6128030 6035023 6136038 6135055 6335071 6434085 6333094 6232097 5932094 5732089 5432081
5525052 5525079 5525096 5725097 5927078 6029050 6032042 6033054 6034069 6034080 6034085 5934090 5733092 5633089 5433082 5332074
5425044 5526071 5526098 5626112 5727096 5728069 5730057 5731058 5632062 5634067 5534069 5534070 5434071 5333069 5233065 5133060
5327040 5327063 5426092 5427109 5526098 5526079 5527067 5428058 5428049 5429040 5331036 5332036 5233039 5133041 5133042 5132042
5328031 5327049 5326080 5226103 5326099 5425089 5425080 5425070 5425062 5324054 5324045 5324035 5225026 5128022 5029023 5030025
5329018 5328035 5327063 5226089 5325099 5425094 5424087 5424079 5423073 5423070 5323068 5322064 5323056 5223044 5124033 5125027
5133039 5231048 5229051 5227052 5125063 5324074 5423078 5322080 5322081 5322083 5322085 5322085 5222081 5223070 5123055 5123042
5031049 5132051 5232044 5331036 5328035 5424046 5423064 5522080 5422093 5321103 5322109 5222110 5022106 5022095 5023077 5123061
5126039 5326028 5527013 5629010 5627016 5624026 5622043 5622063 5622084 5621104 5422119 5222126 5022124 4922115 5022101 5122085
5328033 5528031 5629022 5631016 5630014 5528011 5523017 5522031 5522050 5522073 5522095 5422111 5222118 5023120 5023115 5123105
5632049 5731041 5731024 5729006 5720013 5719021 5620022 5621023 5523028 5423041 5423059 5423079 5323095 5223108 5223115 5223116
5730025 5826005 5814020 5714033 5716038 5717040 5619035 5622031 5524028 5425032 5425043 5425060 5425077 5425092 5525107 5424118
5618014 5713025 5713030 5713025 5613018 5616014 5622014 5626020 5627025 5527029 5527037 5427051 5427066 5527082 5626103 5626121
5408004 5608022 5708023 5606016 5502017 5501021 5536026 5635034 5633041 5632041 5630043 5530049 5429063 5528083 5628110 5627129
5427014 5605016 5706026 5604019 5436019 5436021 5435031 5434047 5433059 5433062 5432060 5432063 5431076 5430099 5530124 5529137
5325039 5430011 5604022 5702028 5736029 5634033 5533046 5433065 5434077 5434085 5434093 5534099 5533108 5532122 5531135 5431133
5226063 5328040 5533030 5735056 5835088 5734099 5734110 5734121 5735130 5635135 5635141 5635141 5634136 5534129 5433121 5332107
5125085 5326073 5528055 5632064 5433092 5334104 5334114 5335123 5435126 5435127 5435127 5435121 5335110 5234097 5234083 5233066
5025088 5125098 5426084 5428073 5329075 5129075 5131067 5133059 5134058 5135059 5136059 5136056 5036050 5135044 5134034 5233024
5126073 5026099 5126102 5226088 5326081 5226084 5126073 5126051 5125029 5125011 5034002 5003007 5003008 5104005 5218003 5321011
5528064 5326082 5325099 5325103 5424096 5424095 5324091 5223080 5122066 5121051 5020038 5019029 5018024 5117024 5318026 5419030
5627054 5626065 5525081 5525097 5624102 5624098 5723095 5623090 5522081 5421070 5320062 5220057 5219051 5319047 5418044 5619044
5325045 5325054 5426067 5526082 5625093 5824093 6023087 6023078 5922073 5921073 5820073 5620071 5620066 5619059 5719055 5820051
5425030 5326046 5227062 5427069 5526067 5725064 5824061 6022060 6122063 6221065 6221065 6121063 6021060 6021057 6021053 6121049
5632033 5430053 5329072 5229073 5229065 5328062 5427057 5726052 6025047 6225043 6324041 6324042 6324044 6223043 6223041 6224042
5533067 5432078 5531087 5631088 5731080 5731068 5731055 5831047 5830045 5930043 5930039 6029036 6128032 6227031 6227032 6127039
5331068 5231074 5431080 5632076 5733068 5733058 5733052 5732048 5732044 5732037 5832029 5832023 5931021 6030022 6029028 6029037
```

Fig. 1-3

```
* WEATHER DATA IN 3 STANDARD PRESSURE LEVELS *
* 34X16 DATA OUT OF 34X20 GRIDPOINTS *
* TEMPERATURE, DIRECTION, SPEED *

3127023 3327026 3426039 3526065 3825086 4024080 4123067 4222059 4321057 4321056 4421057 4421058 4522060 4622055 4722047 4822044
3231027 3330026 3429035 3528048 3626059 3725060 3823056 4022055 4021058 4121061 4221064 4321068 4422071 4522067 4622058 4722050
3433035 3433038 3432044 3530043 3529038 3627034 3725029 3824030 3822036 3922042 4022049 4223055 4323062 4523063 4523060 4624055
3432043 3532052 3633057 3733057 3733050 3833037 3833024 3931014 3926016 3925027 4025036 4126044 4326050 4425054 4525057 4626057
3329049 3630045 3931044 4031043 4033033 4036021 4004018 4006011 4022003 4026020 4127035 4227044 4227049 4328053 4428060 4528064
3426046 3725036 4025033 4225029 4322017 4314010 4210019 4109018 4010003 4027017 4128039 4228052 4228057 4329061 4429069 4529073
3423049 3624049 3825040 4124030 4221023 4318013 4211012 4107010 4134010 4130027 4229048 4229061 4329067 4430074 4530081 4530083
3324040 3525062 3725059 3925039 4124025 4224013 4232006 4134015 4233028 4232044 4331059 4331068 4431074 4530083 4630089 4631086
3325029 3525058 3625069 3825055 4026036 4128026 4231027 4333039 4333052 4433064 4432073 4532078 4631083 4731088 4731087 4731079
3225028 3326052 3426067 3726068 3927056 4129048 4332053 4432064 4533073 4633080 4633085 4733088 4832089 4932086 4931078 5031065
3027031 3226051 3426066 3626078 3927079 4129068 4331065 4432069 4532072 4633074 4733075 4833074 4933072 5032068 5032060 5132050
3029029 3227039 3426060 3626084 3927093 4227081 4427068 4428056 4528046 4629040 4730037 4831033 4931031 5031031 5131031 5231029
3230028 3330033 3528051 3627078 3926090 4226081 4425073 4524070 4623066 4723061 4823055 4922049 5022044 5123041 5224036 5325030
3231036 3431038 3630042 3727052 3825065 4024068 4223070 4422076 4522082 4722085 4822084 4922079 5022076 5123072 5223064 5424051
3330030 3531021 3631014 3627018 3724032 3923046 4022059 4222071 4322081 4522087 4723088 4823084 5023083 5123085 5224080 5424064
3426013 3520004 3511006 3514002 3621009 3722022 3822037 3922053 4122069 4323092 4523089 4723090 4923091 5124094 5225089 5325070
3332006 3433005 3636006 3705007 3712006 3718010 3721020 3722033 3822049 4023[illegible]7 4223085 4523099 4723107 4924112 4925105 5025077
3434016 3635011 3810006 3913020 3915026 3917028 3919028 3821032 3822039 3823051 3923071 4123094 4324115 4524128 4625118 4625085
3605005 3811016 3812029 3813030 3714026 3716025 3820027 3922034 3924037 3824040 3824052 3924075 4024103 4225125 4325125 4425094
3711022 3711031 3711030 3610024 3610020 3612013 3723012 3825029 3826036 3827037 3726041 3726055 3825081 4025108 4225118 4425101
3808025 3808034 3708025 3607019 3706023 3704022 3734020 3830030 3829038 3828039 3728040 3728047 3827065 3927090 4126107 4425099
3803009 3806032 3707027 3706010 3802011 3801028 3736039 3734041 3832042 3831043 3730044 3730049 3829062 3928080 4127095 4327094
3728024 3804018 3705026 3735009 3830022 3833029 3835038 3834048 3834056 3733058 3732060 3732064 3831072 3930083 4129091 4329090
3626052 3730023 3736030 3734043 3633054 3734060 3734070 3735081 3834086 3834089 3834090 3933090 3933091 4032094 4131096 4331092
3526070 3627052 3731045 3733068 3634091 3734102 3935113 4035120 4135120 4135119 4135115 4135109 4234103 4234097 4333091 4532082
3326072 3526071 3727065 3829068 3830070 4132068 4434067 4535070 4635077 4635081 4636080 4635077 4635075 4735069 4834060 4933048
3326067 3426078 3726079 3925083 4125085 4325070 4624050 4824033 5024017 5030008 5135012 5136017 5136021 5135021 5235017 5332008
3425058 3525079 3625084 3924084 4223093 4523095 4722089 4922082 5121073 5321059 5421042 5421026 5320017 5319015 5318016 5417020
3524038 3625066 3624081 3724079 3923079 4123083 4422086 4622091 4821096 5121095 5321081 5421060 5320047 5219045 5218044 5318041
3726022 3726045 3725063 3624070 3724070 3923068 4123071 4423078 4622085 4822091 5022090 5121081 5020072 5019070 5019067 5119059
3729026 3728036 3627044 3626054 3625060 3725062 3925062 4225060 4424058 4623061 4722068 4821074 4820075 4820074 4820072 4920066
3531033 3530041 3530046 3629048 3628051 3728057 3928057 4127047 4226036 4424033 4522039 4621046 4621051 4721053 4721055 4822058
3632043 3631054 3731059 3732056 3731051 3830047 3929041 4128032 4228024 4328018 4428016 4626017 4725019 4724023 4724032 4824044
3732045 3931050 3931049 4032045 3932044 4032042 4131042 4231040 4432034 4633030 4734026 4834023 4934018 4932015 4929018 4927033
```

Fig. 1-5

```
• WEATHER DATA IN 3 STANDARD PRESSURE LEVELS •
• 34X16 DATA OUT OF 34X20 GRIDPOINTS •
• TEMPERATURE, DIRECTION, SPEED •

4227033 4327035 4426047 4526074 4725095 4925083 5023068 5022062 5021060 5122058 5122057 5122058 5222060 5122055 5122048 5122046
4331031 4330029 4429040 4428055 4526065 4524064 4623063 4822066 4822067 4922067 5021069 5122071 5222072 5222068 5222050 5223053
4434039 4434042 4432049 4431049 4429042 4528037 4626033 4724034 4723039 4822046 4923052 4923059 5023065 5123067 5223064 5224058
4331044 4533058 4633069 4633069 4533059 4633044 4833030 4831019 4927020 4926030 5025039 5026047 5025054 5025059 5125062 5125059
4229054 4630055 4831057 4831056 4733042 4835027 4802022 4903013 4927007 5026026 5127041 5127049 5127053 5127057 5127062 5127063
4326053 4625048 4925046 5125037 5122021 5213011 5108025 4908023 4904003 5028025 5128049 5128060 5128063 5129065 5129070 5129070
4324057 4424060 4725054 5024045 5121033 5217015 5109018 5007019 5035012 5131035 5230058 5229071 5229075 5230080 5230082 5030080
4324052 4525073 4625069 4825050 5024032 5125013 5101008 5101016 5133029 5232049 5331066 5331076 5430084 5330090 5231090 5031083
4425041 4525069 4625080 4725068 4926046 5128029 5232027 5233039 5333055 5333069 5432077 5432082 5531088 5431090 5231086 5031076
4225037 4325060 4425078 4626084 4727072 5029054 5231054 5332065 5432075 5533084 5533089 5533090 5432088 5332083 5232074 5131063
4027036 4126057 4326076 4526093 4727095 4928076 5130065 5131068 5232072 5332073 5333073 5333071 5233067 5232062 5132055 5132047
4129032 4227044 4326069 4526096 4727104 4927087 5027070 5028057 5028046 5029040 5030036 5030033 5031031 5031031 5130031 5130030
4230023 4229034 4328061 4527093 4727099 4926084 5025074 5124070 5123064 5123059 5123053 5223046 5223039 5124036 5225033 5226030
4032037 4232046 4430054 4528067 4726078 4824076 5023077 5122082 5322088 5422090 5422087 5422079 5423072 5423066 5324059 5324047
4031040 4232034 4431025 4528028 4526043 4724053 4823063 5022075 5222087 5322096 5423096 5523091 5523086 5524084 5524078 5424062
4126023 4325009 4403001 4432004 4426011 4523022 4622038 4722056 4922073 5122088 5323096 5523097 5523096 5524097 5424088 5324070
4228013 4429012 4533010 4602009 4504003 4522005 4522019 4622035 4622053 4822074 5023093 5223108 5323114 5324114 5224104 5025079
4533030 4633023 4735004 4714017 4715027 4717028 4719026 4621030 4623039 4623054 4723076 4823101 4924121 5024129 4824117 4724089
4732010 4812010 4813029 4714034 4615031 4617029 4720029 4722032 4723034 4624040 4624054 4624079 4724107 4824124 4724122 4624098
4612018 4712032 4711032 4611025 4610019 4613012 4623012 4725026 4626032 4626034 4526041 4626059 4625086 4725111 4725119 4725104
4608019 4708034 4708027 4608019 4507019 4604018 4633019 4630029 4628035 4628037 4628039 4527048 4627069 4726096 4826113 4825105
4732006 4806027 4707027 4608009 4535006 4635026 4635039 4633041 4632041 4631041 4530044 4529051 4629065 4728087 4827104 4927102
4627032 4602012 4705022 4633010 4629029 4632032 4634037 4534045 4534051 4533054 4532058 4532065 4631076 4730091 4829101 4929099
4426060 4529028 4635029 4734045 4632056 4533058 4534068 4534076 4534079 4534083 4634087 4734092 4733099 4732105 4831105 4930099
4326079 4527061 4630052 4633076 4533094 4634104 4734118 4735124 4735120 4735118 4835117 4835116 4834113 4834106 4833097 4932086
4125080 4326083 4527073 4529073 4530076 4732071 4834071 4835074 4835079 4836083 4936085 4935084 4935081 4935073 5034064 5133052
4126072 4226090 4426091 4626088 4726088 4825073 4925051 5025031 5027015 5033013 5036020 5036024 5036025 5135025 5134020 5332012
4526060 4525083 4425095 4524099 4924106 5223104 5423095 5522084 5522071 5421054 5321035 5221021 5120014 5120012 5219013 5418017
4626043 4625067 4525083 4424094 4623103 4923105 5322105 5522105 5722106 5722100 5621082 5421059 5320047 5219044 5319043 5419041
4526033 4526049 4525064 4524077 4624087 4724089 4923092 5223098 5523103 5723103 5822095 5721082 5520074 5419071 5519068 5619061
4527028 4427038 4427050 4526063 4626073 4626077 4825077 5025074 5325071 5523070 5622073 5621076 5620077 5619076 5620074 5720069
4632032 4431044 4330057 4530062 4529061 4628067 4828064 4927054 5126043 5224040 5323044 5422049 5521052 5621054 5621058 5722062
4533056 4532066 4432071 4531068 4631061 4730056 4829050 4929040 5028033 5228027 5328024 5527022 5626022 5725026 5724035 5824049
4532056 4631058 4631056 4632054 4732054 4831053 4931051 5031046 5132039 5333034 5434030 5634026 5733021 5932018 5929021 5927037
```

Fig. 1-6

```
* WEATHER DATA IN 3 STANDARD PRESSURE LEVELS *
* 34X16 DATA OUT OF 34X20 GRIDPOINTS *
* TEMPERATURE, DIRECTION, SPEED *

5427050 5526052 5626060 5626084 5725101 5725080 5623061 5522054 5422050 5422048 5522049 5422050 5323048 5223045 5123041 5023041
5431039 5529039 5528048 5527061 5526072 5524069 5623071 5622074 5722071 5722069 5822070 5422070 5722065 5522059 5323053 5123047
5434043 5533047 5532053 5531051 5430042 5528038 5626037 5725040 5724044 5823050 5823058 5823064 5823067 5723066 5424060 5224053
5331049 5532065 5633078 5633079 5634070 5633053 5833037 5931026 6028026 6026035 6026048 5926052 5925059 5725062 5525060 5225053
5228065 5529065 5631063 5632055 5533042 5535029 5736023 5835016 6029017 6127034 6227049 6127055 6027056 5826057 5527057 5327053
5325064 5525065 5725056 5725037 5623017 5711008 5807022 5805020 5933011 6128038 6228062 6228069 6028064 5828060 5528060 5329058
5424063 5424072 5624070 5824058 5921039 5918014 5908016 5905019 6035018 6231045 6230069 6230077 6030074 5830071 5530069 5330068
5525059 5525079 5625079 5725063 5924040 6025016 6001009 6101019 6334034 6432056 6331072 6231076 5931078 5731077 5431076 5231073
5525052 5525076 5625089 5725082 5825062 5928037 6032028 6233039 6433057 6433071 6232076 6032076 5832076 5532077 5331075 5231070
5425044 5525069 5525089 5625099 5626091 5728063 5830051 6032059 6132069 6033074 5933076 5733075 5532073 5432070 5332066 5232060
5327040 5426059 5526082 5526103 5627104 5628082 5629064 5731062 5731063 5632061 5532058 5432056 5332055 5232052 5132049 5132045
5328031 5326047 5426076 5526106 5627113 5527093 5527072 5527058 5428047 5428039 5329035 5229031 5130030 5130031 5030033 5030033
5329018 5328038 5327071 5527106 5627107 5526084 5425070 5524066 5423060 5423054 5423049 5323042 5224035 5225033 5126032 5127031
5133039 5231054 5330065 5428081 5626088 5624079 5723078 5722081 5722084 5622084 5522079 5522071 5423063 5423056 5324050 5224043
5031049 5231048 5331040 5429042 5527058 5625063 5823068 5923078 5922089 5822095 5622093 5523088 5523082 5523078 5423071 5324058
5126039 5227023 5431013 5432014 5429020 5525026 5622039 5722059 5822077 5822091 5722097 5622096 5523095 5523094 5424086 5324071
5328033 5528026 5631015 5635012 5534010 5529010 5524020 5523038 5622059 5622081 5722097 5623106 5523108 5323107 5224099 5124081
5632049 5732036 5732011 5614013 5616027 5618026 5620023 5522027 5523040 5623059 5623081 5623099 5423111 5224115 5024108 5024090
5730025 5820003 5713026 5714038 5616037 5718033 5720028 5621026 5623029 5624040 5624057 5624080 5524103 5324114 5024112 4924097
5618014 5713027 5712032 5611025 5611017 5713008 5722009 5725017 5626023 5626031 5526043 5525064 5525088 5425106 5125109 4925100
5408004 5608024 5708025 5609018 5509015 5704009 5732015 5729025 5728030 5628033 5627039 5627052 5626073 5526092 5326102 5125098
5427014 5605014 5707021 5612007 5427009 5633026 5733039 5733041 5632040 5531039 5530044 5529054 5528069 5528086 5327096 5227094
5325039 5528009 5704007 5630016 5530034 5632035 5634036 5434041 5434044 5433050 5533058 5432068 5431081 5430092 5329095 5329091
5226063 5428037 5732028 5832046 5832055 5632055 5433061 5334067 5334070 5435076 5535082 5534088 5533096 5432100 5331096 5330087
5125085 5326069 5530059 5532079 5532092 5433096 5434109 5435118 5435117 5336111 5336107 5335105 5334103 5334096 5233086 5232075
5025088 5225095 5327081 5229076 5130079 5231073 5233073 5135077 5136081 5136083 5036084 5036083 5135078 5135069 5234059 5233048
5126073 5126100 5126101 5126090 5126083 5226071 5226050 5126030 5130017 5135021 5036030 5036033 5036031 5035028 5134022 5232016
5528064 5426083 5325098 5324103 5324101 5424099 5423090 5423078 5422062 5322044 5222028 5121017 5021011 5021010 5120011 5320015
5627054 5526064 5425081 5324105 5423118 5623113 5723108 5823103 5722096 5622084 5421069 5220054 5120043 5119039 5219037 5319035
5325045 5325056 5325071 5425091 5625113 5724116 5824111 5823104 5823099 5823092 5722081 5621070 5420063 5419059 5519056 5620052
5425030 5226046 5327067 5527085 5627095 5726096 5826093 5825085 5925075 5924070 6022069 6021068 5920066 5920065 5920063 5921061
5632033 5431053 5330075 5530081 5629076 5729074 5828071 5927061 6026050 6125047 6323050 6422051 6422050 6421051 6322055 6222058
5533067 5432074 5331080 5431079 5531072 5730067 5829060 5929051 6028044 6128039 6228034 6427029 6526027 6525030 6524039 6424049
5331068 5231064 5231063 5331064 5531068 5631066 5731061 5831053 5831044 5832037 5932031 6033025 6132020 6330019 6428026 6426039
```

Fig. 2-1

```
*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*
*X-COMPONENTS OF WIND DATA*

   25    29    34    46    54    54    53    50    49    50    41    36    26    32    31    35    32    23    27    49
   22    23    29    36    41    42    44    43    44    41    43    37    35    38    45    51    42    34    38    52
   17    19    29    32    32    32    34    31    30    32    32    33    36    39    51    57    57    50    49    55
   29    34    39    41    33    19    13     3     7    18    31    38    41    46    57    67    69    61    55    52
   47    41    38    38    29     5   -17   -24   -17     3    25    43    50    54    64    75    76    72    61    48
   47    36    35    34    22     1   -19   -25   -20    -4    17    39    54    63    72    81    77    70    52    42
   41    45    43    37    24     8    -8   -11    -7     0    17    33    56    63    72    81    75    59    47    35
   37    62    65    52    38    20     8     5     8    10    25    42    46    63    64    72    64    52    39    26
   30    59    77    74    58    39    24    21    16    15    31    34    52    52    63    57    47    37    29    18
   28    55    80    90    80    62    53    41    36    28    18    31    32    45    42    36    31    31    19    16
   32    56    78    92    87    78    71    62    50    39    28    24    25    26    26    24    26    22    17     8
   28    43    65    83    83    75    70    65    59    50    44    36    30    26    23    20    20    19    16     7
   23    29    48    66    69    71    63    59    49    49    52    53    49    43    31    25    24    23    19     9
   27    29    32    41    50    58    57    55    47    48    51    62    64    59    43    32    29    29    23    11
   27    18    13    15    26    34    46    51    47    51    69    72    83    70    56    38    35    33    28    15
   15     6     0     3    11    18    26    37    38    47    72    82    96    86    71    47    41    44    34    17
    6     8     8     6     8    10    13    21    30    42    55    68    91    98    87    75    67    57    42    21
    7     7     3    -5    -2     1     9    17    25    31    41    56    72    86    89    85    80    80    58    29
   -2   -10   -18   -18   -13    -4     8    19    29    32    37    50    66    85    99    98   102    95    68    30
  -20   -25   -25   -20   -15   -10     4    19    32    36    40    46    59    74    92   107   114   104    71    30
  -24   -32   -26   -16   -12   -10     0    13    27    34    38    47    55    70    88   108   117   104    67    24
   -3   -27   -28   -10    -4    -5     1    18    20    30    32    40    55    72    85   108   111    90    56    15
   26    -8   -22   -11    -6     1    11    25    26    28    30    47    50    70    90    93    94    70    35     4
   53    25     0     0    13    18    20    21    22    22    23    40    44    63    60    68    64    42    18    -5
   71    56    37    37    37    35    20    22    22    22    21    20    19    34    28    33    29    21     7    -8
   72    73    65    69    78    67    45    27    18    10    10     0     0     7     5    10    14    14     7    -5
   67    78    75    76    88    92    76    51    27    12     6     2    -2    -6    -4     1    10    16    13     1
   56    71    73    68    75    85    85    65    42    25    14     6     0    -5   -11     0    15    22    24    11
   35    59    70    63    66    72    69    69    53    38    25    13    11     0    -7     0    18    35    34    20
   24    45    57    59    63    64    63    55    46    38    28    29    17    12     3    10    30    47    46    28
   25    38    49    60    65    60    47    36    30    27    31    32    32    21    19    27    45    61    54    34
   24    38    49    54    50    42    35    26    23    24    27    31    30    26    26    38    56    69    61    37
   28    42    50    48    40    37    31    29    26    25    23    24    25    24    29    43    61    72    62    39
   31    43    51    44    36    31    30    28    21    18    14    12    14    19    27    42    60    67    60    40
```

Fig. 2-2.

```
*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*
*X-COMPONENTS OF WIND DATA*

   37   38   41   50   60   64   66   69   62   56   52   36   34   32   32   34   30   23   26   46
   25   28   36   41   44   49   55   56   53   48   47   39   31   39   40   45   39   32   36   52
   13   21   33   37   37   38   42   39   35   35   37   35   36   40   51   54   53   47   46   52
   35   38   45   48   38   24   15    5   11   22   33   39   43   48   58   66   65   58   52   52
   54   54   48   45   30    7  -20  -27  -17    6   30   46   52   58   68   73   73   67   59   48
   54   48   47   42   24   -1  -24  -33  -24   -4   25   46   60   68   77   82   74   65   52   45
   50   56   55   44   29    8  -13  -18  -13    2   22   46   62   68   79   79   66   57   46   34
   47   71   72   61   47   25    5    0    9   12   27   46   53   65   64   68   60   50   35   29
   41   69   85   86   70   42   25   22   26   18   31   34   52   52   58   53   45   37   31   21
   36   64   90  103   92   66   52   41   36   25   26   28   29   41   38   33   30   31   21   17
   37   61   87  103   96   83   71   61   50   38   26   22   22   23   23   23   26   22   15   10
   30   47   74   94   94   83   75   67   59   48   41   33   26   24   22   21   21   20   15    8
   19   32   56   79   84   83   72   65   52   52   51   49   45   39   29   25   23   22   17    8
   23   33   40   48   56   64   62   52   50   51   53   55   60   53   44   30   27   27   20   10
   33   25   20   22   30   36   44   53   52   54   74   74   81   70   55   43   37   33   25   14
   26   15    5    6   14   21   27   39   44   52   78   86   86   77   74   52   44   44   33   20
   17   18   15    9   10   11   15   21   31   45   59   73   81   98   89   78   69   59   44   24
   18   20   13    1    1    4    8   15   23   31   43   59   74   85   89   87   83   81   60   33
    9   -3  -15  -20  -15   -5    5   20   27   30   39   53   70   86  101  103  115  101   73   40
  -13  -24  -26  -20  -14   -8    7   20   30   33   38   48   62   78   98  117  127  113   79   41
  -16  -30  -26  -16  -11   -9    0   11   26   32   38   46   57   76   99  124  133  114   76   36
    6  -21  -27  -10   -2   -3    6   17   23   31   39   40   57   79   98  124  124  100   65   28
   33    1  -19  -11   -4    7   18   33   26   29   31   45   52   75   98  102  102   75   44   16
   60   32    3    2   14   21   24   26   23   24   24   24   46   60   62   71   65   45   26    6
   78   65   45   39   34   38   26   23   24   23   22   21   20   35   30   35   31   22   13    0
   77   85   75   71   75   69   49   29   20   10    8    0    0    7    7   11   14   14   10    1
   71   89   89   83   86   88   74   50   26   12    3   -1   -3   -4   -2    4    8   14   13    4
   60   76   85   86   83   90   87   66   44   26   14    6    0   -5   -5    1   13   19   21   12
   65   62   73   75   78   81   74   73   54   37   23   14   11    2    0    2   18   30   30   19
   34   49   60   70   73   73   68   58   48   39   28   28   26   12   11   12   30   43   40   26
   28   41   54   64   67   62   48   38   35   31   33   34   33   24   22   28   45   53   49   32
   21   41   57   64   56   52   46   38   35   31   32   35   37   32   31   39   55   64   54   35
   29   45   60   58   51   43   36   32   33   32   31   31   30   30   32   44   57   65   55   36
   39   51   55   51   40   30   28   27   25   21   17   14   16   21   27   40   53   59   53   37
```

Fig. 2-3.

```
*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*
*X-COMPONENTS OF WIND DATA*

   51   52   48   51   67   75   87   81   80   60   51   33   29   27   33   35   28   23   27   43
   30   39   43   41   43   56   66   68   68   62   50   39   36   35   41   44   36   32   35   48
   15   23   42   45   39   40   46   44   40   40   39   36   36   38   46   49   48   44   44   49
   38   40   48   50   39   34   16   10   14   24   35   41   42   47   54   58   57   53   49   55
   65   64   56   47   28    6  -17  -24  -14   10   32   48   53   58   65   64   63   56   56   49
   61   62   58   46   26   -1  -22  -33  -22    0   30   52   64   69   72   67   64   53   51   47
   55   70   69   54   31   10  -11  -20  -14   11   36   52   64   67   65   62   58   54   41   37
   56   74   79   71   55   31    4    0   10   12   29   47   63   59   56   50   47   41   37   33
   50   75   91   91   79   47   27   27   24   28   29   31   45   44   39   47   42   37   33   23
   41   71   98  111   96   68   49   44   40   23   24   24   24   34   33   30   35   31   22   19
   40   63   91  110   97   78   67   57   48   38   28   23   20   21   21   27   26   24   17   14
   31   49   79  102   97   84   75   66   58   47   39   30   24   22   22   22   23   18   16    8
   17   34   64   89   94   88   75   68   56   54   52   40   43   33   29   25   23   20   15    8
   20   37   49   52   59   64   60   51   52   53   55   55   51   54   42   32   27   23   18   11
   39   34   29   29   34   40   49   51   61   52   70   71   67   60   59   47   33   29   23   16
   39   29   14    9   16   23   28   40   55   52   76   80   79   74   64   55   46   39   31   22
   33   32   22   12   12   11   13   20   32   47   61   71   75   91   88   80   70   59   46   27
   32   32   19    7    4    4    8   12   21   31   45   61   73   83   88   89   82   79   61   39
   23    6  -13  -21  -13   -7    6   20   24   30   40   56   72   86  101  102  110   96   73   46
    1  -18  -23  -19  -14   -5    9   20   25   29   37   51   66   82  101  119  122  105   80   49
   -3  -21  -23  -14   -6   -4    0    5   21   25   37   42   59   83  109  130  128  107   78   46
   15  -11  -23  -12    0    0    5   16   31   31   39   40   58   87  108  129  118   95   67   38
   37   11  -13  -10    0   11   23   33   26   29   32   33   54   78  103  102   98   73   52   27
   62   40   16   11   15   35   39   42   23   23   24   24   47   42   61   69   53   46   30   16
   80   73   55   41   47   36   40   21   22   22   22   21   19   33   28   33   31   25   20    9
   84   93   84   72   70   70   51   30   20   10   -1    0    0    8   12   12   15   16   12    6
   72   98  101   87   79   82   72   50   27   10    1   -4   -4   -3    0    6   10   15   15    8
   63   81   94   98   83   81   78   61   42   26   13    5    0   -4    0    5   11   17   19   13
   55   64   76   92   89   85   73   69   52   35   21   20    9    9    0    9   14   19   21   17
   43   51   67   82   88   82   67   60   47   37   25   24   23   10   10   17   23   33   33   25
   28   45   63   69   66   60   53   39   40   33   33   32   30   30   27   25   37   41   39   30
   21   46   69   70   61   62   58   52   45   41   36   36   38   33   31   36   44   50   44   33
   34   51   68   68   62   53   43   36   40   37   34   34   32   31   32   39   47   52   45   34
   53   58   62   49   33   29   26   31   28   24   19   15   16   19   26   35   44   49   43   33
```

Fig. 2-4.

```
*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*
*X-COMPONENTS OF WIND DATA*

     25    28    40    66    83    71    51    38    29    30    31    31    39    35    30    28    26    19    18    24
     22    23    34    49    59    56    44    37    32    34    33    37    46    43    37    34    33    25    21    29
     17    18    29    37    37    34    28    27    25    28    33    43    48    49    47    48    40    29    26    37
     29    32    30    30    26    20    13    12    17    26    35    44    49    52    55    56    49    40    37    43
     47    41    36    35    18     1   -11    -9     3    21    36    45    50    53    60    63    57    49    47    46
     47    37    34    29    11    -6   -20   -19    -3    19    40    53    57    58    65    68    60    51    49    41
     41    44    41    29    13    -1   -13   -11     3    24    46    59    65    65    70    71    59    54    41    37
     37    61    57    39    23    11     3     3    14    29    46    53    61    73    75    66    58    49    35    23
     30    57    67    55    37    27    20    20    26    33    47    51    65    67    67    60    50    39    28    18
     28    52    68    71    60    46    36    41    40    40    43    44    57    55    57    50    39    30    21    14
     32    51    67    81    83    67    51    46    46    39    38    37    36    43    38    32    26    25    21    14
     28    40    61    86    96    82    68    55    45    38    32    26    24    24    25    23    23    22    20    13
     23    30    52    82    91    81    69    61    51    47    42    32    28    31    31    28    25    24    21    14
     27    29    39    55    65    61    55    50    54    56    54    51    50    54    49    43    33    29    23    14
     27    17    13    20    31    38    40    46    53    57    69    65    64    67    69    55    40    31    23    14
     15     3    -5     0     6    15    24    34    45    62    69    70    71    82    82    65    45    31    25    18
      6     5     1    -5    -4     1    10    21    32    50    66    78    83    97    97    72    44    26    22    17
      7     4    -4   -14   -13    -5     5    16    26    39    55    73   101   111   109    79    45    24    20    21
     -2   -14   -24   -23   -17    -8     9    22    31    35    45    66    90   115   114    87    51    30    23    26
    -20   -29   -28   -24   -20   -11     9    26    34    36    40    55    77   102   111    96    63    36    31    32
    -24   -33   -25   -18   -20   -14     8    26    36    38    39    46    66    91   106    94    74    52    43    38
     -3   -27   -25    -9    -3    -3     2    16    27    33    38    43    59    81    97    96    79    64    52    39
     26   -10   -19     3    21    16     8    16    19    29    39    41    56    74    88    87    82    68    54    33
     53    21     1    15    29    23    24    17    29    30    31    42    47    62    75    74    72    59    45    23
     71    54    37    35    35    35    25    21    21    21    20    19    36    34    47    53    53    42    28    13
     72    73    67    65    62    44    23    12    13    11     0    13    13    12    21    24    28    23    16     6
     67    80    81    80    82    67    44    29    15     7     2     0     0     4     4     6     8    11    15    10
     56    75    81    76    76    75    61    53    39    29    20    13     6     3     0    -2     1     7    16    19
     35    62    72    71    65    68    58    60    53    52    41    30    16     8     2     2     2     9    24    29
     24    45    59    62    64    58    58    64    61    63    59    41    25    12    12    10    11    18    31    38
     25    36    45    55    60    62    61    59    54    49    45    37    26    22    25    23    29    33    44    49
     24    36    42    47    52    58    58    49    37    30    27    25    26    27    28    38    38    48    55    53
     28    41    45    40    41    43    41    33    26    20    18    18    19    21    28    39    48    59    66    57
     31    39    39    30    30    30    34    32    23    15     9     8     7    10    18    34    54    70    74    58
```

Fig. 2-5

```
*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*
*X-COMPONENTS OF WIND DATA*

    37    38    49    75    90    77    51    39    30    36    36    36    39    35    31    30    28    23    20    23
    25    28    40    56    66    56    49    43    44    43    37    46    45    43    39    40    33    26    22    28
    13    16    32    38    38    36    33    31    32    32    41    46    50    51    50    49    40    29    26    37
    35    32    36    36    29    23    16    16    21    31    38    47    52    56    58    54    47    37    35    44
    54    51    45    41    21     5    -6    -5     9    28    43    50    54    57    61    61    53    45    44    43
    54    48    45    35    13    -8   -24   -21     0    28    51    61    62    61    64    64    55    52    46    40
    50    54    53    41    18    -2   -18   -17     2    29    52    67    69    67    69    67    56    51    40    37
    47    70    67    50    30    13    -1    -3    14    32    52    58    70    74    67    62    55    47    36    25
    41    66    77    67    48    31    17    20    28    35    49    52    63    65    64    57    48    35    30    21
    36    58    75    85    76    53    42    41    47    41    43    43    55    51    47    46    39    28    24    16
    37    56    76    94    97    76    57    51    46    45    37    36    34    38    34    30    27    26    19    13
    30    44    70    97   106    88    70    56    45    38    31    29    24    25    27    27    25    23    21    15
    19    33    62    96   101    83    69    60    48    44    40    34    30    31    31    30    27    25    21    14
    23    32    49    69    79    66    59    53    56    57    55    50    53    49    49    40    35    28    23    14
    33    25    22    30    45    49    49    51    56    62    71    69    65    70    65    53    40    31    23    15
    26    11     1     4    13    19    24    36    47    57    71    71    74    81    76    61    44    33    26    19
    17    14     6    -2    -1     4    13    24    35    49    69    83    86    95    89    73    48    32    24    19
    18    14     2   -10   -13    -4     6    15    30    42    59    77   101   109    99    77    54    36    27    26
     9    -7   -22   -22   -15    -4    10    19    25    35    47    68    92   105   104    85    61    41    32    29
   -13   -26   -30   -23   -18    -8     8    22    30    33    40    58    81   103   110    97    69    48    41    34
   -16   -31   -27   -19   -17   -11    10    25    33    35    38    49    70    94   109    97    79    59    50    42
     6   -20   -24    -8     3     7    10    21    26    31    38    49    63    86   102   100    84    69    57    43
    33    -1   -14     7    27    21    13    15    17    27    35    42    59    79    94    91    86    72    59    39
    60    28     8    18    36    30    25    25    26    25    27    31    50    66    79    84    73    66    48    30
    78    63    46    41    50    39    39    22    21    16    16    20    38    35    48    53    54    47    34    20
    77    84    75    69    66    48    27    13    11     0     0    12    14    13    22    26    30    26    20    12
    71    91    92    87    86    69    48    29    15     6     0     0     0     4     7     8    11    14    15    14
    60    79    90    87    91    81    72    55    45    27    18    11     5     4     2     1     5    13    20    20
    45    63    78    83    81    82    70    70    67    62    40    27    16     8     7     7     8    16    24    27
    34    49    62    71    82    82    76    76    78    77    59    40    24    12    12    13    18    24    31    36
    28    40    54    67    77    80    76    72    68    56    46    37    25    15    24    26    30    34    40    47
    21    36    52    57    60    67    65    56    44    37    35    31    27    27    30    39    41    46    54    52
    29    43    49    54    49    50    49    40    34    29    26    24    23    25    31    42    51    61    65    57
    39    45    44    38    39    43    41    36    27    19    12    10    11    13    21    37    57    71    73    58
```

Fig. 2-6.

```
*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*
*X-COMPONENTS OF WIND DATA*

   51    52    60    84    95    75    47    35    32    31    31    32    37    34    31    31    29    21    17    21
   30    38    48    62    72    60    54    48    46    44    46    45    42    38    41    36    31    25    23    25
   15    25    34    39    36    37    36    39    39    39    44    49    51    51    52    46    37    30    27    32
   38    43    39    40    24    27    19    20    26    34    43    51    55    58    56    50    42    35    34    40
   65    62    48    34    21     5     1     4    17    35    50    55    56    56    57    52    46    41    41    41
   61    62    54    35    13    -8   -21   -14     7    38    62    69    63    59    58    54    47    48    44    39
   55    63    62    51    20     0   -16   -15     3    35    61    67    64    60    59    58    57    49    39    38
   56    74    75    60    36    15    -2    -3    12    36    55    58    59    58    57    55    53    40    37    27
   50    71    85    78    59    37    18    20    29    36    49    49    48    48    56    54    40    37    32    23
   41    66    85    94    91    63    44    38    44    37    37    37    46    44    42    38    34    30    26    18
   40    58    81   102   105    81    60    47    48    38    37    36    35    32    31    29    31    24    21    15
   31    46    76   105   114    93    72    58    46    38    33    29    26    27    29    29    27    25    20    13
   17    37    72   107   108    83    66    57    46    41    38    32    30    31    32    31    29    26    21    13
   20    42    57    81    88    68    60    52    54    54    51    46    47    42    42    37    33    27    22    14
   39    38    32    40    59    60    52    61    57    61    59    67    63    59    53    50    39    31    23    15
   39    24    11     9    20    25    25    38    49    58    61    61    73    71    74    61    47    32    27    20
   33    27    11     2     3     9    17    30    38    52    61    81    83    81    86    70    54    36    25    23
   32    24     7    -8    -9     0     8    17    31    45    62    76    84    99    93    78    62    42    29    26
   23     2   -20   -24   -13     0    10    12    22    35    49    69    89    98    96    84    69    48    39    34
    1   -20   -28   -23   -16    -6     6    15    22    31    42    60    83   100   101    94    73    60    49    35
   -3   -23   -25   -18   -15    -6    10    23    30    32    39    52    72    91    99    91    81    68    56    44
   15   -10   -20    -6    10    14    21    21    26    30    38    51    69    85    95    93    84    71    58    45
   37    10    -3    15    29    22    12    14    15    25    28    44    62    80    89    85    81    70    57    42
   62    37    19    31    35    35    32    23    24    12    13    30    48    64    74    74    67    62    49    34
   80    69    51    52    60    49    37    20    20    -1    -1    18    35    33    43    47    49    45    37    27
   84    90    82    71    68    57    38    13    -1     0     0    -1    14    12    20    24    29    28    24    18
   72    99   100    89    82    70    49    30    15     4     0     0     0     5     8    10    13    16    19    17
   63    82    92    89    87    86    69    60    40    28    18     9     6     5     4     5     7    15    18    22
   55    63    76    92    91    88    84    80    62    53    35    17    15     7     6     6    12    19    26    25
   43    53    68    87   108   102    98    80    76    69    51    35    22    10    10    19    16    22    28    32
   28    46    68    87    97    96    93    81    70    62    44    34    23    23    22    32    28    31    35    37
   21    42    66    71    72    70    71    62    50    45    38    33    32    26    35    37    44    45    45    44
   34    48    62    62    56    59    57    49    44    39    34    30    27    28    34    42    48    52    52    48
   53    49    48    50    53    52    48    41    35    25    21    13    13    16    26    38    51    59    59    50
```

Fig. 3-1.

```
* TEMPERATURES OF GRID POINTS IN FLIGHT LEVELS *

  34  35  36  37  37  38  39  40  42  44  45  46  46  47  49  49  48  47  47  48
  35  35  36  37  38  38  39  40  41  43  44  45  46  47  48  48  48  47  48  48
  36  36  36  37  38  39  39  40  41  42  44  44  45  46  47  47  47  47  48  47
  36  37  38  40  41  41  41  42  42  43  43  44  45  45  46  47  47  47  47  47
  35  38  41  43  44  44  44  43  42  42  43  44  44  45  46  47  47  47  47  48
  36  38  41  44  45  45  44  43  42  42  43  44  44  45  46  47  48  48  48  48
  36  38  40  42  44  45  44  44  43  43  44  44  45  46  47  48  49  49  49  50
  35  37  39  41  43  44  45  45  45  45  45  45  46  47  48  49  50  51  51  52
  36  37  39  41  43  45  45  46  46  47  47  47  48  49  49  51  51  52  53  53
  34  35  37  40  44  46  46  47  47  47  47  47  48  49  50  50  51  52  53  55
  32  34  36  39  43  45  47  47  48  48  48  48  48  48  50  51  52  53  54  55
  33  34  36  39  41  44  47  48  49  50  50  50  51  51  52  53  54  55  56  56
  34  35  36  38  40  43  45  47  49  50  51  52  53  53  54  55  56  56  57  57
  34  35  37  37  38  41  44  47  49  50  51  52  52  52  53  55  56  57  57  57
  35  37  37  37  38  39  41  44  45  46  48  50  49  49  49  52  54  55  55  56
  36  37  38  38  38  38  39  40  41  42  44  44  45  45  45  48  51  52  53  55
  35  36  38  39  40  40  39  39  39  39  40  42  42  42  43  45  47  49  51  54
  37  39  40  41  41  42  41  40  39  39  39  39  40  41  41  43  45  48  50  54
  39  40  41  41  40  40  40  40  40  39  38  38  39  39  41  42  44  47  50  55
  39  39  39  38  38  38  39  39  39  39  39  39  39  39  40  42  44  46  50  56
  40  40  39  38  37  38  39  39  39  39  39  39  39  39  40  42  44  47  51  56
  40  40  39  38  38  39  39  40  40  40  40  39  39  40  41  43  44  48  52  57
  39  39  39  39  39  39  39  39  40  40  40  40  40  41  43  44  47  49  53  57
  38  39  39  39  39  39  39  40  40  40  41  41  42  43  45  47  49  51  54  56
  37  38  39  41  43  43  42  43  44  44  44  43  44  46  48  49  51  53  55  56
  35  37  39  41  44  46  47  48  48  48  48  48  48  49  50  52  53  54  55  55
  35  36  38  40  43  47  50  52  53  53  52  52  51  52  53  53  54  54  54  55
  37  37  38  41  43  46  49  51  52  54  54  53  53  53  53  53  54  53  54  54
  38  38  38  39  42  45  47  49  50  52  52  52  53  53  53  53  53  53  53  54
  39  39  39  39  41  43  45  47  49  50  51  51  51  51  51  52  52  52  53  53
  39  39  38  39  40  41  43  44  46  47  48  49  50  50  50  51  52  52  52  52
  38  37  37  38  39  40  41  43  44  46  47  48  49  50  51  52  52  52  52  51
  38  38  38  39  40  41  42  43  45  46  47  49  50  51  52  52  53  52  51  51
  39  40  41  42  42  43  44  45  47  48  49  51  52  53  54  54  53  53  51  50
```

Fig. 3-2.

```
* TEMPERATURES OF GRID POINTS IN FLIGHT LEVELS *

  45  46  47  47  46  46  47  48  50  51  51  52  51  52  52  51  50  50  50  50
  45  46  46  47  47  47  48  49  50  51  52  52  53  53  53  52  51  50  50  50
  46  46  47  47  47  48  49  50  51  52  53  53  54  54  54  53  51  50  50  49
  45  47  48  49  50  49  50  51  52  52  53  54  54  55  54  54  52  50  49  49
  44  47  50  52  51  51  51  51  51  52  53  54  54  55  55  54  52  50  49  49
  45  47  50  52  52  53  53  51  51  52  54  55  54  55  55  54  53  50  49  49
  45  46  49  51  53  54  54  52  52  53  55  55  55  56  56  55  53  51  50  50
  46  47  48  50  52  54  54  54  54  55  55  55  56  55  55  54  52  51  51  51
  46  47  48  49  53  55  55  55  55  55  55  55  55  54  53  53  52  52  52  52
  45  46  46  49  52  53  54  53  53  53  53  53  52  52  52  52  52  52  52  53
  43  44  45  47  50  52  53  52  52  52  51  51  50  50  51  51  52  52  53  53
  44  44  45  46  48  50  52  53  53  52  52  52  51  51  51  52  53  53  53  54
  44  44  45  45  47  49  52  52  53  53  53  54  53  53  53  53  53  54  54  54
  42  44  45  45  45  48  51  52  52  52  53  53  53  52  52  53  54  54  54  55
  42  44  45  45  45  47  49  50  51  51  51  52  51  49  50  52  53  54  54  55
  43  45  47  47  48  47  48  48  49  50  50  50  48  47  48  49  51  52  53  54
  44  46  47  48  48  48  48  47  47  48  48  49  47  47  47  49  49  51  52  54
  47  49  49  50  50  50  49  49  48  47  47  47  47  47  47  48  50  50  51  54
  49  50  50  50  49  49  49  49  48  47  46  46  47  47  49  49  50  51  51  55
  48  49  49  48  48  48  48  48  48  48  47  47  47  47  49  50  50  51  52  56
  48  49  49  48  47  47  48  48  48  48  48  47  47  47  48  50  50  52  53  56
  49  50  49  47  47  47  47  47  47  47  47  47  47  48  49  50  50  51  53  57
  48  48  49  48  48  48  47  47  47  47  47  48  48  49  50  50  51  51  53  56
  46  47  48  49  49  49  48  49  49  48  48  49  49  50  50  51  51  52  54  56
  45  46  48  50  50  49  48  49  50  49  49  49  49  50  50  51  52  53  54  56
  43  44  46  49  50  49  49  49  50  50  50  49  49  50  51  52  53  54  54  56
  43  43  45  47  49  51  52  52  52  52  51  51  51  51  52  53  54  55  55  55
  47  46  46  48  49  52  53  54  53  53  52  52  52  52  54  55  56  55  55  55
  48  48  47  48  50  51  54  54  55  54  54  53  53  53  55  57  57  56  55  55
  47  47  47  47  49  51  53  55  56  56  56  56  56  55  56  58  58  57  55  54
  47  46  46  47  48  49  51  52  54  56  57  57  58  59  58  59  59  57  55  53
  48  46  46  46  46  47  48  50  52  54  56  57  58  59  60  60  59  57  54  53
  47  47  47  48  48  49  50  51  52  53  54  56  58  60  60  60  59  56  53  52
  47  47  48  50  51  52  52  52  53  54  55  56  58  59  59  59  58  56  53  51
```

Fig. 3-3.

```
 * TEMPERATURES OF GRID POINTS IN FLIGHT LEVELS *

     55   56   57   57   55   56   56   57   58   58   56   54   53   53   52   51   51   50   50   50
     55   56   56   57   56   56   57   57   58   59   59   57   56   55   54   53   52   51   51   50
     55   56   57   57   57   58   58   59   60   61   62   62   60   58   57   55   53   52   51   50
     54   56   57   58   58   57   58   59   60   62   63   63   62   61   58   56   54   52   51   50
     53   56   57   58   56   56   57   58   59   61   63   64   63   62   59   56   54   52   50   50
     54   55   57   58   57   58   59   59   59   62   64   65   64   63   59   56   53   51   50   49
     55   55   57   59   61   61   61   60   61   63   66   66   65   63   59   56   53   50   49   49
     56   56   56   58   61   62   61   62   62   64   65   64   63   59   57   54   52   50   50   50
     56   56   56   58   60   61   61   61   61   60   60   59   57   56   54   53   51   50   50   50
     55   56   56   57   58   57   57   57   56   56   55   55   54   53   52   51   51   51   51   51
     54   54   55   55   56   55   55   54   54   54   53   53   52   51   51   51   51   51   51   51
     54   54   54   53   54   54   54   54   54   53   53   53   52   51   50   50   51   51   51   52
     54   54   54   53   54   54   54   54   54   54   53   53   53   52   51   51   51   52   52   52
     52   53   53   53   52   54   54   53   53   53   53   53   52   52   51   51   52   52   52   53
     51   52   53   54   54   55   55   55   54   53   53   52   50   50   50   51   52   52   53   53
     52   54   56   57   57   57   57   57   57   57   54   52   50   49   50   51   51   52   53   53
     54   56   57   57   57   56   56   56   56   56   56   55   52   50   50   51   51   52   52   53
     57   58   58   58   58   58   57   57   56   55   55   55   54   52   52   52   52   52   53   54
     58   59   59   58   58   58   57   57   56   55   55   55   55   55   56   54   54   53   53   54
     57   58   58   58   57   57   57   57   57   56   56   55   55   56   57   57   55   53   53   55
     55   57   58   57   56   56   56   57   57   57   57   56   55   56   57   57   55   54   54   55
     55   57   58   57   55   55   55   55   55   55   55   55   55   55   56   56   54   53   54   55
     54   55   57   58   58   57   56   55   55   55   55   56   56   56   56   54   53   52   53   56
     53   54   56   58   59   58   58   58   58   57   57   57   57   56   54   53   52   52   53   56
     52   54   56   57   54   53   53   53   54   54   54   54   53   52   52   52   52   52   54   56
     51   52   55   55   53   51   51   51   51   51   51   51   50   51   51   52   53   53   54   56
     52   51   52   53   53   52   51   51   51   51   50   50   50   51   52   53   54   54   55   56
     56   54   54   54   54   54   53   52   51   51   50   50   50   51   53   54   55   55   55   55
     57   57   56   56   57   56   57   56   55   54   53   52   52   53   54   56   56   56   55   55
     54   54   55   56   57   59   61   60   59   59   58   56   56   56   57   58   58   57   55   55
     55   54   53   55   56   58   59   61   62   63   63   61   60   60   60   61   60   58   56   54
     57   55   54   53   53   54   55   58   61   63   64   64   63   62   62   62   61   58   55   54
     56   55   56   57   58   58   58   59   59   60   59   60   61   62   62   61   60   57   54   53
     54   53   55   57   58   58   58   57   57   57   58   58   59   60   60   60   58   56   54   52
```

Fig. 3-4.

* TEMPERATURES OF GRID POINTS IN FLIGHT LEVELS *

34	35	36	37	40	42	43	44	45	45	46	46	47	47	48	49	49	48	47	47
35	35	36	37	38	39	40	42	42	43	44	45	46	47	47	48	49	49	48	47
36	36	36	37	37	38	39	40	40	41	42	44	45	46	47	47	48	48	48	47
36	37	38	39	39	40	40	41	41	41	42	43	45	45	46	47	47	48	48	47
35	38	41	42	42	42	42	42	42	42	43	44	44	45	46	46	47	48	47	47
36	39	42	44	45	45	44	43	42	42	43	44	44	45	46	46	47	47	47	47
36	38	40	43	44	45	44	43	43	43	44	44	45	46	47	46	47	46	47	49
35	37	39	41	43	44	44	43	44	44	45	45	46	47	47	47	47	47	49	50
36	37	38	40	42	43	44	45	45	46	46	47	48	49	48	48	48	49	50	52
34	35	36	39	41	43	45	46	47	48	48	49	49	50	50	50	50	52	52	53
32	34	36	38	41	43	45	46	47	48	48	49	50	50	50	51	52	53	54	54
33	34	36	38	41	44	45	45	46	47	48	48	49	50	51	52	54	55	55	55
34	35	37	38	41	44	45	46	47	48	49	50	50	51	52	53	55	56	56	56
34	36	38	39	40	42	44	46	47	49	49	50	51	52	52	54	56	57	57	57
35	37	38	38	39	41	42	44	45	47	49	50	51	52	53	54	56	57	58	57
36	37	37	37	38	39	40	41	43	45	47	49	50	52	52	53	55	56	57	57
35	36	38	39	39	39	39	39	40	42	44	47	48	50	50	50	51	54	56	57
37	38	40	41	41	41	41	40	40	40	41	43	44	46	46	46	48	52	54	56
39	40	40	40	39	39	40	41	41	40	40	41	42	43	44	44	47	50	53	55
39	39	39	38	38	38	39	40	40	40	39	39	40	42	43	45	46	49	53	55
40	40	39	38	39	39	39	40	40	40	39	39	40	41	43	45	47	50	52	54
40	40	39	39	40	40	39	39	40	40	39	39	40	41	43	44	47	49	51	53
39	40	39	39	40	40	40	40	40	39	39	39	40	41	43	44	46	48	51	53
38	39	39	39	38	39	39	39	40	40	40	41	41	42	43	44	46	48	50	53
37	38	39	39	38	39	41	42	42	42	43	43	43	43	44	46	48	49	51	54
35	37	39	40	40	42	45	46	46	46	47	47	47	47	48	49	50	52	54	55
35	36	39	41	42	44	47	48	50	50	51	51	51	51	52	53	53	54	55	56
37	37	38	40	44	47	49	50	52	53	54	54	53	53	53	54	54	55	55	55
38	38	38	39	41	43	46	48	50	52	54	54	53	52	52	53	54	54	55	55
39	39	39	38	39	41	43	46	48	50	52	52	51	51	51	52	52	53	54	55
39	39	38	38	38	39	41	44	46	48	49	50	50	50	50	51	52	53	54	54
38	37	37	38	38	39	41	43	44	46	47	48	48	49	49	50	51	52	53	53
38	38	39	39	39	40	41	43	44	45	46	48	49	49	49	50	51	52	53	53
39	41	41	41	41	42	43	44	46	48	49	50	51	51	51	51	51	52	52	52

Fig. 3-5

• TEMPERATURES OF GRID POINTS IN FLIGHT LEVELS •

45	46	47	47	49	51	51	51	51	52	52	52	52	51	51	51	51	50	49	49
45	46	46	46	47	47	48	50	50	51	52	53	53	53	52	52	51	51	50	50
46	46	46	46	46	47	48	49	49	50	51	51	52	52	52	52	51	51	50	50
45	47	48	48	47	48	50	50	51	51	52	52	52	52	52	51	51	50	50	50
44	48	50	50	49	50	50	51	51	52	53	53	53	53	52	51	50	50	49	49
45	48	51	52	52	53	53	51	51	52	53	53	53	53	52	51	50	49	49	49
45	46	49	52	53	54	53	52	52	53	54	54	54	53	53	51	49	48	48	49
46	47	48	50	52	53	53	53	54	55	55	55	55	54	52	50	49	49	49	50
46	47	48	49	51	53	54	54	55	55	56	55	56	54	52	50	50	50	50	51
45	46	46	48	49	52	53	55	56	56	56	55	54	53	52	51	51	51	52	52
43	44	46	47	49	51	52	52	53	54	53	53	52	52	51	51	52	52	52	53
44	44	45	47	49	50	51	51	51	51	51	50	50	50	51	51	52	53	53	53
44	44	45	47	49	50	51	52	52	52	52	52	52	51	52	52	53	53	53	54
42	44	46	47	49	50	52	52	54	54	54	54	54	54	53	53	54	54	54	54
42	44	46	47	47	49	50	52	54	54	54	55	55	55	55	54	54	54	55	55
43	45	46	46	46	47	48	49	51	53	54	55	55	55	54	53	53	54	55	56
44	46	47	48	47	47	47	48	48	50	52	53	53	53	52	50	51	53	55	56
47	48	49	49	49	49	49	48	48	48	49	50	50	50	48	48	49	52	55	56
49	50	50	49	48	48	49	49	49	48	48	48	49	49	48	47	48	51	55	56
48	49	49	48	48	48	48	49	48	48	47	48	48	49	48	47	48	51	55	57
48	49	49	48	47	48	48	48	48	48	48	47	48	49	49	49	49	52	54	56
49	50	49	48	47	48	48	48	48	48	47	47	48	49	49	50	50	52	53	55
48	48	49	48	48	48	48	47	47	47	47	47	48	49	49	50	51	51	53	54
46	47	48	49	49	47	47	47	47	47	48	49	49	49	49	50	51	51	52	54
45	47	48	48	47	48	49	49	49	48	49	49	49	49	49	50	50	51	52	54
43	45	47	47	46	48	49	49	49	49	49	49	49	49	50	51	52	52	54	55
43	44	46	47	48	49	50	50	50	50	50	50	50	51	51	53	54	55	55	56
47	47	46	47	50	52	54	55	55	54	53	52	51	51	52	54	56	57	57	57
48	48	47	46	48	51	54	56	57	57	56	54	53	52	53	54	56	58	58	57
47	47	47	47	48	49	51	53	56	57	58	57	55	54	55	56	57	58	59	58
47	46	46	47	48	48	50	52	54	56	57	57	57	57	57	57	58	59	59	58
48	46	45	47	47	48	50	51	53	54	55	56	57	58	58	58	59	60	59	58
47	47	46	47	48	49	50	51	52	54	55	57	58	59	59	59	60	60	59	57
47	47	47	48	49	50	51	52	53	54	55	57	58	60	60	60	60	59	58	56

Fig. 3-6.

```
• TEMPERATURES OF GRID POINTS IN FLIGHT LEVELS •

  55  56  57  57  58  58  56  55  54  54  55  54  53  52  51  50  49  49  49  49
  55  56  56  56  56  56  57  57  58  58  59  59  57  55  53  51  50  50  50  50
  55  56  56  56  55  56  57  58  58  59  59  59  59  57  54  52  51  50  50  50
  54  56  57  57  57  57  59  60  61  61  61  60  60  58  55  52  51  51  50  50
  53  56  57  57  56  56  58  59  61  62  63  62  61  59  55  53  51  50  50  50
  54  56  58  57  56  57  59  59  60  62  63  63  61  59  55  53  51  50  50  49
  55  55  57  59  60  60  60  60  61  63  63  63  61  58  55  53  51  50  50  49
  56  56  57  58  60  61  61  62  64  65  64  63  59  57  54  52  51  50  50  49
  56  56  57  58  59  60  61  63  65  65  63  60  58  55  53  52  51  51  50  50
  55  56  56  57  57  58  58  61  62  60  59  57  55  54  53  52  51  51  50  50
  54  55  56  56  57  57  56  57  57  56  55  54  53  52  51  51  51  51  51  51
  54  54  55  56  57  55  55  55  54  54  53  52  51  51  50  50  50  51  51  51
  54  54  54  56  57  55  54  55  54  54  54  53  52  52  51  51  51  51  51  52
  52  53  54  55  57  57  58  57  57  56  55  55  54  54  53  52  52  52  52  52
  51  53  54  55  56  57  59  60  60  58  56  55  55  55  54  53  52  52  53  53
  52  53  55  55  55  56  57  58  59  59  57  56  55  55  54  53  52  52  53  54
  54  56  57  57  56  56  56  56  57  57  58  56  55  53  52  51  51  52  53  54
  57  58  58  57  57  57  57  56  56  57  57  57  54  52  50  50  50  52  54  55
  58  59  58  58  57  58  58  57  57  57  57  57  56  53  50  49  49  51  54  55
  57  58  58  57  57  58  58  58  57  57  56  56  56  55  51  49  50  52  54  56
  55  57  58  57  56  58  58  58  58  57  57  57  57  56  53  51  51  52  54  56
  55  57  58  57  55  57  58  58  57  56  56  56  56  56  53  52  52  53  54  55
  54  56  58  57  56  57  57  55  55  55  56  55  55  55  53  53  53  53  54  55
  53  55  58  59  59  57  55  54  54  55  56  56  56  55  53  53  53  53  54  55
  52  54  56  56  56  55  55  55  55  53  53  53  53  53  52  52  52  53  53  54
  51  53  54  53  51  52  52  51  51  51  50  50  51  51  52  52  52  53  53  54
  52  52  52  51  51  52  52  51  51  51  50  50  50  50  51  52  53  54  54  55
  56  55  54  54  53  54  54  54  54  53  52  51  50  50  51  53  54  55  56  56
  57  56  55  54  55  57  57  58  57  56  54  52  51  51  52  53  55  57  57  57
  54  54  54  55  57  58  59  58  58  58  57  56  54  54  55  56  57  58  58  58
  55  53  54  56  57  58  59  59  59  59  60  60  59  59  59  59  59  59  59  58
  57  55  54  56  57  58  59  60  61  62  64  65  65  65  64  62  61  61  60  59
  56  55  54  55  56  58  59  60  61  62  63  65  66  66  66  64  63  61  60  58
  54  52  52  54  56  57  58  59  59  58  59  60  61  63  64  64  63  61  59  57
```

Fig. 4.

```
• PERFORMANCE DATA FOR  DC10 •

• SPECIFIC RANGE •

     757  748  740  731  722  713  704  694  684  674  664  654  643  633  623  612
     856  842  828  814  799  784  769  754  739  722  706  690  674  657  640  623
     897  880  862  844  825  807  787  768  748  728  708  688  668  648  627  606
     601  590  579  568  557  545  534  522  511  499  488  476  465  453  440  428   -1
     607  590  573  556  538  520   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1
     584   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1   -1

• MAX. WEIGHT •

   269437270344271251272158*00000*00000*00000
   227705228613229520230427231334232242233149
   188696189150189604190057190511190964191418
```

Fig. 5-1.

```
* CLIMB TIME *

          160   162   163   165   167   167   168   182   198
          178   180   182   183   185   187   188   202   222
          205   207   208   210   212   213   215   228   250
          165   167   168   170   172   172   173   187   205
          185   187   188   190   192   193   195   208   230
          212   213   215   217   220   222   223   238   260
          170   172   173   175   177   178   180   193   212
          190   192   193   195   197   198   202   215   238
          220   222   223   225   228   230   232   247   270
          175   177   178   180   182   183   185   200   220
          197   198   200   202   203   205   208   223   247
          228   230   232   233   237   238   240   257   282
          180   182   183   185   187   188   190   207   227
          203   205   207   208   210   212   215   230   255
          237   238   240   243   245   247   250   267   293
          187   188   190   192   192   193   197   212   235
          210   212   213   215   217   218   222   238   265
          245   247   250   252   255   257   260   277   305
          192   193   195   197   198   200   202   218   242
          217   218   220   222   225   227   228   247   273
          255   257   260   262   263   267   270   288   318
          197   198   200   202   203   205   208   227   250
          223   225   228   230   232   233   237   255   283
          265   267   270   272   275   277   280   298   332
```

Fig. 5-2.

```
 203   205   207   208   210   212   215   233   258
 230   233   235   237   240   242   243   263   295
 275   278   280   283   285   288   292   312   347
 208   210   213   215   217   218   220   240   268
 238   240   243   245   247   250   252   273   305
 287   290   292   295   298   300   303   325   362
 215   217   218   222   223   225   227   248   277
 247   248   252   253   255   258   260   283   317
 300   302   305   308   312   313   317   340   378
 222   223   225   228   230   232   235   257   287
 255   257   260   262   263   267   270   293   330
 313   317   320   322   325   328   332   355   397
 228   230   232   235   237   238   242   265   297
 263   265   268   270   273   275   278   303   342
 328   332   335   338   342   345   348   373   418
 235   237   240   242   243   247   248   273   307
 272   275   277   280   283   285   288   313   355
 347   350   353   357   360   363   367   393   443
 242   245   247   248   252   253   257   282   318
 282   285   287   290   293   295   298   325   370
  -1    -1    -1    -1    -1    -1    -1    -1   472
 250   252   255   257   258   262   265   292   330
 292   295   298   300   303   307   310   338   385
  -1    -1    -1    -1    -1    -1    -1    -1    -1
 257   260   262   265   267   270   273   302   342
 303   305   308   312   315   317   322   352   402
  -1    -1    -1    -1    -1    -1    -1    -1    -1
```

Fig. 5-3.

265	268	270	273	275	278	282	312	355
315	317	320	323	327	330	333	365	420
-1	-1	-1	-1	-1	-1	-1	-1	-1
273	277	278	282	285	287	290	322	368
327	330	333	337	340	343	347	380	440
-1	-1	-1	-1	-1	-1	-1	-1	-1
282	285	288	290	293	297	300	333	383
340	343	347	350	353	357	362	397	462
-1	-1	-1	-1	-1	-1	-1	-1	-1
292	295	297	300	303	305	310	345	398
355	358	362	365	368	372	377	415	487
-1	-1	-1	-1	-1	-1	-1	-1	-1
300	303	307	310	312	315	320	358	415
372	375	378	382	385	390	393	435	513
-1	-1	-1	-1	-1	-1	-1	-1	-1
310	313	317	320	322	325	330	370	432
390	393	397	402	405	408	413	457	545
-1	-1	-1	-1	-1	-1	-1	-1	-1
320	323	327	330	333	337	340	385	450
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
332	335	338	342	345	348	352	400	472
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
343	347	350	353	357	360	365	415	493
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1

Fig. 5-4.

355	358	362	367	370	373	378	433	520
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
368	373	377	380	383	387	392	453	548
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
383	387	392	395	398	403	408	473	582
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
400	403	408	412	415	420	425	498	620
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
418	422	427	430	435	438	445	525	667
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
438	443	447	452	455	460	467	557	725
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
462	467	472	477	480	485	492	595	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1

Fig. 6-1.

* CLIMB DISTANCE *

52	53	54	55	56	57	58	64	72
60	61	62	64	65	66	67	74	83
72	73	74	76	77	79	80	87	97
54	55	56	57	58	59	60	66	74
62	64	65	66	67	69	70	77	86
75	76	78	79	81	82	84	91	102
56	57	58	59	60	61	62	69	77
65	66	67	69	70	71	73	80	90
78	79	81	83	84	86	88	95	106
58	59	60	61	62	63	64	71	80
67	69	70	71	73	74	76	83	94
81	83	84	86	88	90	91	99	111
60	61	62	63	64	65	67	74	84
70	71	73	74	75	77	78	86	98
85	86	88	90	92	93	95	103	116
62	63	64	65	66	68	69	77	87
72	74	75	77	78	80	81	89	102
88	90	92	94	96	98	100	108	122
64	65	66	68	69	70	72	80	90
75	77	78	80	81	83	85	93	106
92	94	96	98	100	102	104	113	127
66	67	69	70	71	72	74	82	94
78	80	81	83	84	86	88	97	110
96	98	100	102	104	106	109	118	133

Fig. 6-2.

```
  68   70   71   72   74   75   77   85   97
  81   82   84   86   87   89   91  100  115
 118  121  124  126  129  131  134  144  162
  71   72   73   75   76   78   79   88  101
  84   86   87   89   91   93   95  104  120
 106  108  110  112  115  117  119  130  147
  73   74   76   77   79   80   82   92  105
  87   89   91   92   94   96   98  108  125
 111  113  116  118  120  123  125  136  155
  75   77   78   80   82   83   85   95  109
  90   92   94   96   98  100  102  113  130
 117  119  122  124  127  129  132  143  164
  78   80   81   83   84   86   88   98  113
  94   96   98  100  102  104  106  117  136
 123  126  129  131  134  136  139  151  173
  81   82   84   86   87   89   91  102  118
  98  100  102  104  106  108  110  122  142
 131  134  137  139  142  145  148  161  185
  83   85   87   89   90   92   94  106  123
 101  104  106  108  110  112  115  127  148
  -1   -1   -1   -1   -1   -1   -1   -1  199
  86   88   90   92   93   95   97  110  128
 106  108  110  112  114  117  119  133  155
  -1   -1   -1   -1   -1   -1   -1   -1   -1
  89   91   93   95   97   99  101  114  133
 110  112  115  117  119  122  124  139  163
  -1   -1   -1   -1   -1   -1   -1   -1   -1
```

Fig. 6-3.

```
 93   94   96   98  100  102  104  118  138
115  117  120  122  124  127  130  145  171
 -1   -1   -1   -1   -1   -1   -1   -1   -1
 96   98  100  102  104  106  108  123  144
120  122  125  128  130  133  136  152  180
 -1   -1   -1   -1   -1   -1   -1   -1   -1
 99  101  103  105  108  110  112  128  151
126  128  131  134  136  139  142  159  190
 -1   -1   -1   -1   -1   -1   -1   -1   -1
103  105  107  109  112  114  116  133  157
132  135  137  140  143  146  149  167  201
 -1   -1   -1   -1   -1   -1   -1   -1   -1
107  109  111  113  116  118  121  138  165
139  142  145  148  151  153  157  176  214
 -1   -1   -1   -1   -1   -1   -1   -1   -1
111  113  115  118  120  122  125  144  173
147  150  153  156  159  162  166  187  229
 -1   -1   -1   -1   -1   -1   -1   -1   -1
115  117  120  122  125  127  130  151  181
 -1   -1   -1   -1   -1   -1   -1   -1  248
 -1   -1   -1   -1   -1   -1   -1   -1   -1
120  122  125  127  130  132  135  157  191
 -1   -1   -1   -1   -1   -1   -1   -1   -1
 -1   -1   -1   -1   -1   -1   -1   -1   -1
125  127  130  132  135  138  141  165  201
 -1   -1   -1   -1   -1   -1   -1   -1   -1
 -1   -1   -1   -1   -1   -1   -1   -1   -1
```

Fig. 6-4.

130	133	135	138	141	144	147	173	213
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
136	139	141	144	147	150	154	182	227
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
142	145	148	151	154	157	161	192	242
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
149	152	155	159	162	165	169	203	260
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
157	160	164	167	170	173	178	215	283
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
166	169	173	176	180	183	188	230	310
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
176	180	184	187	191	195	199	248	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1

Fig. 7-1.

```
* CLIMB FUEL *

          2893   2938   2983   3028   3073   3118   3167   3279   3424
          3080   3139   3190   3240   3290   3341   3395   3515   3677
          3317   3373   3430   3486   3543   3600   3660   3786   3961
          2970   3017   3064   3110   3157   3205   3255   3375   3528
          3177   3229   3281   3333   3386   3439   3495   3622   3795
          3419   3478   3537   3596   3656   3715   3778   3912   4098
          3049   3097   3146   3195   3244   3293   3346   3472   3634
          3266   3320   3375   3430   3484   3540   3598   3733   3916
          3524   3586   3647   3709   3772   3834   3900   4041   4240
          3129   3179   3230   3281   3332   3383   3438   3572   3743
          3357   3414   3471   3528   3586   3643   3704   3847   4042
          3633   3698   3763   3827   3892   3957   4026   4176   4387
          3211   3263   3316   3369   3422   3475   3533   3674   3856
          3452   3511   3570   3630   3690   3750   3813   3964   4171
          3747   3814   3882   3949   4017   4086   4157   4315   4541
          3295   3349   3404   3460   3515   3570   3630   3779   3972
          3549   3611   3672   3734   3797   3859   3926   4085   4305
          3866   3935   4006   4077   4148   4219   4294   4462   4702
          3381   3438   3495   3552   3610   3668   3729   3887   4092
          3649   3713   3778   3842   3907   3972   4042   4210   4445
          3989   4062   4136   4210   4284   4359   4437   4614   4872
          3470   3529   3588   3648   3708   3768   3832   3999   4216
          3753   3819   3886   3954   4021   4089   4161   4339   4590
          4119   4196   4273   4350   4428   4506   4588   4775   5051
```

Fig. 7-2.

```
3561  3622  3684  3746  3808  3871  3938  4114  4345
3860  3929  3999  4069  4139  4210  4284  4473  4740
4257  4337  4418  4498  4579  4660  4746  4944  5241
3655  3719  3783  3847  3912  3977  4046  4233  4479
3971  4043  4116  4188  4262  4335  4413  4613  4898
4403  4487  4571  4655  4740  4825  4915  5125  5444
3752  3818  3885  3952  4019  4086  4159  4356  4618
4086  4161  4237  4313  4389  4465  4546  4758  5063
4560  4647  4736  4824  4921  5001  5095  5317  5662
3852  3921  3990  4059  4129  4199  4274  4483  4762
4206  4285  4363  4442  4521  4600  4685  4909  5236
4729  4821  4913  5006  5099  5192  5290  5526  5899
3956  4027  4099  4171  4243  4315  4393  4616  4912
4331  4413  4494  4576  4658  4741  4829  5068  5418
4915  5012  5108  5205  5303  5400  5504  5754  6160
4063  4137  4211  4286  4361  4436  4517  4753  5070
4462  4547  4632  4717  4803  4888  4980  5233  5610
5124  5226  5328  5430  5532  5635  5743  6008  6455
4174  4205  4327  4405  4482  4560  4645  4895  5234
4599  4687  4775  4864  4953  5043  5138  5407  5813
  -1    -1    -1    -1    -1    -1    -1    -1  6799
4288  4368  4448  4528  4609  4690  4778  5043  5406
4743  4835  4927  5019  5112  5205  5304  5591  6030
  -1    -1    -1    -1    -1    -1    -1    -1    -1
4408  4490  4573  4657  4740  4824  4915  5198  5586
4896  4891  5087  5183  5279  5376  5480  5785  6260
  -1    -1    -1    -1    -1    -1    -1    -1    -1
```

Fig. 7-3

```
4532  4618  4704  4790  4877  4964  5059  5360  5777
5057  5157  5257  5357  5457  5558  5667  5992  6509
  -1    -1    -1    -1    -1    -1    -1    -1    -1
4662  4750  4840  4930  5019  5110  5209  5530  5978
5230  5334  5438  5542  5647  5752  5865  6213  6778
  -1    -1    -1    -1    -1    -1    -1    -1    -1
4797  4890  4982  5076  5169  5263  5365  5709  6192
5417  5525  5634  5742  5852  5962  6079  6452  7073
  -1    -1    -1    -1    -1    -1    -1    -1    -1
4938  5034  5130  5226  5323  5421  5527  5895  6417
5617  5731  5844  5958  6072  6187  6310  6710  7398
  -1    -1    -1    -1    -1    -1    -1    -1    -1
5076  5175  5275  5376  5476  5577  5687  6084  6650
5830  5948  6067  6186  6305  6424  6553  6984  7754
  -1    -1    -1    -1    -1    -1    -1    -1    -1
5222  5325  5429  5532  5636  5741  5856  6283  6899
6070  6194  6318  6442  6567  6692  6827  7291  8166
  -1    -1    -1    -1    -1    -1    -1    -1    -1
5376  5483  5590  5698  5806  5914  6034  6496  7078
  -1    -1    -1    -1    -1    -1    -1    -1  8661
  -1    -1    -1    -1    -1    -1    -1    -1    -1
5539  5650  5762  5874  5986  6098  6222  6724  7463
  -1    -1    -1    -1    -1    -1    -1    -1    -1
  -1    -1    -1    -1    -1    -1    -1    -1    -1
5713  5828  5943  6060  6176  6293  6428  6969  7785
  -1    -1    -1    -1    -1    -1    -1    -1    -1
  -1    -1    -1    -1    -1    -1    -1    -1    -1
```

Fig. 7-4.

```
      5899  6019  6139  6260  6381  6502  6638  7230  8143
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
      6100  6225  6350  6475  6602  6728  6870  7526  8545
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
      6318  6448  6578  6710  6841  6972  7121  7846  9002
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
      6557  6693  6829  6965  7102  7239  7396  8202  9530
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
      6821  6963  7105  7247  7390  7534  7698  8603 10159
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
      7116  7264  7414  7563  7712  7862  8036  9062 10932
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
      7453  7609  7765  7792  8078  8235  8419  9602    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1
        -1    -1    -1    -1    -1    -1    -1    -1    -1


   *** INPUT PARAMETERS ***
          2    862 177356   3629      -1 259458 190964 190581       1       0
          0    115
```

Fig. 8.

ORIGINAL PAGE IS
OF POOR QUALITY

```
                    *** [illegible] ***


MIN [illegible]
[illegible]

                                                   34   1181/05   [illegible]

     [illegible]                                  [illegible]      TIME   CONS       [illegible]

                                                                                   122054

   102    47   350                  7                   131   24    24   4353   115095

   123    47   350    -5   465      9          52       153    4    25    592   115504

   104    54   350    -4   459     24         255       290   29    57   2744   115020

   105    57   350     0   464     42         150       340   18   114   1577   115545

    97    55   350     5   467     34         142       455   17   131   1055   111745

    91    54   350     5   469     36          94       137   12   143   1105   110547

    57    45   350     7   471     25         136       357   22   205   1999   108558

    77    59   350     7   471     22         215      1185   25   232   2577   105311

    53    55   350     1   465     32         454      1327   53   325   4387   101724

    52    75   390    -9   450     17         535      1875   45   410   5793    97923

    57    89   390   -12   450      5         527      2400   45   454   5552    94054

    21    97   390   -15   445     22         557      2957   45   557   5164    91410

    11    95   390   -15   449     20         175      3122   24   580   1722    89025

     5   105   390   -15   452     35          95      3215   12   612    954    85854

     2   115   390    -5   454     45         147      3362   18   635   1257    85457

   107   121   390    -5   457     45          21      3382    2   652    172    87425

     0                             35                  3400   19   651   1271    85154

TRIP FUEL [illegible]
CONS 9914049


      *** END OF RUN ***
```

Fig. 9.

```
                    *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE              ORIGINAL PAGE IS
FUEL                            OF POOR QUALITY

                                                  CH    1160768   0.00

NO. HEAD   FL  TMP  TAS  WIND     DIST  ACCD  TIME  ACCT  BURN  WEIGHT

                                                                124528

100   51  350             0             151    27    27  4752  119776

113   51  350    0  464   0        12   163     2    28   157  119619

108   68  350    0  464   0       233   396    30    58  2946  116673

103   62  350    0  464   0       150   546    19   118  1550  114623

 97   61  350    0  464   0       142   688    18   136  1723  113100

 91   59  350    0  464   0        99   787    13   149  1191  111909

 87   51  350    0  464   0       180   967    23   212  2128  109781

 75   69  350    0  464   0       189  1156    25   237  2198  107583

 54   64  350    0  464   0       415  1571    54   331  4885  102698

 50   71  390    0  463   0       373  1944    48   419  4150   98548

 38   80  390    0  463   0       345  2289    45   504  3370   95178

 21   90  390    0  463   0       336  2625    43   547  3340   91838

 11   99  390    0  463   0       185  2810    24   611  1796   90042

  6  108  390    0  463   0        93  2903    12   623   894   89148

  2  124  390    0  463   0       147  3050    19   642  1392   87756

103  124  390    0  463   0        30  3080     4   646   284   87472

  0                       0             3193    19   705  1271   86201

TRIP FUEL 38327
COST10357.21


        *** END OF RUN ***
```

Fig. 10.

```
                    *** DETAILED FLIGHT PLAN ***

MIN. TRACK IN SPACE
FUEL

                                                CR   1160768   0.00

NO. HEAD   FL   TMP   TAS  WIND     DIST  ACCD  TIME  ACCT  CONS  WEIGHT

                                                                  124477

100    50  350               0             150    26    26  4760  119717

110    50  350     0  462    0        12   162     2    28   158  119559

100    60  350     0  462    0       232   394    30    58  2932  116627

100    61  350     0  462    0       150   544    19   117  1843  114784

 97    66  350     0  462    0       142   686    18   135  1719  113065

 93    57  350     0  462    0        99   785    13   149  1186  111879

 87    49  350     0  462    0       180   965    23   212  2120  109759

 76    55  350     0  462    0       189  1154    24   236  2189  107570

 54    60  350     0  462    0       414  1568    54   330  4671  102899

 50    67  390     0  462    0       372  1940    48   415  4148   98751

 30    78  390     0  462    0       345  2285    45   503  3568   95183

 21    86  390     0  462    0       335  2620    44   547  3338   91845

 11    97  390     0  462    0       183  2803    24   611  1795   90050

  6   107  390     0  462    0        92  2895    12   623   892   89158

  2   122  390     0  462    0       147  3042    19   642  1391   87767

102   123  390     0  462    0        30  3075     4   646   284   87483

  0                              0          3188    19   705  1271   86212

TRIP FUEL 38265
COST10535.73


    *** END OF JOB ***
```

Fig. 11.

ORIGINAL PAGE IS
OF POOR QUALITY

```
              [illegible]

[illegible]
FLT.

                                                  [illegible]

[illegible]                              DIST   [illegible]  TIME  ACCT  [illegible]

                                                                        [illegible]
[illegible]                         29          10[illegible]   14    19  3[illegible]  11[illegible]
11[illegible]        3[illegible]  -1  47[illegible]  39     54   162    [illegible]   25   5[illegible]3  115[illegible]5
10[illegible]                      -1           54    252   396    27    52   2627   11[illegible]7[illegible]
10[illegible]  63    3[illegible]  -[illegible]   [illegible]5    150   546    17   109   1344   111[illegible]34
 97                                -[illegible]   71    142   [illegible]   1[illegible]   125   152[illegible]   110[illegible]6
 91    61    310   -[illegible]   [illegible]7   71    9[illegible]   765    11   135   1641   10[illegible]45
 37                                             1[illegible]   [illegible]   [illegible]   1[illegible]   155   17[illegible]   107[illegible]57
 77                                            116   216   1161    22   217   1[illegible]   10[illegible]
                                              116   467   15[illegible]    [illegible]   [illegible]   3[illegible]   101[illegible]46
                                              2[illegible]   3[illegible]   1[illegible]    37   355   3511    979.35
 37    72                                                              35   411   [illegible]   [illegible]17
 21                                                        336                   36                 [illegible]13
 11                                                        1[illegible]   2[illegible]    21    [illegible]7              [illegible]3
                                                            93   29[illegible]    1[illegible]                      [illegible]5
  2                                                   [illegible]   147                 17           14[illegible]
 10[illegible]                     -[illegible]         1[illegible]    51   [illegible]                  [illegible]
  [illegible]                                 11          [illegible]1[illegible]                 [illegible]    [illegible]              [illegible]13

T[illegible] FUEL [illegible]
COST   15.1[illegible]
```

Fig. 12.

```
                         *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FLTIME
                                                   CH    1190581   0.00

NO. HEAD   FL   TMP   TAS   WIND    DIST  ACCD  TIME  ACCT  BURN  WEIGHT
                                                                  231393
TOC   46  310                 29           120    20    20  5756  225637
113   46  310    -2   479     39      42   162     5    25   708  224929
108   66  310    -3   478     54     232   394    26    51  3741  221188
103   63  310    -4   477     65     150   544    17   107  2335  218853
 97   62  310    -5   477     71     142   686    15   123  2167  216686
 91   61  310    -5   476     81      99   785    11   134  1472  215214
 87   51  350     2   475    100     180   965    19   152  2689  212525
 77   54  310    -1   480    112     216  1181    22   214  2914  209611
 65   62  310     3   484    116     407  1588    41   255  5525  204086
 51   66  310     4   486    106     364  1952    37   332  4898  199188
 37   72  350     5   478    104     336  2288    35   406  4445  194743
 21   90  350     4   478     93     336  2624    35   441  4297  190446
 11   89  310     0   482     55     185  2809    21   502  2521  187925
  6  101  310    -1   480     44      93  2902    11   513  1330  186595
  2  118  350     4   477     41     147  3049    17   530  2105  184490
TOD  121  310    -4   477     15      51  3100     6   536   683  183807
  0                           12          3192    16   552  1203  182604

TRIP FUEL 48789
COST11129.51


          *** END OF RUN ***
```

Fig. 13.

```
                        *** DETAILED FLIGHT PLAN ***

   MIN TRACK IN SPACE
   FLTIME

                                                        CH    1190581  0.00

   NO. HEAD   FL  TMP   TAS  WIND    DIST- ACCD  TIME  ACCT  BURN  WEIGHT
                                                                   251697

 ✓ TOC  262  310              3           136    23    23  6611  245086
     1  262  310   -4   477   3       0   136  -100    23    -2  245088
     4  282  310   -1   480 -22     242   378    32    55  5065  240023
     8  279  310    2   484 -36     115   493    16   111  2475  237548
    17  292  310    3   485 -73      57   550     8   120  1313  236235
    30  268  310    6   487 -39     397   947    54   214  8260  227975
    44  268  310    7   489 -55     394  1341    55   308  8026  219949
    60  270  310    7   488 -64     394  1735    56   404  7912  212037
    75  265  310    2   484 -75     394  2129    59   503  7972  204065
    87  261  310   -2   480 -78     179  2308    27   530  3588  200477
    92  257  310   -4   477 -74     161  2469    24   554  3119  197358
    98  238  310   -5   476 -80      66  2535    10   604  1283  196075
   104  239  310   -4   477 -68     217  2752    32   636  4054  192021
   109  246  310   -3   478 -50     189  2941    27   702  3330  188691
   113  219  310   -2   479 -26     180  3121    24   726  2974  185717
   TOD  237  310   -1   480 -22      77  3198    10   736  1262  184455
 ✓ 115                      -17          3283    16   752  1203  183252

   TRIP FUEL 68445
   COST15383.19

              *** END OF RUN ***
```

Fig. 14-1.

```
                         *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FLTIME

                                                CH    1190581   0.00

NO. HEAD    FL   TMP   TAS   WIND     DIST   ACCD   TIME   ACCT   BURN   WEIGHT

                                                                          251699
  ~~2.3280,+01  0.0000,+00~~

TOC  262   310.              3              136     23     23   6611   245088
  2.3280,+01  0.0000,+00

  1  262   310    -4   477   3          0   136      0     23      0   245088
  2.3280,+01  1.9204,+04

  4  282   310    -1   480  -22       242   378     32     55   5065   240023
  2.3280,+01  9.4894,+03

  8  279   310     2   484  -36       115   493     16    111   2475   237548
  2.3280,+01  5.0680,+03

 17  292   310     3   485  -73        57   550      8    120   1313   236235
  2.3280,+01  3.2515,+04

 30  268   310     6   487  -39       397   947     54    214   8260   227975
  2.3280,+01  3.2738,+04

 44  268   310     7   489  -55       394  1341     55    308   8026   219949
  2.3280,+01  3.3479,+04

 60  270   310     7   488  -64       394  1735     56    404   7912   212037
  2.3280,+01  3.5169,+04

 75  265   310     2   484  -75       394  2129     59    503   7972   204065
  2.3280,+01  1.6322,+04

 87  261   310    -2   480  -78       179  2308     27    530   3588   200477
  2.3280,+01  1.4450,+04

 92  257   310    -4   477  -74       161  2469     24    554   3119   197358
  2.3280,+01  6.0011,+03

 98  238   310    -5   476  -80        66  2535     10    604   1283   196075
  2.3280,+01  1.9166,+04

104  239   310    -4   477  -68       217  2752     32    636   4054   192021
  2.3280,+01  1.5906,+04
```

Fig. 14-2.

```
109  246  310   -3  478   -50      189  2941     27    702  3330  188691
  2.3280,+01  1.4323,+04

113  219  310   -2  479   -26      180  3121     24    726  2974  185717
  1.5810,+01  6.1052,+03

TOD  237  310   -1  480   -22       77  3198     10    737  1262  184455
  1.5810,+01  6.1052,+03

115                       -17           3283     16    752  1203  183252

TRIP FUEL 68447
COST15383.64


      *** END OF RUN ***


PROGRAM COMPLETED IN   130,1426 CRU.

END OF RUN:    137 CRU    ACCOUNT: 510055,1     CARDS:     4 IN     0 OUT     P

  BASE CRU:    137.035    SEQ. NO.  122425      CPU TIME:   130.79 SEC.       E

  CORE FACTOR:  1.000     DATE: 22 MAY 81       TIME: 19:59:28                T

  TOTAL:       5852 CRU   LIMIT:       7500
```

Fig. 15.

```
                         *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FUEL
                                                      CH    1190581   0.00

NO. HEAD   FL  TMP   TAS  WIND    DIST  ACCD  TIME  ACCT  BURN  WEIGHT
                                                                232421
TOC   46  310              29      -     121    20    20  5796  226625
113   46  310   -2   479   39     41     162     5    25   694  225931
108   66  350    2   475   55    232     394    26    51  4622  221309
103   63  350    1   474   66    150     544    17   108  2407  218902
 97   62  350    1   474   74    142     686    16   123  2210  216692
 91   62  350    1   474   85     99     785    11   134  1496  215196
 87   51  350    2   475  102    180     965    19   153  2598  212598
 77   55  350    4   477  114    216    1181    22   215  3002  209596
 65   62  350    6   479  116    407    1588    41   256  5493  204103
 51   66  350    6   479  110    364    1952    37   333  4814  199289
 37   72  350    5   478  106    336    2288    35   407  4368  194921
 21   90  350    4   478   99    336    2624    35   443  4317  190604
 11   89  350    5   478   65    185    2809    21   503  2491  188113
  6  101  390    3   476   47     93    2902    11   514  1989  186124
  2  118  390    4   476   41    147    3049    17   531  2023  184101
TOD  121  390    4   477   22     26    3075     3   534   361  183740
  0                        17           3192    19   553  1271  182469

TRIP FUEL 49952
COST11320.37
```

Fig. 16.

```
                     *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FLTIME

                                                 CH    1190581   0.00

NO. HEAD   FL  TMP   TAS  WIND     DIST  ACCD  TIME  ACCT  BURN  WEIGHT

                                                                 233624

TOC   46  310              29            122    20    20  5843   227781

113   46  310   -2   479   39        40  162     5    25   678   227103

108   66  310   -4   478   56       232  394    26    51  3787   223316

103   63  310   -4   477   66       150  544    17   108  2360   220956

 97   62  310   -5   476   73       142  686    15   123  2185   218771

 91   61  310   -5   477   84        99  785    11   134  1488   217283

 87   51  350    2   475  102       180  965    19   152  3331   213952

 77   54  310   -1   480  112       216 1181    22   214  2986   210966

 65   62  310    2   483  116       407 1588    41   255  5554   205412

 51   66  310    4   486  109       364 1952    37   332  4930   200482

 37   72  350    5   478  106       336 2288    35   406  5096   195386

 21   90  350    4   478   99       336 2624    35   442  4331   191055

 11   89  310    1   482   62       185 2809    21   502  2580   188475

  6  101  310   -1   481   49        93 2902    11   513  1333   187142

  2  118  350    4   477   42       147 3049    17   530  2717   184425

TOD  121  310   -4   477   19        51 3100     6   536   727   183698

  0                        14           3192    16   552  1203   182495

TRIP FUEL 51129
COST11480.87


       *** END OF RUN ***
```

Fig. 17.

```
                    *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FUEL

                                                CH    1190581  0.00

NO. HEAD   FL  TMP  TAS  WIND    DIST  ACCD  TIME  ACCT  BURN  WEIGHT

                                                               232450

TOC   46  310             29            121    20    20  5797  226653

113   46  310   -2  479   39      41    162     5    25   694  225959

108   66  350    2  475   55     232    394    26    51  4622  221337

103   63  350    1  474   66     150    544    17   108  2407  218930

 97   62  350    1  474   74     142    686    16   123  2210  216720

 91   62  350    1  474   85      99    785    11   134  1496  215224

 87   51  350    2  475  102     180    965    19   153  2598  212626

 77   55  350    4  477  114     216   1181    22   215  3002  209624

 65   62  350    6  479  116     407   1588    41   256  5493  204131

 51   66  350    6  479  110     364   1952    37   333  4814  199317

 37   72  350    5  478  106     336   2288    35   407  4361  194956

 21   90  350    4  478   99     336   2624    35   443  4317  190639

 11   89  350    5  478   65     185   2809    21   503  2491  188148

  6  101  390    3  476   47      93   2902    11   514  1991  186157

  2  118  390    4  476   41     147   3049    17   531  2023  184134

TOD  121  390    4  477   22      26   3075     3   534   361  183773

  0                       17           3192    19   553  1271  182502

TRIP FUEL 49948
COST11319.77
```

Fig. 18.

```
                  *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FLTIME

                                               CH    1190581   0.00

NO. HEAD   FL  TMP  TAS  WIND     DIST  ACCD  TIME  ACCT  BURN  WEIGHT

                                                               233643
TOC   46  310              29        -   122    20    20  5844 227799
113   46  310   -2  479    39       40   162     5    25   678 227121
108   66  310   -4  478    56      232   394    26    51  3787 223334
103   63  310   -4  477    66      150   544    17   108  2360 220974
 97   62  310   -5  476    73      142   686    15   123  2185 218789
 91   61  310   -5  477    84       99   785    11   134  1488 217301
 87   51  350    2  475   102      180   965    19   152  3331 213970
 77   54  310   -1  480   112      216  1181    22   214  2986 210984
 65   62  310    2  483   116      407  1588    41   255  5545 205439
 51   66  310    4  486   109      364  1952    37   332  4930 200509
 37   72  350    5  478   106      336  2288    35   406  5096 195413
 21   90  350    4  478    99      336  2624    35   442  4331 191082
 11   89  310    1  482    62      185  2809    21   502  2580 188502
  6  101  310   -1  481    49       93  2902    11   513  1333 187169
  2  118  350    4  477    42      147  3049    17   530  2714 184455
TOD  121  310   -4  477    19       51  3100     6   536   727 183728
  0                        14            3192    16   552  1203 182525

TRIP FUEL 51118
COST11479.22

     *** END OF RUN ***
```

Fig. 19.

```
                        *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FUEL

                                              CH    2190581  0.00

NO. HEAD  FL  TMP  TAS  WIND    DIST  ACCD  TIME  ACCT  BURN  WEIGHT

                                                              252998

TOC  262  310            3            136    24    24  6678  246320

  1  262  310   -4  477    3       0   136     0    24     0  246320

  4  282  310   -1  480  -22     242   378    32    56  5105  241215

  8  279  310    2  484  -36     115   493    16   111  2490  238725

 17  292  310    3  485  -73      57   550     8   120  1320  237405

 30  268  310    6  487  -39     397   947    54   214  8306  229099

 44  268  350    7  480  -57     394  1341    56   310  9187  219912

 60  270  350    7  480  -66     394  1735    57   407  8142  211770

 75  264  350    5  478  -74     394  2129    60   507  7979  203791

 87  260  350    2  475  -78     179  2308    28   534  3549  200242

 92  257  350    1  474  -74     161  2469    24   559  3063  197179

 98  238  350    1  474  -78      66  2535    10   609  1250  195929

104  239  350    1  474  -66     217  2752    32   641  3926  192003

109  246  350    2  475  -48     189  2941    27   707  3197  188806

113  219  390    2  474  -28     180  3121    24   731  3590  185216

TOD  236  390    0  473  -27      55  3176     7   739   875  184341

115                      -20            3283    19   758  1271  183070

TRIP FUEL 69928
COST15666.98
                026704
```

```
                        *** DETAILED FLIGHT PLAN ***
```

Fig. 20.

```
                    *** DETAILED FLIGHT PLAN ***

MIN TRACK IN SPACE
FUEL

                                                  CH    2190581  0.00

NO. HEAD   FL  TMP   TAS  WIND    DIST  ACCD  TIME  ACCT  BURN  WEIGHT

                                                                253489

TOC  262  310                3             126    24    24  6704  246785

  1  262  310   -4   477     3      10     136     1    25   200  246585

  4  282  310   -1   480   -22     242     378    32    57  5105  241480

  8  279  310    2   484   -36     115     493    16   113  2490  238990

 17  292  310    3   485   -73      57     550     8   121  1323  237667

 30  268  310    6   487   -39     397     947    54   215  8322  229345

 44  268  350    7   480   -57     394    1341    56   311  9203  220142

 60  270  350    7   480   -66     394    1735    57   409  8142  212000

 75  264  350    5   478   -74     394    2129    60   508  7993  204007

 87  260  350    2   475   -78     179    2308    28   536  3555  200452

 92  257  350    1   474   -74     161    2469    24   560  3068  197384

 98  238  350    1   474   -78      66    2535    10   610  1250  196134

104  239  350    1   474   -66     217    2752    32   642  3926  192208

109  246  350    2   475   -48     189    2941    27   709  3202  189006

113  219  390    2   474   -28     180    3121    24   733  3594  185412

TOD  236  390    0   473   -27      67    3188     9   742  1056  184356

115                        -20            3283    19   801  1271  183085

TRIP FUEL 70404
COST15769.97
```

Program Listing (DC10).

```
Δ ELT DC10,1,810528,065609                                    ΔK*>WKEDAEAGA Δ
   COMMENT KNMI/KLM 020169 SPG NAV 20 JNG,
   NAVIGATION-FLIGHTPLANNING-MODULE FOR THE PRODUCTION OF
   FLIGHTPLANS OVER THE NORTH ATLANTIC.
   THE MAIN FEATURES
 ARE:
   1. INCLUSION OF 6 METEOROLOGICAL PARAMETER FIELDS,
      TEMPERATURE AND WIND DATA FOR 300,250 AND 200MB
   VALID FOR TWO STANDARD TIMES 12 HOURS APART.
   2.  WEATHER INFO. COMES FROM NATIONAL WEATHER SERVICE.
   3. INCLUSION OF A DYNAMIC PROCESS BASED ON INTERPOLATION OF
   TWO PARAMETER FIELDS.
   4. OPTIMALISATION OF FUEL, COSTS OR FLIGHT TIME IN SPACE AS
   WELL AS IN THE HORIZONTAL AS IN THE VERTICAL.
   5. IN - AND OUTBOUND TRAFFIC (IO= 1 OUTBOUND IO= -1 INBOUND).
   6. FLIGHTPLAN PRODUCTION FOR OPTIMUM TRACKS, SPECIFIC ROUTES
   ETC. E.G. ALTERNATIVE ATC MINIMUM ROUTES.
   7. BLOCKING E.G. SECTOR BLOCKING OVER THE NORTH ATLANTIC
   BLOCKING IN AIRWAYS AND ATC RESTRICTED AREAS.
   8. STANDARD CRUISE AND USE OF PERFORMANCE TABLES$

      CONSTANT KQ= 14 , W= 3$
      INTEGER     P1,P2,P3,Q1,Q2,Q3,S,TTT,
      G1,G2,G3,G4,B1,B2,B3,MQ,G1Q,G2Q,Q,
      RES1,RES2,RES3,I,J,N1,N2,N3,M,G,
      M0,M1,M2,M3,M4,M5,M6,K,IO,
      A1,A2,A3,A4,A5,T1,T2,T3,KOUNT,
      K1,K2,L1,L2,K1Q,K2Q,L1Q,L2Q,KK1,KK2,LL1,LL2,
      FLUR,TAXI,TOW,GRW,RESERVE,MAXTOW,MAXLW,
      DATE,ST,ST1,ROUTE,DISTANCE,FLTIME,BURN,
      GRWQ,TOWQ,LW,COST,LE,TEL$
      REAL    C0,CG,P,C1,X1,X2,Y1,Y2,X,Y,E1,E2,E3,
      GG1,GG2,D,ANGLE,LALAT,LALONG,TT,LA1,LO1,LA2,LO2,
      XX1,XX2,YY1,YY2,AID1,AID2,AID3,AID4,XSTER,YSTER,
      MINTIME,TC,TASM,WX,WY,TAS,ENDU,TEM$
      INTEGER ARRAY A(-1..KQ+1), V(0..41),
      TEMPTIME,TEMPDIST,TEMPFUEL(1..99, 1..9), MW(1..3, 1..7),
      RANGE(1..3, 1..33)$
      ARRAY DD(0..KQ)$
      ARRAY WCX,WCY,TPC(1..120, 1..34)$
      BOOLEAN ED$
      BOOLEAN ARRAY BK(0..2,1..6,1..6), BKQ(9..KQ,1..6,1..6)$
      FORMAT FO1(E2, S10, D7.0, A5),
             PEJT (E2, ' ', A1),
             FOT(X5, *I6, A2),
             FOD(*I8, A1),
             FO2(X5, *I7, A2)$
      LOCAL LABEL ERR,EOP$


      COMMENT PROCEDURE 'WDATA' SCANS  WEATHER DATA AND OBTAINS
              THE REQUIRED WEATHER INFO. FOR NORTH AMERICA AND
              NORTH ATLANTIC AREA $
```

```
PROCEDURE WDATA$
BEGIN INTEGER I,X,Y1,Y2,Y3,R,T,U,V$
STRING ARRAY WD(7..1..120,1..34)$
STRING S(81)$
STRING TW(AG(2), SP(3), TP(2))$
  STRING BA(BD(1),BL(3),AR(2),TM(1),B(3)),
         GW(UG(2),W1(7),WWW(W2(7),W3(7),W4(7)),HH(2),G(4)),
         SW (WT(2),DR(2),WS(3))$
LOCAL LABEL FINI$
FORMAT F(A,S80)$
FORMAT F1(*S8, A1),
       F2('BLOCKETTE #=',S3,'   AREA = NORTH AMERICA' , A1),
       F4('BLOCKETTE #=',S3,'   AREA = NORTH ATLANTIC',  A1),
       F5(3(X3,S7), A1),
       FW(*I6,A1),
       PJ(E2,' ',A1)$


PROCEDURE CHOP$
BEGIN INTEGER K,L$
IF I LEQ 45 THEN
   BEGIN GW(1,36)=S(I,36)$  I=I+36$
         IF I GTR 80 THEN
            BEGIN READ(PCF('NWS'),F,S,FINI)$ I=1$ END$
   END
ELSE BEGIN K=81-I$ GW(1,K)=S(I,K)$ READ(PCF('NWS'),F,S,FINI)$
          L=36-K$ GW(K+1, L)=S(1,L)$ I=L+1$
     END$
IF S(I) EQL '\' THEN
   BEGIN READ(PCF('NWS'),F,S,FINI)$  I=1$
   END$
END$ COMMENT CHOP$



PROCEDURE STOR$
BEGIN INTEGER J$
LOCAL LABEL L3$
STRING ARRAY SA(36..1..8)$
FOR J=1 STEP 1 UNTIL 8 DO
   BEGIN CHOP$ SA(1,36..J)=GW$ END$
FOR J=1 STEP 1 UNTIL 4 DO
   BEGIN GW=SA(1,36..J)$
         IF ALPHABETIC(WWW)  THEN
            BEGIN WRITE('W-ERROR')$  WRITE(WWW)$  GOTO L3$
            END$
         WD(1,7..Y1,X)=W2$
         WD(1,7..Y2,X)=W3$
         WD(1,7..Y3,X)=W4$
         Y1=Y1-1$ Y2=Y2-1$ Y3=Y3-1$
   END$
X=X-1$ Y1=Y1+4$ Y2=Y2+4$ Y3=Y3+4$
FOR J=5 STEP 1 UNTIL 8 DO
   BEGIN GW=SA(1,36..J)$
```

```
              IF ALPHABETIC(WWW) THEN
                 BEGIN WRITE('W-ERROR')$  WRITE(WWW)$  GOTO L3$
                 END$
              WD(1,7..Y1,X)=W2$
              WD(1,7..Y2,X)=W3$
              WD(1,7..Y3,X)=W4$
              Y1=Y1-1$ Y2=Y2-1$ Y3=Y3-1$
        END$
L3.. END$ COMMENT STOR$


     PROCEDURE AREA$
     BEGIN INTEGER K,L$
     IF I LEQ 71 THEN
        BEGIN BA(1,10)=S(I,10)$ I=I+10$
              IF I GTR 80 THEN
               BEGIN READ(PCF('NWS'),F,S,FINI)$ I=1$ END$
        END
     ELSE BEGIN K=81-I$ BA(1,K)=S(I,K)$
               READ(PCF('NWS'),F,S,FINI)$ L=10-K$
               BA(K+1,L)=S(1,L)$ I=L+1$
          END$
     IF S(I) EQL '\' THEN
        BEGIN READ(PCF('NWS'),F,S,FINI)$  I=1$
        END$
     END$ COMMENT AREA$


     PROCEDURE FINDNEXT$
     BEGIN
     IF S(I) EQL ' ' THEN
        FOR I=I WHILE S(I) NEQ '*' DO
           BEGIN READ(PCF('NWS'), F,S,FINI)$  I=1$
           END$
     END$


     PROCEDURE GETDATA$
     BEGIN INTEGER GMT,A,B,C,D,E,F,J,S,T$
     LOCAL LABEL  L5, NAM, NAT$
     A=20$ B=40$ C=60$ D=80$ E=100$ F=120$
     AREA$
     IF AR EQL '21' THEN
        BEGIN IF TM EQL '1' THEN
          BEGIN GMT=0$ X=14$ Y1=A$ Y2=B$ Y3=C$
                WRITE('GMT = 0000')$
          END
         ELSE IF TM EQL '3' THEN
          BEGIN GMT=12$ X=14$ Y1= D$ Y2=E$ Y3=F$
                WRITE('GMT = 1200')$
          END
          ELSE BEGIN WRITE('W-ERROR')$  GOTO L5$  END$
        GOTO NAM$
        END
     ELSE IF AR EQL '22' THEN
        BEGIN IF TM EQL '1' THEN
          BEGIN GMT=0$ X=34$ Y1=A$ Y2=B$ Y3=C$
                WRITE('GMT = 0000')$
          END
           ELSE IF TM EQL '3' THEN
           BEGIN GMT=12$ X=34$ Y1=D$ Y2=E$ Y3=F$
                WRITE('GMT = 1200')$
           END
           ELSE BEGIN WRITE('W-ERROR')$  GOTO L5$  END$
          GOTO NAT$
        END$
     WRITE('W-ERROR')$
     GOTO L5$
NAM.. FOR S=1 STEP 1 UNTIL 7 DO
        BEGIN FOR T=1 STEP 1 UNTIL 5 DO
              BEGIN
              WRITE(F2,BL)$
               STOR$  X=X+1$  AREA$
               IF NUMERIC(BL)  THEN
                 IF INTEGER(BL) GEQ 259 THEN
                   FOR J=1 STEP 1 UNTIL 3 DO CHOP
                  ELSE J=1
                ELSE BEGIN WRITE('W-ERROR')$  GOTO L5$
                     END$
              END$
              X=X-2$
              IF GMT EQL 0 THEN
               BEGIN Y1=A$ Y2=B$ Y3=C$ END
              ELSE BEGIN Y1=D$ Y2=E$ Y3=F$ END$
              IF S LSS 7  THEN AREA$
        END$
      GOTO L5$
NAT.. FOR S=1 STEP 1 UNTIL 10 DO
        BEGIN FOR T=1 STEP 1 UNTIL 5 DO
              BEGIN
              WRITE(F4,BL)$
               STOR$  X=X+1$  AREA$
               IF NUMERIC(BL)  THEN
                 IF INTEGER(BL) GEQ 253 THEN
                   FOR J=1 STEP 1 UNTIL 3 DO CHOP
                  ELSE J=1
                ELSE BEGIN WRITE('W-ERROR')$  GOTO L5$
                     END$
              END$
              X=X-2$
              IF GMT EQL 0 THEN
               BEGIN Y1=A$ Y2=B$ Y3=C$ END
              ELSE BEGIN Y1=D$ Y2=E$ Y3=F$ END$
              IF S LSS 10 THEN AREA$
        END$
L5.. END$ COMMENT GETDATA$
```

```
     PROCEDURE SCAN$
     BEGIN
     LOCAL LABEL BEGN,LO$
BEGN.. READ(PCF('NWS'),F,S,FINI)$
     I=1$
LO.. IF S(I,2) EQL '**' THEN
        BEGIN I=I+38$  GETDATA$  FINDNEXT$  GOTO LO$
        END
      ELSE GOTO BEGN$
     END$  COMMENT SCAN$

     COMMENT COMPUTATION OF X,Y COMPONENTS OF WIND VECTOR$

     PROCEDURE WINDCOMP(X,Y)$
     VALUE X,Y$
     INTEGER X,Y$
     BEGIN REAL A$
           IF X GTR 36 THEN WRITE('WIND ANGLE ERROR')$
        IF X LEQ 9 THEN
           BEGIN A=(90-X*10)*CG$
                WX=-Y*COS(A)$
                WY=-Y*SIN(A)$
           END
           ELSE IF X LEQ 18 THEN
           BEGIN A=(X*10-90)*CG$
                WX=-Y*COS(A)$
                WY=Y*SIN(A)$
           END
           ELSE IF X LEQ 27 THEN
           BEGIN A=(270-X*10)*CG$
                WX=Y*COS(A)$
                WY=Y*SIN(A)$
           END
           ELSE BEGIN A=(X*10-270)*CG$
                     WX=Y*COS(A)$
                     WY=-Y*SIN(A)$
                END$
      END$ COMMENT WINDCOMP$



      COMMENT LINEAR INTERPOLATION$

      INTEGER PROCEDURE INTP(F,F1,F2,F1Q,F2Q)$
      INTEGER F,F1,F2,F1Q,F2Q$
      INTP=((F-F1)*F2Q + (F2-F)*F1Q) /(F2-F1)$
      COMMENT INTP$


      SCAN$
 FINI..
      FOR R=0 STEP 1 UNTIL 5 DO
         BEGIN T=R*20$  WRITE(PJ)$
              WRITE('* WEATHER DATA IN 3 STANDARD PRESSURE LEVELS *')$
         WRITE('* 34X16 DATA OUT OF 34X20 GRIDPOINTS *')$
         WRITE('* TEMPERATURE, DIRECTION, SPEED *')$
         WRITE(' ')$  WRITE(' ')$
         FOR U=1 STEP 1 UNTIL 34 DO
             WRITE( F1, 20, FOR V=T+20 STEP -1
                    UNTIL T+5 DO WD(1,7..V,U))$
   END$

COMMENT CONVERSION OF WEATHER INFO. IN STANDARD PRESSURE LEVELS
        BY INTERPOLATION SCHEME TO WEATHER INFO. IN FLIGHT LEVELS$

FOR I=1 STEP 1 UNTIL 20, 61 STEP 1 UNTIL 80 DO
   FOR J=1 STEP 1 UNTIL 34 DO
BEGIN
   SW=WD(1,7..I,J)$
   M1=INTEGER(DR)$
   M2=INTEGER(WS)$
   T1=INTEGER(WT)$
   IF M1 GEQ 50 THEN BEGIN M1=M1-50$  T1=-T1$ END$
   WINDCOMP(M1,M2)$
   M1=WX$  M2=WY$
   SW=WD(1,7..I+20,J)$
   M3=INTEGER(DR)$
   M4=INTEGER(WS)$
   T2=INTEGER(WT)$
   IF M3 GEQ 50 THEN BEGIN M3=M3-50$  T2=-T2$  END$
   WINDCOMP(M3,M4)$
   M3=WX$  M4=WY$
   SW=WD(1,7..I+40,J)$
   M5=INTEGER(DR)$  M6=INTEGER(WS)$
   T3=INTEGER(WT)$
   IF M5 GEQ 50 THEN BEGIN M5=M5-50$  T3=-T3$  END$
   WINDCOMP(M5,M6)$
   M5=WX$  M6=WY$
   WCX(I,J)=INTP(P1,Q1,Q2,M1,M3)$
   WCY(I,J)=INTP(P1,Q1,Q2,M2,M4)$
   TPC(I,J)=INTP(P1,Q1,Q2,T1,T2)$
   WCX(I+20,J)=INTP(P2,Q2,Q3,M3,M5)$
   WCY(I+20,J)=INTP(P2,Q2,Q3,M4,M6)$
   TPC(I+20,J)=INTP(P2,Q2,Q3,T2,T3)$
   WCX(I+40,J)=INTP(P3,Q2,Q3,M3,M5)$
   WCY(I+40,J)=INTP(P3,Q2,Q3,M4,M6)$
   TPC(I+40,J)=INTP(P3,Q2,Q3,T2,T3)$
 END$

 FOR A1=0 STEP 1 UNTIL 5 DO
    BEGIN A2= A1*20$  WRITE(PEJT)$
    WRITE('*INTERPOLATED GRIDPOINT VALUES IN FLIGHT LEVELS*')$
           WRITE('*X-COMPONENTS OF WIND DATA*')$
           WRITE(' ')$  WRITE(' ')$
           FOR A4=1 STEP 1 UNTIL 34 DO
            WRITE(FW, 20, FOR A5=A2 + 20 STEP -1
              UNTIL A2+1 DO WCX(A5, A4))$
    END$
```

```

END$  COMMENT END OF WEATHER DATA COLLECTION$

P1=31000$  P2=35000$  P3=39000$
Q1=30065$  Q2=33999$  Q3=38662$
TEL=0$
CG=0.0174532925$
TASM=0.82*38.9826$

WRITE('* SCANNING WEATHER DATA IN NORTH AMERICA AND-')$
WRITE('                      NORTH ATLANTIC AREA *')$


WDATA$



COMMENT RELATIONSHIP BETWEEN GRAPH POINT NUMBERS AND ZONE INDICES$
A(-1)= -1$ A(0)= 0$ A(1)= 2$ A(2)= 6$ A(3)= 11$ A(4)= 26$
A(5)= 41$ A(6)= 56$ A(7)= 71$ A(8)= 86$ A(9)= 90$ A(10)= 96$
A(11)= 102$ A(12)= 107$ A(13)= 111$ A(14)= 114$ A(15)= 115$


COMMENT COORDINATES OF CHECK POINTS$

V(0)= 52290477$
V(1)= 51850115$ V(2) = 53640150$
V(3)= 51000200$ V(4) = 51990539$ V(5) = 53580300$
V(6)= 55000200$ V(7) = 49980632$ V(8) = 51840849$
V(9)= 52700892$ V(10)= 53500630$ V(11)= 55490459$
V(12)= 54271005$ V(13)= 48905454$ V(14)= 51375560$
V(15)= 53735697$ V(16)= 58476263$ V(17)= 47005800$
V(18)= 48545856$ V(19)= 55506000$ V(20)= 57006000$
V(21)= 53286035$ V(22)= 54836683$ V(23)= 46156006$
V(24)= 48006000$ V(25)= 49846439$ V(26)= 50506500$
V(27)= 51006750$ V(28)= 52006900$ V(29)= 45006300$
V(30)= 46176459$ V(31)= 47506600$ V(32)= 48576826$
V(33)= 50007050$ V(34)= 43836608$ V(35)= 44846867$
V(36)= 45327179$ V(37)= 454673855$ V(38)= 41287003$
V(39)= 423670995 V(40)= 43007250$ V(41)= 40647378$

COMMENT DETERMINATION OF CONNECTIVITY AMONG GRAPH POINTS.
        FALSE=CONNECTED, TRUE=BLOCKED $
FOR I= 1 STEP 1 UNTIL 6 DO
FOR J= 1 STEP 1 UNTIL 6 DO
BEGIN
   FOR MQ= 0,1,2 DO BK(MQ,I,J) = TRUE$
   FOR MQ= 9 STEP 1 UNTIL  KQ DO BKQ(MQ,I,J) = TRUE
END$
BK(1,1,1)= BK(2,1,1)= BK(1,2,2)= BK(2,2,2)=
BKQ(9,1,1)= BKQ(10,1,1)= BKQ(11,1,1)= BKQ(12,1,1)=
BKQ(13,1,1)= BKQ(14,1,1)= BKQ(9,2,2)= BKQ(10,2,2)=
BKQ(11,2,2)= BKQ(12,2,2)= BKQ(13,2,2)= BKQ(10,3,3)=
BKQ(11,3,3)= BKQ(13,3,3)= BKQ(11,4,4)= BK(2,3,4)= BK(2,4,5)=
    BKQ(12,4,4) = BKQ(10,6,6) = BK(1,1,2) =
    BK(1,2,3) = BK(1,2,4) = BK(2,2,3) =
    BKQ(9,1,2)= BKQ(9,2,3)= FALSE$
    BKQ(9,2,5)= BKQ(9,3,4)= BKQ(9,3,5)= BKQ(9,4,6)= BKQ(10,2,1)=
    BKQ(10,4,3)= BKQ(10,5,3)= BKQ(10,5,4)= BKQ(10,6,5)=
    BKQ(11,1,2)= BKQ(11,3,4)= BKQ(11,5,4)= FALSE$
    BKQ(11,6,5)= BKQ(12,2,1)= BKQ(12,3,2)= BKQ(12,4,3)=
    BKQ(12,5,4)= BKQ(13,1,2)= BKQ(14,2,1)= BKQ(14,3,1)= BK(0,1,1)=
    BK(0,1,2) =FALSE$


    BEGIN
      REAL ARRAY AB(0..10), LENGTH(1..10),
      HEADING(1..W),
      TIME,AIRDIST,TDEV,WIX,WIY,TAX,TAY(1..W)$
      REAL COSA,COSD$
      LOCAL LABEL DONE$

      INTEGER PROCEDURE ZONEI(Q)$  INTEGER Q$
      COMMENT THIS SUBROUTINE DETERMINES THE INDEX OF THE
      ZONE ASSOCIATED WITH A STATIONSNUMBER$
      BEGIN
        LOCAL LABEL  AGAIN$ INTEGER I$
        I=  -1$
AGAIN..
        I= I+1$ IF A(I)-Q LSS 0 THEN GOTO AGAIN$
        ZONEI= I
      END$  COMMENT ZONEI$

      PROCEDURE DT(K)$ INTEGER K$
      COMMENT THIS SUBROUTINE DISSECTS LATITUDE AND
      LONGITUDE FROM THE COMPRESSED COORDINATES$
      BEGIN
        INTEGER K0$
        K0= ENTIER(V(K)/10000)$
        LALAT= K0/100$
        LALONG= (V(K)-10000*K0)/100
      END$  COMMENT DISSECT$

      PROCEDURE LIS(U)$ INTEGER U$
      COMMENT DETERMINATION OF LATITUDE AND LONGITUDE FOR
           GRAPH POINTS$
      COMMENT THE SIGN OF LONGITUDE IS CHANGED FOR
           EAST OF GREENWHICH$
      BEGIN
        IF U EQL 19 THEN DT(12)
        ELSE IF U LEQ 11 THEN
        BEGIN
          DT(U)$ IF U EQL 0 OR U EQL 1 OR
          U EQL 2 THEN LALONG= -LALONG
        END
        ELSE IF U LEQ 26 THEN
        BEGIN
          LALAT= 35+U$ LALONG= 10
```

```
     END
     ELSE IF U LEQ 41 THEN
     BEGIN
        LALAT= 20+U$ LALONG= 20
     END
     ELSE IF U LEQ 56 THEN
     BEGIN
        LALAT= 5+U$ LALONG= 30
     END
     ELSE IF U LEQ 71 THEN
     BEGIN
        LALAT= U-11$ LALONG= 40
     END
     ELSE IF U LEQ 86 THEN
     BEGIN
        LALAT= U-26$ LALONG= 50
     END
     ELSE DT(U-74)
  END$  COMMENT LIS$

  PROCEDURE CTQ$
  COMMENT THIS SUBROUTINE PREPARES THE TIME INSTANTS AT WHICH
  IN EACH ZONE THE METEOROLOGICAL PARAMATER WILL BE
  DERIVED FROM BOTH PARAMETER FIELDS, COMPOSITE CHARTS
  IN TIME ARE SIMULATED$
  BEGIN
     INTEGER J0,M0$
     REAL A,SUM,RES$
     A= 0.4$
     J0 = IF IO EQL 1 THEN KQ ELSE 0$  SUM= 0$
     FOR I= KQ-J0 STEP IO UNTIL J0 DO
     BEGIN
        SUM= SUM+(IF I GTR 3 AND I LSS 8 THEN 2*A ELSE A)$
        DD(I)= SUM
     END$
     M0 = ZONEI(ST)$  RES= IF IO EQL 1 THEN DD(M0) ELSE
     DD(M0-1)$
     FOR I= 0 STEP 1 UNTIL KQ DO DD(I)=  IF TT GTR 12
     THEN TT-12 ELSE TT+A/2+DD(I)-RES$
  END$  COMMENT CTQ$

COMMENT COMPUTATION OF GRID POINT VALUES FOR DYNAMIC PROCESS$

   REAL PROCEDURE HH(V,W,F,SELECT)$
   VALUE F,V,W,SELECT$
   INTEGER F,V,W,SELECT$
   BEGIN
      M5= (F-1)*20$
      M6= (F+2)*20$
      IF SELECT EQL 1 THEN
        BEGIN M1=WCX(V+M5,W)$  M2=WCX(V+M6,W)$
              HH=((12-DD(M))*M1 + DD(M)*M2) /12$ END
      ELSE IF SELECT EQL 2 THEN
        BEGIN M3=WCY(V+M5,W)$  M4=WCY(V+M6,W)$
```

```
           HH=((12-DD(M))*M3 + DD(M)*M4) /12$ END
      ELSE  BEGIN T1=TPC(V+M5, W)$  T2=TPC(V+M6,W)$
             HH=((12-DD(M))*T1 + DD(M)*T2) /12$ END$
    END$  COMMENT HH$

  COMMENT COMPUTATION OF GRID POINT VALUE BY BI-LINEAR INTERPOLATION
       FOR AN ARBITRARY POINT (TEMP., DIRECTION AND SPEED)$

    REAL PROCEDURE GEOP(V,W,C,SELECT)$
    VALUE C,SELECT,V,W$
     INTEGER C,SELECT$  REAL V,W$
    BEGIN
      INTEGER V1,W1$ REAL A,B,B1,B2,B3,B4$
      V1= ENTIER(V)$ W1= ENTIER(W)$
      B1=IF SELECT EQL 1 THEN HH(V1,W1,C,1)
        ELSE IF SELECT EQL 2 THEN HH(V1,W1,C,2)
          ELSE HH(V1,W1,C,3)$
      B2=IF SELECT EQL 1 THEN HH(V1+1,W1,C,1)
        ELSE IF SELECT EQL 2 THEN HH(V1+1,W1,C,2)
          ELSE HH(V1+1,W1,C,3)$
      B3=IF SELECT EQL 1 THEN HH(V1+1,W1+1,C,1)
        ELSE IF SELECT EQL 2 THEN HH(V1+1,W1+1,C,2)
          ELSE HH(V1+1,W1+1,C,3)$
      B4=IF SELECT EQL 1 THEN HH(V1,W1+1,C,1)
        ELSE IF SELECT EQL 2 THEN HH(V1,W1+1,C,2)
          ELSE HH(V1,W1+1,C,3)$
      A= V1+1-V$ B= W1+1-W$
      GEOP= A*B*B1+(1-A)*B*B2+(1-A)*(1-B)*B3+A*(1-B)*B4$
    END$  COMMENT GEOP$

  COMMENT COMPUTATION OF GREAT CIRCLE DISTANCE BETWEEN TWO
       POINTS BY USING THE GONIOMETRIC RELATIONS$

    REAL PROCEDURE GEODIST(LAT1,LON1,LAT2,LON2)$
    VALUE LAT1,LAT2,LON1,LON2$
    REAL  LAT1,LAT2,LON1,LON2$
    BEGIN
    LOCAL LABEL SKP$
    IF LAT1 EQL LAT2 AND LON1 EQL LON2 THEN
      BEGIN GEODIST=0$  GOTO SKP$ END$
    GEODIST=60*ARCCOS(SIN(LAT1*CG)*SIN(LAT2*CG)
        +COS(LAT1*CG)*COS(LAT2*CG)*COS((LON2-LON1)*CG))/CG$
SKP..
    END$  COMMENT GEODIST$

  COMMENT COMPUTATION OF COORDINATES OF TWO GEOGRAPHICAL POINTS
       ON THE  GRID PLANE$

    PROCEDURE GEOMGRID(PP,GG)$ INTEGER PP,GG$
    COMMENT X1,X2=ROW INDICES AND Y1,Y2=COL. INDICES $
    BEGIN
      LIS(PP)$ LA1= LALAT$ LO1= LALONG$
      LIS(GG)$ LA2= LALAT$ LO2= LALONG$
      X1=(70-LA1)/2.5$  Y1=(130-LO1)/5$
      X2=(70-LA2)/2.5$  Y2=(130-LO2)/5$
    END$  COMMENT GEOMGRIDS$


COMMENT COMPUTATION OF GREAT CIRCLE DISTANCE AND TRUE COURSE
         OF A SEGMENT FLOWN$
    PROCEDURE PART2GEOM$
    BEGIN
    REAL A$
    INTEGER N$
        REAL TOTL,LA11,LA22,LO11,LO22,XX1,XX2,YY1,YY2$
        K1= ENTIER(X1)$ K2= K1+1$ L1= ENTIER(Y1)$
        L2= L1+1$ K1Q= ENTIER(X2)$ K2Q= K1Q+1$
        L1Q= ENTIER(Y2)$ L2Q= L1Q+1$
        N1= ABS(K1Q-K1)$
        N2= ABS(L1Q-L1)$
        N3 = IF K1Q EQL K1 AND L1Q EQL L1 THEN 0
        ELSE IF N1 LEQ N2 THEN N2 ELSE N1$
        BEGIN
            INTEGER I,J$
            LOCAL LABEL LAB1$
            AB(0)= 0$ AB(N3+1)= 1$
            IF N3 EQL 0 THEN GOTO LAB1$ IF N3 EQL N1 THEN
            BEGIN
                IF K1 GTR K1Q THEN
                BEGIN
                    RES1= K1$
                    RES2= K2Q$ RES3= -1
                END
                ELSE
                BEGIN
                    RES1= K2$ RES2= K1Q$
                    RES3= 1
                END$
                FOR I= RES1 STEP RES3 UNTIL RES2 DO
                BEGIN
                    J= RES3*(I-RES1)+1$ AB(J)= (I-X1)/
                    (X2-X1)
                END
            END
            ELSE
            BEGIN
                IF L1 GTR L1Q THEN
                BEGIN
                    RES1= L1$
                    RES2= L2Q$ RES3= -1
                END
                ELSE
                BEGIN
                    RES1= L2$ RES2= L1Q$
                    RES3=  1
                END$
                FOR I= RES1 STEP RES3 UNTIL RES2 DO
                BEGIN
```

```
                      J= RES3*(I-RES1)+1$ AB(J)= (I-Y1)/
                      (Y2-Y1)
                   END
                END$
LAB1..
                TOTL=0$
                LA11=LA1$  LO11=LO1$
                N=N3+1$
                FOR I=1 STEP 1 UNTIL N DO
                BEGIN
                   XX2=AB(I)*X2 + (1-AB(I))*X1$
                   YY2=AB(I)*Y2 + (1-AB(I))*Y1$
                   LA22=70-(XX2*2.5)$
                   LO22=130-(YY2*5.0)$
                   LENGTH(I)=GEODIST(LA11,LO11,LA22,LO22)$
                   LA11=LA22$  LO11=LO22$
                   TOTL=LENGTH(I)+TOTL$
                END$
                D=TOTL$
                TC=ARCCOS((SIN(LA2*CG) - SIN(LA1*CG)*COS(D/60*CG)) /
                   (SIN(D/60*CG)*COS(LA1*CG)))$
                TC=TC/CG$
                IF  SIN((LO2-LO1)*CG) GTR 0 THEN TC=360 - TC$
                IF TC LEQ 90 THEN
                  BEGIN A=(90-TC)*CG$
                    E1=COS(A)$  E2=SIN(A)$  END
                ELSE IF TC LEQ 180 THEN
                  BEGIN A=(TC-90)*CG$
                    E1=COS(A)$  E2=-SIN(A)$  END
                ELSE IF TC LEQ 270 THEN
                  BEGIN A=(270-TC)*CG$
                    E1=-COS(A)$  E2=-SIN(A)$  END
                ELSE BEGIN A=(TC-270)*CG$
                    E1=-COS(A)$  E2=SIN(A)$  END$
             END
          END$  COMMENT PART2GEOM$

          REAL PROCEDURE SUM1(A, B, ARR)$
          VALUE A, B$
          INTEGER A, B$  ARRAY ARR$
          BEGIN  INTEGER I$  REAL S$
             S= 0$
             FOR I= A STEP 1 UNTIL B DO
                S= S + ARR(I)$
             SUM1 = S$
          END$  COMMENT SUM1$

          REAL PROCEDURE SUM2(A, B, ARR1, ARR2)$
          VALUE A, B$
          INTEGER A, B$  ARRAY ARR1, ARR2$
          BEGIN  INTEGER I$  REAL S$
             S= 0$
             FOR I= A STEP 1 UNTIL B DO
                S = S + ARR1(I)*ARR2(I)$
```

```
       SUM2 = S$
     END$  COMMENT  SUM2$

  COMMENT COMPUTATION OF A DRIFT ANGLE FOR THE SEGMENTS$

     REAL PROCEDURE DRIFT(X,Y)$
     VALUE X,Y$
     REAL  X,Y$
     BEGIN
     REAL SPD,ANG,DEG,W$
       LOCAL LABEL BR$
       IF X EQL 0 AND Y EQL 0 THEN
         BEGIN SPD=DEG=0$  GOTO BR$ END$
       SPD=SQRT(X**2 + Y**2)$
       W=ABS(X)$
       IF SPD EQL 0 THEN GOTO BR$
       ANG=ARCCOS(W/SPD) /CG$
       IF X GEQ 0 AND Y LEQ 0 THEN ANG=270+ANG
       ELSE IF X GEQ 0 AND Y GEQ 0 THEN
            ANG=270-ANG
       ELSE IF X LEQ 0 AND Y GEQ 0 THEN ANG=90+ANG
       ELSE IF X LEQ 0 AND Y LEQ 0 THEN ANG=90-ANG
       ELSE WRITE('WIND DIRECTION ERROR')$
       DEG=ANG-TC$
BR..
       DRIFT=ANG=SPD*SIN(DEG*CG) /TAS$
       COSA=COS(ARCSIN(ANG))$
     END$ COMMENT DRIFT$


     PROCEDURE METPROC(S)$ VALUE S$  INTEGER S$
     COMMENT PROCESSING METEOROLOGICAL PARAMETERS$
     BEGIN
       INTEGER I,N$
       REAL HD,GS,TASS$
       ARRAY TEMP,WNDX,WNDY,ENDUR(1..N3+1)$
       XX1=X1$  YY1=Y1$
       N=N3+1$
       FOR I=1 STEP 1 UNTIL N DO
       BEGIN
         XX2= AB(I)*X2+(1-AB(I))*X1$
         YY2= AB(I)*Y2+(1-AB(I))*Y1$
         XSTER= (XX1+XX2)/2$
         YSTER= (YY1+YY2)/2$
         TEMP(I)= -GEOP(XSTER,YSTER,S,3)$
         WNDX(I)=GEOP(XSTER,YSTER,S,1)$
         WNDY(I)=GEOP(XSTER,YSTER,S,2)$
         XX1=XX2$  YY1=YY2$
       END$
       WX=SUM1(1,N,WNDX) /N$
       WY=SUM1(1,N,WNDY) /N$
       TEM=SUM1(1,N,TEMP) /N$
       TAS=TASM*SQRT(273.16+TEM)$
       ANGLE=DRIFT(WX,WY)$
```

```
    FOR I=1 STEP 1 UNTIL N DO
       BEGIN
       TASS=TASM*SQRT(273.16 + TEMP(I))$
       GS=TASS*COSA + (WNDX(I)*E1 + WNDY(I)*E2)$
       ENDUR(I)=36000*LENGTH(I) / GS$
       END$
    ENDU=SUM1(1,N,ENDUR)$
    RES1= 31+(S-1)*4$
    WIX(S)=SUM2(1,N,ENDUR,WNDX)/ENDU$
    WIY(S)=SUM2(1,N,ENDUR,WNDY)/ENDU$
    TEM=SUM2(1,N,ENDUR,TEMP)/ENDU$
    TAS=TASM*SQRT(273.16+TEM)$
    TDEV(S)= IF RES1 LEQ 35.332 THEN TEM +1.98*RES1-15 ELSE
    55+TEM$
    TIME(S)= ENDU$
    AIRDIST(S)= ENDU *TAS/36000$ LE= D$
    HEADING(S)=TC+ARCSIN(ANGLE)/CG$
    HD=HEADING(S)$
    IF HD LEQ 90 THEN
      BEGIN HD=(90-HD)*CG$
         TAX(S)=TAS*COS(HD)$  TAY(S)=TAS*SIN(HD)$  END
    ELSE IF HD LEQ 180 THEN
      BEGIN HD=(HD-90)*CG$
         TAX(S)=TAS*COS(HD)$  TAY(S)=-TAS*SIN(HD)$  END
    ELSE IF HD LEQ 270 THEN
      BEGIN HD=(270-HD)*CG$
         TAX(S)=-TAS*COS(HD)$  TAY(S)=-TAS*SIN(HD)$  END
    ELSE BEGIN HD=(HD-270)*CG$
         TAX(S)=-TAS*COS(HD)$
         TAY(S)=TAS*SIN(HD)$  END$
  END$  COMMENT METPROC$

  COMMENT 'TABLE' AND 'CLIMBCORT' ARE TABLE LOOK-UP PROCEDURES$

  INTEGER PROCEDURE TABLE(P,R,MA)$
  VALUE P,R$  INTEGER P,R$  INTEGER ARRAY MA$
  BEGIN
  INTEGER G,B,D$  REAL R0,C$
  LOCAL LABEL LAB20$
  B=IF P EQL 0 THEN 4536 ELSE 5$
  C=IF P EQL 0 THEN GRWQ ELSE TDEV(R)$
  D=IF P EQL 0 THEN 127006 ELSE -15$
  R0=(C - D)/B$
  G=ENTIER(R0)$
  R0=R0 - G$
  IF MA(R,G+1) EQL -1 THEN
     BEGIN TABLE=-1$  GOTO LAB20$
     END$
  TABLE=R0*MA(R,G+1) + (1-R0)*MA(R,G)$
-AB20..
  END$  COMMENT TABLE$

  INTEGER PROCEDURE CLIMBCORT(L,MA)$
  VALUE L$  INTEGER L$  INTEGER ARRAY MA$
```

```
          BEGIN
          INTEGER R,C,E1,E2,E3,E4$
          REAL G,R0,R1$
          LOCAL LABEL LAB21$
          G=(GRWQ - 127006)/4536$
          R=ENTIER(G)$
          R0=G-R$
          R=R*3 + L$
          G=(TDEV(L) + 25)/5$
          C=ENTIER(G)$
          R1=G-C$
          E1=MA(R,C)$
          E2=MA(R+3,C)$
          E3=MA(R,C+1)$
          E4=MA(R+3,C+1)$
          IF E1 EQL -1 OR E2 EQL -1 OR E3 EQL -1 OR E4 EQL -1 THEN
             BEGIN CLIMBCORT=-1$ GOTO LAB21$ END$
          CLIMBCORT=R0*R1*E4 + (1-R0)*(1-R1)*E1
                   + (1-R0)*R1*E3 + R0*(1-R1)*E2$
-AB21..
          END$  COMMENT CLIMBCORT$

          PROCEDURE READQ(Y0,Y1,Y2,Y3,ARR)$
          VALUE Y0,Y1,Y2,Y3$  INTEGER Y0,Y1,Y2,Y3$
          INTEGER ARRAY ARR$
          BEGIN
             FOR I=Y0 STEP 1 UNTIL Y1 DO
                READ(PCF('DC10F'), FOR J=Y2 STEP 1 UNTIL Y3
                DO ARR(I,J), ERR,ERR)$
          END$  COMMENT READ$

          PROCEDURE WRITEQ(Y0,Y1,Y2,Y3,ARR)$
          VALUE Y0,Y1,Y2,Y3$
          INTEGER Y0,Y1,Y2,Y3$
          INTEGER ARRAY ARR$
          BEGIN
          FOR I=Y0 STEP 1 UNTIL Y1 DO
             WRITE(FOT,Y3,FOR J=Y2 STEP 1 UNTIL Y3 DO
                   ARR(I,J), ERR)$
          END$  COMMENT WRITEQ$

          READQ(1,3,1,16,RANGE)$
          READQ(1,3,17,33,RANGE)$
          READQ(1,99,1,9,TEMPTIME)$
          READQ(1,99,1,9,TEMPDIST)$
          READQ(1,99,1,9,TEMPFUEL)$
          READQ(1,3,1,7,MW)$

          WRITE(PEJT)$
          WRITE('* PERFORMANCE DATA FOR  DC10 *')$ WRITE(' ')$
          WRITE(' ')$  WRITE(' ')$
          WRITE('* SPECIFIC RANGE *')$  WRITE(' ')$
          WRITEQ(1,3,1,16,RANGE)$
          WRITEQ(1,3,17,33,RANGE)$
```

```
      WRITE(' ')$  WRITE(' ')$  WRITE(' ')$  WRITE(' ')$
      WRITE('* MAX. WEIGHT *')$  WRITE(' ')$
      WRITEQ(1,3,1,7,MW)$
      WRITE(PEJT)$
      WRITE('* CLIMB TIME *')$  WRITE(' ')$
      WRITEQ(1,99,1,9,TEMPTIME)$
      WRITE(PEJT)$
      WRITE('* CLIMB DISTANCE *')$  WRITE(' ')$
      WRITEQ(1,99,1,9,TEMPDIST)$
      WRITE(PEJT)$
      WRITE('* CLIMB FUEL *')$  WRITE(' ')$
      WRITEQ(1,99,1,9,TEMPFUEL)$



RUN..
      READ(PCF('DC10F'),  FLUR, TAXI, GRW, RESERVE, TOW,
      MAXTOW,MAXLW,DATE,IO,TTT,ROUTE,DONE, ERR)$
      WRITE(' ')$  WRITE(' ')$  WRITE(' ')$
      WRITE('     *** INPUT PARAMETERS ***')$
      WRITE(F02, 11, FLUR,TAXI,GRW,RESERVE,TOW,
            MAXTOW,MAXLW,DATE,IO,TTT,ROUTE)$
      IF ROUTE GEQ 1111 THEN
      BEGIN
        READ(PCF('DC10F'),  Q, DONE, ERR)$
        WRITE(F02, 1, Q)$
           Q=Q-1$
      END
      ELSE
      BEGIN
        READ(PCF('DC10F'),  ST,ST1, DONE, ERR)$
        WRITE(F02, 2, ST, ST1)$
         G1Q= ZONEI(ST)$ G2Q= ZONEI(ST1)$ Q= ABS(G2Q-G1Q)
      END$
      TT = TTT$

      BEGIN
         INTEGER ARRAY STOREI,STOREIQ,STOREG,STOREGQ,E(0..Q),
         F(-1..Q)$
         FORMAT F04(X63, I6, A2),
                F05('TRIP FUEL', I6, A1),
                F06('COST', D8.2, A1)$
         REAL ARRAY DECLTIME,DECLFUEL,DECLDIST(1..W)$
         INTEGER PROCEDURE SQ(H1,H2)$ INTEGER H1,H2$
         SQ= A(G1Q+IO*H1-1)+H2$
         INTEGER PROCEDURE JQ(H1,H2)$ INTEGER H1,H2$
         JQ= F(H1-1)+H2-STOREI(H1)+1$

         PROCEDURE DESCLIMB(S)$ VALUE S$ INTEGER S$
         COMMENT COMPUTATION OF PERFORMANCE IN CLIMB OR DESCENT$
         BEGIN
            INTEGER ALT$
            LOCAL LABEL JUM$
            REAL TASQRT$
            TASQRT=SQRT(TAX(S)**2 + TAY(S)**2)$
           ALT= 31+(S-1)*4$
           DECLTIME(S)=IF MQ EQL 0 THEN 60*CLIMBCORT(S,TEMPTIME)
           /1000 ELSE 3.1+0.41*ALT$
           DECLDIST(S)=IF MQ EQL 0 THEN CLIMBCORT(S,TEMPDIST)
           ELSE 9+3*ALT$
           DECLFUEL(S)=IF MQ EQL 0 THEN CLIMBCORT(S,TEMPFUEL)
           ELSE 940+8.5*ALT$ IF DECLTIME(S) LSS 0 THEN GOTO JUM$

           AID4=DECLDIST(S)*60/(DECLTIME(S)*TASQRT)$
           DECLDIST(S)= SQRT((0.75*WIX(S)+TAX(S)*AID4)**2+
           (0.75*WIY(S)+TAY(S)*AID4)**2)*(DECLTIME(S)-2)/60$
           IF DECLDIST(S) GTR LE THEN DECLDIST(S)=LE$
           AIRDIST(S)= (1-DECLDIST(S)/LE)*AIRDIST(S)$
           TIME(S)=AIRDIST(S)/TASQRT*36000$
           LE=LE-DECLDIST(S)$
JUM..
        END$  COMMENT DECLIMB$


        PROCEDURE  LINE(N1,N2)$ BOOLEAN N1,N2$
        COMMENT PRINT-OUT PROCEDURE$
        BEGIN
           INTEGER I,J,K,L,M,N,P,Q,R,S,T    $
           FORMAT P1('TOC', 2I5, X12, I4, X10, I4, 3I6, I8,  A2),
                  P2(I3, X22, I4, X10, I4, 3I6, I8, A2),
                  P3('TOD', 4I5, I6, I8, I6, 3I6, I8, A2),
                  P4(I3, 4I5, I6, I8, 4I6, I8, A2)$
           LOCAL LABEL LAB30,LAB31,LAB32$
           IF N1 THEN GOTO LAB30$
           IF N2 THEN GOTO LAB31$
           I=G3$  S=MQ$  T=B3$
 LAB30..   GEOMGRID(G4, G3)$
           J=HEADING(G)$
           K= 310 + (G-1)*40$
           IF NOT N1 AND NOT N2 THEN BEGIN
           L= TDEV(G)$
           M= SQRT(TAX(G)**2 + TAY(G)**2)$
         END$
 LAB31..
           N= (IF N1 OR N2 THEN .75 ELSE 1)*
                  (WIX(G)*E1 + WIY(G)*E2)$
           M3= IF N1 OR N2 THEN
                  DECLTIME(G)*600 ELSE TIME(G)$
           P= 100*ENTIER(M3/36000) + MOD(M3, 36000)/600$
           Q= 100*ENTIER(FLTIME/36000) + MOD(FLTIME, 36000)/600$
           R= BURN - GRWQ$
         IF N1 THEN
           BEGIN
              WRITE(P1,J,K,N,DISTANCE,P,Q,R,GRWQ)$
              GOTO LAB32
           END$
         IF N2 THEN
           BEGIN
              WRITE(P2,ST1,N,DISTANCE,P,Q,R,GRWQ)$
           GOTO LAB32
         END$
         IF S EQL T THEN
           WRITE(P3,J,K,L,M,N,LE,DISTANCE,P,Q,R,GRWQ)
          ELSE    TE(P4,I,J,K,L,M,N,LE,DISTANCE,P,Q,R,GRWQ)$

LAB32..
       END$ COMMENT LINES$

       PROCEDURE SPACEOPT(FBQ,FB,BI,BG,RR)$ INTEGER FB,RR$
       BOOLEAN FBQ,BI,BG$
       COMMENT FBQ DETERMINES WETHER THE FLIGHTPLAN
       COMPUTATION WILL BE PERFORMED BACKWARDS(FALSE) OR FOR
       WARDS(TRUE).
       FB DETERMINES WETHER COSTS(+1) OR FUEL(0) OR FLIGHT
       TIME(-1) WILL BE OPTIMIZED. RR IS TAKE OFF WEIGHT OR
       LANDINGWEIGHT.
       BI DEFINES WETHER THE NAVIGATION REGIME IS FREE IN THE
       HORIZONTAL(FALSE) OR BOUNDED BY ONE POINT(TRUE).
       BG DEFINES WETHER THE CRUISING ALTITUDE IS FREE(FALSE)
       OR BOUNDED(TRUE)$
       BEGIN
         LOCAL LABEL ITER, ENDG, CI, REPEAT, EIND$
         INTEGER AA,QUANT,II,JJ,GG,GRWP, DDG, DTG, TMG$
         BOOLEAN ITERATION, BCLIMB,BNQ,BN$
         INTEGER ARRAY ROW,ROWQ,QUANTQ(0..F(Q))$

         PROCEDURE PREP(A1,B1,AA,BB)$ INTEGER A1,B1$
         COMMENT COMPUTATION OF DISTANCE AND FLIGHT TIME, AND
              PRINT-OUT OF THE FLIGHT PLAN$
         BOOLEAN AA,BB$
         COMMENT AA REFERS TO CLIMB, BB TO DESCENT$
         BEGIN
           DISTANCE= DISTANCE+A1$ FLTIME= FLTIME+B1$
           LINE(AA,BB)$ BURN= GRWQ
         END$  COMMENT PREP$

         PROCEDURE EDITING$
         BEGIN LOCAL LABEL ENDED$
           FORMAT FO3(X44, 'CH', I4, I6, D6.2, A2)$

           PROCEDURE QP$
           BEGIN
           IF FB EGL 1 THEN
             WRITE('COST')
           ELSE IF FB EQL 0 THEN
             WRITE('FUEL')
           ELSE IF FB EQL -1 THEN
             WRITE('FLTIME')$
           GOTO ENDED$
           END$  COMMENT QP$


           WRITE(PEJT)$
 WRITE('                *** DETAILED FLIGHT PLAN ***')$
```

```
                WRITE(' ')$  WRITE(' ')$
                IF ROUTE EQL 1111 THEN
                BEGIN
                   WRITE('NON SPECIAL MIN TRACK')$
                   QP
                END$
                IF ROUTE EQL 4444 THEN
                BEGIN
                   WRITE('ATC I')$   QP$
                END$
                IF ROUTE EQL 5555 THEN
                BEGIN
                   WRITE('ATC II')$   QP$
                END$
                IF ROUTE EQL 7777 THEN
                BEGIN
                   WRITE('GRC')$   QP$
                END$
                IF ROUTE EQL 9999 THEN
                BEGIN
                   WRITE('SPECIAL TRACK')$   QP$
                END$
                WRITE('MIN TRACK IN SPACE')$   QP$
ENDED..
                WRITE(F03, FLUR, DATE, TT)$
                WRITE(' ')$
WRITE(
'NO. HEAD   FL  TMP  TAS  WIND   DIST  ACCD  TIME  ACCT  BURN  WEIGHT'
    )$
             END$  COMMENT EDITING$

             PROCEDURE QQ(O)$ INTEGER O$
             COMMENT COMPUTATION OF SEGMENT CONTRIBUTION. COST,
                  TIME, FUEL, WEIGHT ETC.$
             BEGIN
                LOCAL LABEL JM1, JM2, JM3, JM, ENDQ$

                PROCEDURE CL(M1,M2)$ REAL M1,M2$
                COMMENT M1 DENOTES SEGMENT FUEL, M2 SEGMENT TIME$
                BEGIN
                   LOCAL LABEL JU$ REAL MM$
                   IF BCLIMB THEN
                   BEGIN
                      MM=  0$ GOTO JU
                   END$
                   IF ITERATION THEN
                   BEGIN
                      MM=  IF G GEQ GG THEN -28
                      ELSE -700$ GOTO JU
                   END$
                   MM= IF (O EQL -1 AND G GEQ GG) OR (O EQL 1 AND
                          G LSS GG) THEN 700 ELSE 28$
JU..
                   MM=  M1+ O*(GG-G)* (IF MQ NEQ B1 THEN MM ELSE 0)$
```

```
            GRWQ= GRWQ + 0*MM$
            GUANT= QUANT+(IF FB EQL 1 THEN 1083.3*M2+15*MM
           ELSE IF FB EQL 0 THEN MM ELSE 600*M2)$
        END$  COMMENT CL$

        QUANT = QUANTQ(I)$
        ITERATION= IF NOT FBQ AND MQ EQL 0 AND 0 EQL -1
           THEN TRUE
        ELSE FALSE$ IF ROW(I) EQL  0 THEN GOTO JM$
        GRWQ=  IF ITERATION THEN ROW(0) ELSE ROW(I)$
        IF MQ EQL  B1 OR ITERATION THEN
        BEGIN
           IF DECLFUEL(G)  LSS  0 THEN GOTO JM$
           BCLIMB= TRUE$ CL(DECLFUEL(G), DECLTIME(G))$
           BCLIMB=  FALSE$
           DDG= DECLDIST(G)$  DTG= DECLTIME(G)*600 $
           IF ED THEN PREP(DDG, DTG, TRUE, FALSE)$
        END$
        IF NOT FBQ THEN GOTO JM2$
JM1..
        AID4=  IF TDEV(G) LSS -10 THEN 64.0&+6
        ELSE TABLE(1,G,MW)$
        IF GRWQ GEQ AID4 THEN GOTO JM$
        IF NOT FBQ THEN GOTO JM3$
JM2..
        GRWP=  GRWQ$ GRWQ= GRWQ + 0* AIRDIST(G)/ 2*
        (IF G EQL 1 THEN 2.366+0.0000722* GRWQ ELSE
        IF G EQL 2 THEN 0.35 + 0.0000792* GRWQ
        ELSE -0.83+ 0.000089* GRWQ)$
        AA= TABLE(0,G,RANGE)$ GRWQ=  GRWP$
        IF AA  LSS  0 THEN GOTO JM$
        CL(AIRDIST(G)*10**4/AA,TIME(G)/600)$
        TMG= TIME(G)$
        IF ED THEN PREP(LE,   TMG ,FALSE,FALSE)$
        IF NOT FBQ THEN GOTO JM1$
JM3..
        IF MQ EQL B3 AND NOT ITERATION THEN
        BEGIN
           BCLIMB=
           TRUE$ CL(DECLFUEL(G),DECLTIME (G))$
           BCLIMB=  FALSE$
           DDG= DECLDIST(G)$  DTG= DECLTIME(G)*600 $
           IF ED THEN PREP(DDG, DTG, FALSE, TRUE)$
        END$
        GOTO ENDQ$
JM..
        GRWQ= 0$ QUANT= 64.0&+6$
ENDQ..
      END$
    COMMENT QQ$


      ED=  IF BI AND BG THEN TRUE ELSE FALSE$
      FOR AA= 0 STEP 1 UNTIL F(Q) DO
```

```
BEGIN
   ROW(AA)= ROWQ(AA)= 0$   QUANTQ(AA)= 64.0&+6
END$
QUANT= 0$ BCLIMB=  FALSE$
IF FBQ THEN
BEGIN
   ROW(0)= RR$ QUANTQ(0)= 0
END
ELSE
BEGIN
   ROW(F(Q))= RR$ QUANTQ(F(Q))= 0
END$
IF ED THEN
BEGIN
   DISTANCE= FLTIME= 0$ BURN= ROW(0)$
   EDITING$   WRITE(FO4, RR)$
END$
IF FBQ THEN
BEGIN
   B1= 0$ B2= 1$ B3= Q-1
END
ELSE
BEGIN
   B1= Q-1$ B2= -1$ B3= 0
END$

COMMENT ZONE CYCLE BEGINS$
FOR MQ= B1 STEP B2 UNTIL B3 DO
BEGIN
   G1=  IF FBQ THEN MQ+1 ELSE MQ$
   G2=  IF FBQ THEN MQ ELSE MQ+1$
   M= IO*MQ+G1Q+(IF IO EQL -1 THEN -1 ELSE 0)$

   COMMENT A CYCLE FOR GRAPH POINTS ALONG MERIDIAN
           BEGINS$
   FOR JJ= STOREI(G1) STEP 1 UNTIL STOREIQ(G1) DO
   BEGIN
      G3= SQ(G1,JJ)$ J= JQ(G1,JJ)$
      COMMENT A CYCLE FOR GRAPH POINTS
              ALONG NEXT MERIDIAN BEGINS$
      FOR II= STOREI(G2) STEP 1 UNTIL STOREIQ(G2) DO
      BEGIN
         IF NOT BI THEN
         BEGIN
            IF(M GEQ 3 AND M LEQ 8) AND ABS
            (II-JJ) GEQ 10 THEN GOTO CI$
            BNQ= (FBQ AND IO EQL -1) OR (NOT FBQ AND
            IO EQL 1)$
            BN= (FBQ AND IO EQL  1) OR (NOT FBQ AND
            IO EQL -1)$
            IF M LSS 3 THEN
            BEGIN
               IF(BNQ AND BK
               (M,JJ,II)) OR (BN AND BK(M,II,
                    JJ)) THEN GOTO CI
                ENDS
                IF M GTR 8 THEN
                BEGIN
                    IF (BNQ AND
                    BKQ(M,JJ,II)) OR (BN AND BKQ
                    (M,II,JJ)) THEN GOTO CI
                END
            END$
            G4= SQ(G2,II)$ I= JQ(G2,II)$
            IF FBQ THEN GEOMGRID(G4,G3)ELSE
            GEOMGRID(G3,G4)$
            PART2GEOM$
            IF MQ NEQ B1 THEN GG=STOREG(MQ - B2)$

            COMMENT A CYCLE FOR FLIGHT LEVELS BEGINS$
            FOR G= STOREG(MQ) STEP 1 UNTIL
            STOREGQ(MQ) DO
            BEGIN
                IF NOT FBQ AND MQ EQL 0 THEN
                BEGIN
                    TEL = 0$ IF G NEQ GG THEN
                    GOTO ENDG$
                    N1=  ROW(0)$ ROW(0)=  GRWQ=
                    MAXTOW-20000$
REPEAT..
                    TEL = TEL +1$ IF G  LSS  1
                    THEN
                    BEGIN
                        G=  GG$ GRWQ=  0$
                        GOTO ENDG
                    END$
                    METPROC(G)$ DESCLIMB(G)$
                    IF DECLFUEL(G)  LSS  0 THEN
                    BEGIN
                        TEL= 0$ G=  G-1$ GOTO REPEAT
                    END$
                    QQ(-1)$  IF GRWQ EQL 0 THEN GOTO ENDG$
                    N2=  ROW(I) - GRWQ$ IF N2  GTR  10
                    AND ROW(0)  EQL  MAXTOW THEN GOTO
                    ENDG$
                    ROW(0)=  ROW(0) +  (IF TEL  GTR  10
                    THEN N2/2 ELSE N2)$
                    GRWQ=  ROW(0)$ IF ABS(N2)  LSS  10
                    OR TEL  GTR  10 THEN
                    BEGIN
                        ROW(0)=  N1$ GOTO ITER
                    END
                    ELSE GOTO REPEAT$
ITER..
                END
                ELSE
                BEGIN
                    METPROC(G)$
```

```
                              IF MQ EQL 0 OR MQ EQL Q-1 THEN
                              BEGIN
                                 GRWQ= RRS DESCLIMB(G)
                              END$
                              IF FBQ THEN QQ(-1) ELSE QQ(1)$
                              IF GRWQ EQL  0 THEN GOTO ENDG
                           END$
                           IF QUANT LEQ QUANTQ(J) THEN
                           BEGIN
                              ROW(J)= GRWQ$ QUANTQ(J)= QUANT$
                              ROWQ(J)= I$ IF BI AND NOT BG
                              THEN STOREG(MQ)= G$
                              IF ITERATION THEN G=  GG
                           END$
ENDG..
                        END$  COMMENT FLIGHT-LEVEL CYCLE$
EI..
                     END$  COMMENT ALONG-NEXT-MERIDIAN CYCLE$

                  END$  COMMENT ALONG-CURRENT-MERIDIAN CYCLE$

                  IF BI AND NOT BG THEN
                     STOREGQ(MQ) = STOREG(MQ)
               END$  COMMENT ZONE CYCLE$


                  IF ROW(0) EQL 0 THEN GOTO EIND$
                  IF BI THEN GOTO EIND$  J=IF FBQ THEN F(Q) ELSE F(0)$
               FOR MQ= B3 STEP -B2 UNTIL B1 DO
               BEGIN
                  K=  IF FBQ THEN MQ ELSE MQ+1$ I= ROWQ(J)$
                  STOREIQ(MQ +(IF B3 EQL 0 THEN 1 ELSE 0))= I-F(K-1)+
                  STOREI(K)-1$ J= I$
               END$
               FOR MQ= 0 STEP 1 UNTIL Q DO STOREI(MQ)= STOREIQ(MQ)$
EIND..
               TOWQ= ROW(0)$ M1=  IF FBQ THEN QUANTQ(F(Q)) ELSE
               QUANTQ(0)$
               LW= ROW(F(Q))$ IF ED THEN
               BEGIN
                     WRITE(' ')$
                     WRITE(F05, TOWQ-LW)$
                     WRITE(F06, IF FB EQL 1 THEN M1/100 ELSE
                  ((TOWQ-LW)*15+1083.3*FLTIME/600)/100)$
               END$
            END$  COMMENT SPACEOPT$

            PROCEDURE FF(UU,VV)$   INTEGER ARRAY UU,VV$
            BEGIN
               F(-1)= -1$ FOR K= 0 STEP 1 UNTIL Q DO
               F(K)= F(K-1)+VV(K)-UU(K)+1
            END$  COMMENT FF$

            PROCEDURE PROCES(S1,S2,S3,PR,FFBB,BB,FACTORISATION)$
```

```
          INTEGER S1,S2,S3,FFBB$ REAL PR$
          BOOLEAN BB,FACTORISATION$
          COMMENT S1 AND S2 DETERMINE THE LIMITS OF FLIGHT
          LEVELS.
          S3 IS A DUMMY,EXCEPT WHEN
          FACTORISATION IS TRUE,
          THEN S3 IS A DATUM FOR THE FLIGHT
          LEVEL
          IN WHICH THE M,F,P, IS COMPUTED,
          PR IS PERCENTAGE TRIP FUEL
          FFBB DETERMINES WETHER COSTS(+1),
          FUEL(0) OR TIME(-1) IS OPTIMIZED,
          BB..USE LANDING WEIGHT(TRUE),USE
          TAKE OFF WEIGHT(FALSE)
          FACTORISATION(TRUE) INITIALISES
          PROCES IN THE HORIZONTAL ON THE
          BASIS OF FLIGHT TIME,
          FOLLOWED BY COMPILATION OF FLIGHT
          PLAN IN THE VERTICAL ON THE BASIS
          OF COSTS,FUEL OR TIME$
          BEGIN
            INTEGER NN$
            LOCAL LABEL ENDPR,ING$

            PROCEDURE AA$
            COMMENT SPECIFICATION OF LIMITS IN HORIZONTAL OF
                 GRAPH POINTS ALONG MERIDIAN$
            BEGIN
            LOCAL LABEL UTRECHT$
              STOREI(0)=STOREIQ(0)=ST - A(G1Q-1)$
              STOREI(Q)= STOREIQ(Q)= ST1-A(G2Q-1)$
              IF ROUTE GEQ 1111 THEN GOTO UTRECHT$
              FOR K= 1 STEP 1 UNTIL Q-1 DO
              BEGIN
                STOREI(K)= 1$ STOREIQ(K)= E(K)
              END$
UTRECHT..
              FF(STOREI,STOREIQ)
            END$  COMMENT AA$

            PROCEDURE       AAA(F1,F2)$ INTEGER F1,F2$
            COMMENT SPECIFICATION OF LIMITS IN THE VERTICAL$
            FOR K= 0 STEP 1 UNTIL Q-1 DO
            BEGIN
              STOREG(K) = F1$  STOREGQ(K) = F2$
            END$  COMMENT AAA$

            PROCEDURE EP$
            IF TOWQ GEQ MAXTOW OR TOWQ EQL 0 THEN
            BEGIN
                WRITE('SUPER CRITICAL')$
                GOTO ENDPR$
            END$  COMMENT EP$
```

```
           INTEGER PROCEDURE SS$
           SS= GRW+RESERVE+PR*((GRW+RESERVE)*
           (MINTIME*0.08-0.15)-500*MINTIME+7300+TAXI)$

           INTEGER PROCEDURE SSS$
           SSS= GRW+RESERVE+PR*(TOWQ-LW+TAXI)$

           INTEGER PROCEDURE SSSS$
           SSSS= TOWQ*(GRW+RESERVE+PR*TAXI)/((1+PR)*NN-PR*TOWQ)$

           PROCEDURE NNQ(A)$ VALUE A$ INTEGER A$
           NN= IF NOT BB THEN TOW ELSE A$

           PROCEDURE TW$
           COMMENT SAFEGUARDING AGAINST OVERLOADINGS$
           IF TOWQ GEQ MAXTOW OR TOWQ EQL  0 THEN
           BEGIN
              WRITE(PEJT)$  WRITE('DECR LANDING W')$
              GOTO ENDPR
           END$


           COMMENT IF FACTORIZATION IS TRUE, THEN THE OPTIMIZATION
                   TAKES PLACE FIRST IN THE HORIZONTAL AND THEN
                   IS FOLLOWED BY IN THE VERTICAL$
           IF FACTORISATION THEN
           BEGIN
              AA$ AAA(S3,S3)$ NNQ(SS)$ ROUTE= 1111$
              SPACEOPT(IF BB THEN FALSE ELSE TRUE,-1,FALSE,
              TRUE,NN)$ TW$
              AAA(S1,S2)$ GOTO ING
           END
           ELSE
           BEGIN
              AA$ AAA(S1,S2)$ NNQ(SS)$
              SPACEOPT(IF BB THEN FALSE ELSE TRUE,FFBB,FALSE,
              FALSE,NN)$ TW$

ING..
              EP$ NNQ(SSS)$
              SPACEOPT (IF BB THEN FALSE ELSE TRUE,FFBB, TRUE,
              FALSE,NN)$ TW$
              EP$ NNQ(SSSS)$
              SPACEOPT(TRUE,FFBB,TRUE,TRUE,NN)$ EP$
              IF FACTORISATION THEN ROUTE=  1000
           END$

ENDPR..
         END$  COMMENT PROCESS$

         IF ROUTE GEQ 1111 THEN
         BEGIN
            READ(PCF('DC10F'),  ST, ST1, DONE, ERR)$
            G1Q= ZONEI(ST)$
            FOR K= 1 STEP 1 UNTIL Q-1 DO
            BEGIN READ(PCF('DC10F'), STOREI(K), DONE, ERR)$
```

```
                    WRITE(F02, 1, STOREI(K))$
              STOREI(K)= STOREIQ(K)= STOREI(K)-A(G1Q+IO*K-1)$
              END$
              G2Q = ZONEI(ST1)$
              WRITE(F02, 2, ST,ST1)$
           END$
            GEOMGRID(ST,ST1)$
            MINTIME=GEODIST(LA1,LO1,LA2,LO2)/465$
            CTQ$
           E(0)= E(Q)= 1$ FOR K= 1 STEP 1 UNTIL Q-1 DO
           E(K)= A(G1Q+IO*K)-A(G1Q+IO*K-1)$

           COMMENT
           1. WHEN A FLIGHT PLAN FOR THE OPTIMAL COST TRACK IS NEEDED
           CALL 'PROCES(1, 3, 1, 0.03, 1, TRUE, FALSE)'
           2. WHEN A FLIGHT PLAN FOR THE OPTIMAL FUEL TRACK NEEDED
           CALL 'PROCES(1, 3, 1, 0.03, 0, TRUE, FALSE)'
           3. WHEN A FLIGHT PLAN FOR THE LEAST TIME TRACK NEEDED
           CALL 'PROCES(1, 3, 1, 0.03,-1, TRUE, FALSE)'$

           PROCES(1, 3, 1, 0.03, 0, TRUE, FALSE)$
           PROCES(1, 3, 1, 0.03, -1, TRUE, FALSE)$

           GOTO RUN
        END$
 DONE.. END$
        WRITE(' ')$  WRITE(' ')$
        WRITE('      *** END OF RUN ***')$
        GO TO EOP$
 ERR..
        WRITE('      *** INPUT DATA ERROR ***')$
 EOP..
```

Program Listing (DC8).

```
Δ ELT DC8,1,810528,065734                                        ΔC*>WK#AAC Δ]Δ  ΔX
   COMMENT KNMI/KLM 020169 SPG NAV 20 JNG,
   NAVIGATION-FLIGHTPLANNING-MODULE FOR THE PRODUCTION OF
   FLIGHTPLANS OVER THE NORTH ATLANTIC.
   THE MAIN FEATURES
  ARE:
   1. INCLUSION OF 6 METEOROLOGICAL PARAMETER FIELDS,
   TEMPERATURE AND GEOPOTENTIAL FIELDS FOR 300 250 AND 200 MB
   VALID FOR TWO STANDARD TIMES 12 HOURS APART.
   2. NAVIGATIONAL GRID BASED ON REPORTING POINTS OF THE KLM.
   3. INCLUSION OF A DYNAMIC PROCESS BASED ON INTERPOLATION OF
   TWO PARAMETER FIELDS.
   4. OPTIMALISATION OF FUEL, COSTS OR FLIGHT TIME IN SPACE AS
   WELL AS IN THE HORIZONTAL AS IN THE VERTICAL.
   5. IN - AND OUTBOUND TRAFFIC (IO= 1 OUTBOUND IO= -1 INBOUND).
   6. FLIGHTPLAN PRODUCTION FOR OPTIMUM TRACKS, SPECIFIC ROUTES
   ETC. E.G. ALTERNATIVE ATC MINIMUM ROUTES.
   7. BLOCKING E.G. SECTOR BLOCKING OVER THE NORTH ATLANTIC
   BLOCKING IN AIRWAYS AND ATC RESTRICTED AREAS.
   8. STANDARD CRUISE AND USE OF PERFORMANCE TABLES$

      CONSTANT KQ= 14 , W= 3$
```

```
INTEGER     P1,P2,P3,Q1,Q2,Q3,S,TTT,
G1,G2,G3,G4,B1,B2,B3,MQ,G1Q,G2Q,Q,
RES1,RES2,RES3,I,J,N1,N2,N3,M,G,
M0,M1,M2,M3,M4,M5,M6,K,I0,
K1,K2,L1,L2,K1Q,K2Q,L1Q,L2Q,KK1,KK2,LL1,LL2,
ENDU,TEM,TAS,FLUR,TAXI,TOW,GRW,RESERVE,MAXTOW,MAXLW,
DATE,ST,ST1,ROUTE,DISTANCE,FLTIME,BURN,
GRWQ,TOWQ,LW,COST,LE,TEL$
REAL    C0,CG,P,C1,X1,X2,Y1,Y2,X,Y,E1,E2,E3,
GG1,GG2,D,ANGLE,LALAT,LALONG,TT,LA1,LO1,LA2,LO2,
XX1,XX2,YY1,YY2,AID1,AID2,AID3,AID4,XSTER,YSTER,
MINTIME$
INTEGER ARRAY HT(1..72,4..19), A(-1..KQ+1), V(0..41),
SR(1..3,0..19), MW(1..3,0..6), CD,CT,CF(1..3,0..17), CTC,
CFC(0..7,0..8), CDC(0..6,0..8)$
ARRAY DD(0..KQ)$
BOOLEAN ED$
BOOLEAN ARRAY BK(0..2,1..6,1..6), BKQ(9..KQ,1..6,1..6)$
FORMAT F01(E2, S10, D7.0, A5),
       PEJT (E2, ' ', A1),
       FOT(X5, *I6, A2),
       F02(X5, *I7, A2)$
STRING STR(10)$

INTEGER PROCEDURE INTERPOL(F,F1,F2,F1Q,F2Q)$
INTEGER F,F1,F2,F1Q,F2Q$
COMMENT LINEAR INTERPOLATION$
INTERPOL =  ((F- F1) * F2Q + (F2 - F) * F1Q) / (F2 - F1)$

REAL PROCEDURE SUM1(A, B, ARR)$
VALUE A, B$
INTEGER A, B$  ARRAY ARR$
BEGIN  INTEGER I$  REAL S$
   S= 0$
   FOR I= A STEP 1 UNTIL B DO
      S= S + ARR(I)$
   SUM1 = S$
END$  COMMENT SUM1$

REAL PROCEDURE SUM2(A, B, ARR1, ARR2)$
VALUE A, B$
INTEGER A, B$  ARRAY ARR1, ARR2$
BEGIN  INTEGER I$  REAL S$
   S= 0$
   FOR I= A STEP 1 UNTIL B DO
      S = S + ARR1(I)*ARR2(I)$
   SUM2 = S$
END$  COMMENT  SUM2$


PROCEDURE WW(AA, BB) $   REAL AA$ STRING BB$
WRITE(F01, BB, AA)$
```

```
P1 =  31000$  P2 =  35000$  P3 =  39000$  Q1 =  30065$
Q2 =  33999$  Q3 =  38662$  TEL =  0$
FOR S= 0 STEP 1 UNTIL 5 DO
BEGIN
   RES1= S*12$
   FOR J=19 STEP -1 UNTIL  4 DO
   FOR I= RES1 + 1 STEP 1 UNTIL
   RES1 + 12 DO READ(PCF('NASAD'), HT(I,J), ERR, ERR)$
END$
FOR S=0 STEP 1 UNTIL 5 DO
  BEGIN RES1 =S*12$  WRITE(PEJT)$
     FOR J=19 STEP -1 UNTIL 4 DO
        WRITE(F02, 12, FOR I=RES1 + 1
              STEP 1 UNTIL RES1 + 12 DO HT(I,J))$
   END$

COMMENT ARRAY 'HT' IS REFILLED WITH INTERPOLATED PARAMETERS
        WHICH APPLY IN GRID POINTS AT FLIGHT LEVELS 31000,
        35000, AND 39000 FT$
FOR I= 1 STEP 1 UNTIL 12, 37 STEP 1 UNTIL 48 DO
   FOR J= 4 STEP 1 UNTIL 19 DO
BEGIN
   M0= HT(I,J)$
   M1= ENTIER(M0/100)$ M2= M0-100*M1$
   M0= HT(I+12,J)$
   M3= ENTIER(M0/100)$ M4= M0-100*M3$
   M0= HT(I+24,J)$
   M5= ENTIER(M0/100)$ M6= M0-100*M5$ -
   HT(I,J)    = 100*INTERPOL(P1,Q1,Q2,M1,M3)+
   INTERPOL(P1,Q1,Q2,M2,M4)$
   HT(I+12,J)= 100*INTERPOL(P2,Q2,Q3,M3,M5)+
   INTERPOL(P2,Q2,Q3,M4,M6)$
   HT(I+24,J)= 100*INTERPOL(P3,Q2,Q3,M3,M5)+
   INTERPOL(P3,Q2,Q3,M4,M6)
END$
COMMENT THIS ROUTINE CONCLUDES THE PRODUCTION OF THE
METEOROLOGICAL PARAMETERS GEOPOTENTIAL AND TEMPERATURE IN THE
FLIGHT LEVELS$

COMMENT RELATIONSHIP BETWEEN GRAPH POINT NUMBERS AND ZONE INDICES$
A(-1)= -1$ A(0)= 0$ A(1)= 2$ A(2)= 6$ A(3)= 11$ A(4)= 26$
A(5)= 41$ A(6)= 56$ A(7)= 71$ A(8)= 86$ A(9)= 90$ A(10)= 96$
A(11)= 102$ A(12)= 107$ A(13)= 111$ A(14)= 114$ A(15)= 115$
V(0)= 52290477$
V(1)= 51850115$ V(2) = 53640150$
V(3)= 51000200$ V(4) = 51990539$ V(5) = 53580300$
V(6)= 55000200$ V(7) = 49980632$ V(8) = 51840849$
V(9)= 52700892$ V(10)= 53500630$ V(11)= 55490459$
V(12)= 54271005$ V(13)= 48905454$ V(14)= 51375560$
V(15)= 53735697$ V(16)= 58476263$ V(17)= 47005800$
V(18)= 48545856$ V(19)= 55506000$ V(20)= 57006000$
V(21)= 53286035$ V(22)= 54836683$ V(23)= 46156006$
V(24)= 48006000$ V(25)= 49846439$ V(26)= 50506500$
V(27)= 51006750$ V(28)= 52006900$ V(29)= 45006300$
```

```
V(30)= 46176459$ V(31)= 47506600$ V(32)= 48576826$
V(33)= 50007050$ V(34)= 43836608$ V(35)= 44846867$
V(36)= 45327179$ V(37)= 45467385$ V(38)= 41287003$
V(39)= 42367099$ V(40)= 43007250$ V(41)= 40647378$

COMMENT DETERMINATION OF CONNECTIVITY AMONG GRAPH POINTS.
        FALSE=CONNECTED, TRUE=BLOCKED $
FOR I= 1 STEP 1 UNTIL 6 DO
FOR J= 1 STEP 1 UNTIL 6 DO
BEGIN
   FOR MQ= 0,1,2 DO BK(MQ,I,J) = TRUE$
   FOR MQ= 9 STEP 1 UNTIL  KQ DO BKQ(MQ,I,J) = TRUE
END$
BK(1,1,1)= BK(2,1,1)= BK(1,2,2)= BK(2,2,2)=
BKQ(9,1,1)= BKQ(10,1,1)= BKQ(11,1,1)= BKQ(12,1,1)=
BKQ(13,1,1)= BKQ(14,1,1)= BKQ(9,2,2)= BKQ(10,2,2)=
BKQ(11,2,2)= BKQ(12,2,2)= BKQ(13,2,2)= BKQ(10,3,3)=
BKQ(11,3,3)= BKQ(13,3,3)= BKQ(11,4,4)= BK(2,3,4)= BK(2,4,5)=
BKQ(12,4,4) = BKQ(10,6,6) = BK(1,1,2) =
BK(1,2,3) = BK(1,2,4) = BK(2,2,3) =
BKQ(9,1,2)= BKQ(9,2,3)= FALSE$
BKQ(9,2,5)= BKQ(9,3,4)= BKQ(9,3,5)= BKQ(9,4,6)= BKQ(10,2,1)=
BKQ(10,4,3)= BKQ(10,5,3)= BKQ(10,5,4)= BKQ(10,6,5)=
BKQ(11,1,2)= BKQ(11,3,4)= BKQ(11,5,4)= FALSE$
BKQ(11,6,5)= BKQ(12,2,1)= BKQ(12,3,2)= BKQ(12,4,3)=
BKQ(12,5,4)= BKQ(13,1,2)= BKQ(14,2,1)= BKQ(14,3,1)= BK(0,1,1)=
BK(0,1,2) =FALSE$


BEGIN
   REAL ARRAY AB(0..10), LENGTH(1..10),
   TIME,AIRDIST,TDEV,WIX,WIY,TAX,TAY(1..W)$

   INTEGER PROCEDURE ZONEI(Q)$  INTEGER Q$
   COMMENT THIS SUBROUTINE DETERMINES THE INDEX OF THE
   ZONE ASSOCIATED WITH A STATIONSNUMBER$
   BEGIN
      LOCAL LABEL  AGAIN$ INTEGER I$
      I=  -1$
AGAIN..
      I= I+1$ IF A(I)-Q LSS 0 THEN GOTO AGAIN$
      ZONEI= I
   END$  COMMENT ZONEI$

   PROCEDURE DT(K)$ INTEGER K$
   COMMENT THIS SUBROUTINE DISSECTS LATITUDE AND
   LONGITUDE FROM THE COMPRESSED COORDINATES.
   EAST OF GREENWHICH THE SIGN OF LONGITUDE CHANGES$
   BEGIN
      INTEGER K0$
      K0= ENTIER(V(K)/10000)$
      LALAT= K0/100$
      LALONG= (V(K)-10000*K0)/100
   END$  COMMENT DISSECT$
```

```
PROCEDURE LIS(U)$ INTEGER U$
COMMENT DETERMINATION OF LATITUDE AND LONGITUDE FOR
        GRAPH POINTS$
BEGIN
   IF U EQL 19 THEN DT(12)
   ELSE IF U LEQ 11 THEN
   BEGIN
      DT(U)$ IF U EQL 0 OR U EQL 1 OR
      U EQL 2 THEN LALONG= -LALONG
   END
   ELSE IF U LEQ 26 THEN
   BEGIN
      LALAT= 35+U$ LALONG= 10
   END
   ELSE IF U LEQ 41 THEN
   BEGIN
      LALAT= 20+U$ LALONG= 20
   END
   ELSE IF U LEQ 56 THEN
   BEGIN
      LALAT= 5+U$ LALONG= 30
   END
   ELSE IF U LEQ 71 THEN
   BEGIN
      LALAT= U-11$ LALONG= 40
   END
   ELSE IF U LEQ 86 THEN
   BEGIN
      LALAT= U-26$ LALONG= 50
   END
   ELSE DT(U-74)
END$  COMMENT LIS$

PROCEDURE CTQ$
COMMENT THIS SUBROUTINE PREPARES THE TIME INSTANTS AT WHICH
IN EACH ZONE THE METEOROLOGICAL PARAMATER WILL BE
DERIVED FROM BOTH PARAMETER FIELDS, COMPOSITE CHARTS
IN TIME ARE SIMULATED$
BEGIN
   INTEGER J0,M0$
   REAL A,SUM,RES$
   A= 0.4$
   J0 = IF I0 EQL 1 THEN KQ ELSE 0$  SUM= 0$
   FOR I= KQ-J0 STEP I0 UNTIL J0 DO
   BEGIN
      SUM= SUM+(IF I GTR 3 AND I LSS 8 THEN 2*A ELSE A)$
      DD(I)= SUM
   END$
   M0 = ZONEI(ST)$  RES= IF I0 EQL 1 THEN DD(M0) ELSE
   DD(M0-1)$
   FOR I= 0 STEP 1 UNTIL KQ DO DD(I)=  IF TT GTR 12
   THEN TT-12 ELSE TT+A/2+DD(I)-RES$
END$  COMMENT CTQ$
```

```
REAL PROCEDURE HH(V,W,F,CHOICE)$ INTEGER F,V,W$
BOOLEAN CHOICE$
COMMENT COMPUTATION OF GRID POINT VALUES FOR GEOPOTENTIAL
        OR TEMPERATURE (CHOICE = TRUE OR FALSE) $
BEGIN
   M5= (F-1)*12$
   M6= (F+2)*12$
   M3= HT(V+M5,W)$
   M4= HT(V+M6,W)$
   M1= ENTIER(M3/100)$
   M2= ENTIER(M4/100)$
   IF NOT CHOICE THEN
   BEGIN
      M1= M3-100*M1$
      M2= M4-100*M2
   END$
   HH= ((12-DD(M))*M1+DD(M)*M2)/12
END$  COMMENT HH$

REAL PROCEDURE GEOP(V,W,C,CHOICE)$ INTEGER C$
REAL V,W$ BOOLEAN CHOICE$
COMMENT COMPUTATION OF GEOPOTENTIAL OR TEMPERATURE IN
        ARBITRARY POINTS$
BEGIN
   INTEGER V1,W1$ REAL A,B,B1,B2,B3,B4$
   V1= ENTIER(V)$ W1= ENTIER(W)$
   B1=  IF CHOICE THEN HH(V1,W1,C,TRUE)
   ELSE HH(V1,W1,C,FALSE)$
   B2=  IF CHOICE THEN HH(V1+1,W1,C,TRUE)
   ELSE HH(V1+1,W1,C,FALSE)$
   B3=  IF CHOICE THEN HH(V1+1,W1+1,C,TRUE)
   ELSE HH(V1+1,W1+1,C,FALSE)$
   B4= IF CHOICE THEN HH(V1,W1+1,C,TRUE)
   ELSE HH(V1,W1+1,C,FALSE)$
   A= V1+1-V$ B= W1+1-W$
   GEOP= A*B*B1+(1-A)*B*B2+(1-A)*(1-B)*B3+A*(1-B)*B4
END$  COMMENT GEOP$

PROCEDURE XYTRANSF(LAT,LON)$ REAL LAT,LON$
BEGIN
   REAL R,S$
   R= (45-LAT/2)*CG$
   S= (80-LON) * CG$
   X= 0.7909*C0*SIN(R) *SIN(S)/COS(R)$
   Y= (P-C0*SIN(R)*COS(S)/COS(R))*0.7909
END$  COMMENT XYTRANSF$

REAL PROCEDURE GEODIST(DDD,LAT)$ REAL DDD,LAT$
COMMENT COMPUTATION OF DISTANCE ON THE SPHERE FROM
        EUCLIDEAN DISTANCE$
GEODIST=  219.721*DDD*COS((45-LAT/2)*CG)**2$

PROCEDURE GEOMGRID(PP,GG)$ INTEGER PP,GG$
```

```
COMMENT COMPUTATION OF EUCLIDEAN DISTANCE AND UNIT VECTORS
BEGIN
   LIS(PP)$ LA1= LALAT$ LO1= LALONG$
   LIS(GG)$ LA2= LALAT$ LO2= LALONG$
   XYTRANSF(LA1,LO1)$ X1= X$ Y1= Y$
   XYTRANSF(LA2,LO2)$ X2= X$ Y2= Y$
   E3= SQRT((X2-X1)**2+(Y2-Y1)**2)$
   E1= (X2-X1)/E3$
   E2= (Y2-Y1)/E3
END$  COMMENT GEOMGRID$

PROCEDURE PART2GEOM$
COMMENT DETERMINATION OF LENGTH OF SEGMENT BY SUMMATION OF
        CONTRIBUTIONS FROM SUBSEGMENTS$
BEGIN
   K1= ENTIER(X1)$ K2= K1+1$ L1= ENTIER(Y1)$
   L2= L1+1$ K1Q= ENTIER(X2)$ K2Q= K1Q+1$
   L1Q= ENTIER(Y2)$ L2Q= L1Q+1$
   D= GEODIST(E3,(LA1+LA2)/2)$
   N1= ABS(K1Q-K1)$
   N2= ABS(L1Q-L1)$
   N3 = IF K1Q EQL K1 AND L1Q EQL L1 THEN 0
   ELSE IF N1 LEQ N2 THEN N2 ELSE N1$
   BEGIN
      INTEGER I,J$
      LOCAL LABEL LAB1$
      AB(0)= 0$ AB(N3+1)= 1$
      IF N3 EQL 0 THEN GOTO LAB1$ IF N3 EQL N1 THEN
      BEGIN
         IF K1 GTR K1Q THEN
         BEGIN
            RES1= K1$
            RES2= K2Q$ RES3= -1
         END
         ELSE
         BEGIN
            RES1= K2$ RES2= K1Q$
            RES3= 1
         END$
         FOR I= RES1 STEP RES3 UNTIL RES2 DO
         BEGIN
            J= RES3*(I-RES1)+1$ AB(J)= (I-X1)/
            (X2-X1)
         END
      END
      ELSE
      BEGIN
         IF L1 GTR L1Q THEN
         BEGIN
            RES1= L1$
            RES2= L2Q$ RES3= -1
         END
         ELSE
         BEGIN
```

```
                      RES1= L2$ RES2= L1Q$
                      RES3=  1
                   END$
                   FOR I= RES1 STEP RES3 UNTIL RES2 DO
                   BEGIN
                      J= RES3*(I-RES1)+1$ AB(J)= (I-Y1)/
                      (Y2-Y1)
                   END
                END$

LAB1..
                FOR I= 1 STEP 1 UNTIL N3+1 DO
                BEGIN
                   AID2= ((AB(I)+AB(I-1))*LA2+(2-(AB(I)
                   +AB(I-1)))*LA1)/2$
                   LENGTH(I)= GEODIST(E3*(AB(I)-AB(I-1)), AID2)
                END
             END
          END$  COMMENT PART2GEOM$

          PROCEDURE METPROC(S)$ INTEGER S$
          COMMENT PROCESSING METEOROLOGICAL PARAMETERS$
          BEGIN
             INTEGER I$
             ARRAY TEMP,WNDX,WNDY,ENDUR(1..N3+1)$
             GG1= GEOP(X1,Y1,S,TRUE)$
             GG2= GEOP(X2,Y2,S,TRUE)$
             ANGLE= C1*(GG2-GG1)/(470*D*SIN((LA1+LA2)*CG/2))$
             XX1= X1$ YY1= Y1$
             FOR I= 1 STEP 1 UNTIL N3+1 DO
             BEGIN
                XX2= AB(I)*X2+(1-AB(I))*X1$
                YY2= AB(I)*Y2+(1-AB(I))*Y1$
                AID2= ((AB(I)+AB(I-1))*LA2+(2-(AB(I)+AB(I-1)))*
                LA1)/2$
                XSTER= (XX1+XX2)/2$
                YSTER= (YY1+YY2)/2$
                KK1= ENTIER(XSTER)$ KK2= KK1+1$ LL1= ENTIER(YSTER)$
                LL2= LL1+1$
                TEMP(I)= -GEOP(XSTER,YSTER,S,FALSE)$
                AID1= C1*E3/(D*SIN(AID2*CG))$ AID2= XSTER-KK1$
                AID3= YSTER-LL1$
                WNDX(I)= -AID1*(AID2*(HH(KK2,LL2,S,TRUE)-HH
                (KK2,LL1,S,TRUE))+(1-AID2)*(HH(KK1,LL2,S,TRUE)
                -HH(KK1,LL1,S,TRUE)))$
                WNDY(I)=  AID1*(AID3*(HH(KK2,LL2,S,TRUE)-HH
                (KK1,LL2,S,TRUE))+(1-AID3)*(HH(KK2,LL1,S,TRUE)
                -HH(KK1,LL1,S,TRUE)))$
                TAS= 0.8034*38.9826*SQRT(273.16+TEMP(I))$
                ENDUR(I)= 36000*LENGTH(I)/(TAS*SQRT(1-ANGLE**2)+
                (WNDX(I)*E1+WNDY(I)*E2))$
                XX1= XX2$ YY1= YY2
             END$
             ENDU= SUM1(1, N3+1, ENDUR)$
             TEM= SUM2(1, N3+1, ENDUR, TEMP)/ENDU$
```

```
      TAS= 0.8034*38.9826*SQRT(273.16+TEM)$
      RES1= 31+(S-1)*4$
      TDEV(S)= IF RES1 LEQ 35.332 THEN TEM +1.98*RES1-15 ELSE
      55+TEM$
      TIME(S)= ENDU$
      AIRDIST(S)= ENDU *TAS/36000$ LE= 0$
      WIX(S)= SUM2(1, N3+1, ENDUR, WNDX)/ENDU$
      WIY(S)= SUM2(1, N3+1, ENDUR, WNDY)/ENDU$
      TAX(S)= TAS*(E1*SQRT(1-ANGLE**2)+E2*ANGLE)$
      TAY(S)= TAS*(E2*SQRT(1-ANGLE**2)-E1*ANGLE)$
   END$  COMMENT METPROC$

   COMMENT PROCEDURES 'TABLE' AND 'TABLEQ' ARE FOR  TABLE LOOK-UP$
   INTEGER PROCEDURE TABLE(P,R,MA)$ INTEGER P,R$ INTEGER ARRAY
            MA$
   BEGIN
      INTEGER G0,D,B$ REAL R0,C$ LOCAL LABEL LAB20$
      D=IF P EQL 0 THEN 68000 ELSE IF P EQL 1 THEN -10 ELSE 80000$
      B=IF P EQL 1 THEN 5 ELSE 4000$
      C= IF P EQL 1 THEN TDEV(R) ELSE GRWQ$
      R0= (C-D)/B$
      G0= ENTIER(R0)$
      R0= R0-G0$
      IF MA(R,G0+1) EQL -1 THEN
      BEGIN
         TABLE= -1$ GOTO LAB20
      END$
      TABLE= R0*MA(R,G0+1)+(1-R0)*MA(R,G0)$
LAB20..
   END$  COMMENT TABLE$

   INTEGER PROCEDURE  TABLEQ(Q,L,MA0,MA1)$ INTEGER Q,L$
   INTEGER ARRAY MA0,MA1$
   BEGIN
      INTEGER H,S0,Z,R,S00$
      LOCAL LABEL LAB21$
      REAL T0, T00$
      H= TABLE(Q,L,MA0)$
      IF H EQL -1 THEN
      BEGIN
         TABLEQ= -1$ GOTO LAB21
      END$
      Z= IF Q EQL 2 THEN 0 ELSE IF Q EQL 3 THEN 50 ELSE 500$
      R= IF Q EQL 2 THEN 50 ELSE IF Q EQL 3 THEN 100 ELSE 1000$
      T0= (H - Z)/R$
      T00= (TDEV(L)+20)/5$
      S0= ENTIER(T0)$
      S00= ENTIER(T00)$
      T0= T0-S0$
      T00= T00-S00$
      IF MA1(S0+1,S00+1) EQL -1 THEN
      BEGIN
         TABLEQ= -1$
         GOTO LAB21
```

```
            ENDS
            TABLEQ=T0*T00*MA1(S0+1,S00+1)+(1-T0)*(1-T00)*
            MA1(S0,S00)+(1-T0)*T00*MA1(S0,S00+1)+T0*(1-T00)*MA1
            (S0+1,S00)$
LAB21..
         ENDS  COMMENT TABLEQS

         PROCEDURE READQ(Y0,Y1,Y2,ARR)$ INTEGER Y0,Y1,Y2$
         INTEGER ARRAY ARR$
         BEGIN
            FOR I= Y0 STEP 1 UNTIL Y1 DO
              FOR J=0 STEP 1 UNTIL Y2 DO
                  READ(PCF('NASAD'), ARR(I, J), ERR, ERR)
         ENDS  COMMENT READS

         PROCEDURE WRITEQ(Y0, Y1, Y2, ARR)S
         INTEGER Y0,Y1,Y2$
         INTEGER ARRAY ARR$
         BEGIN
            FOR I= Y0 STEP 1 UNTIL Y1 DO
               WRITE( FOT, Y2 + 1, FOR J= 0 STEP 1
                      UNTIL Y2 DO ARR(I, J))$
         ENDS  COMMENT WRITEQS

         READQ(1,3,19,SR!$ READQ(1,3,6,MW)$ READQ(1,3,17,CD)$
         READQ(1,3,17,CT)$ READQ(1,3,17,CF)$ READQ(0,7,8,CTC)$
         READQ(0,7,8,CFC)$ READQ(0,6,8,CDC)$

         WRITE(PEJT)$
         WRITEQ(1,3,19,SR)$
         WRITE(' ')$  WRITE(' ')$
         WRITEQ(1,3,6,MW)$
         WRITE(' ')$  WRITE(' ')$
         WRITEQ(1,3,17,CD)$
         WRITE(' ')$  WRITE(' ')$
         WRITEQ(1,3,17,CT)$
         WRITE(PEJT)$
         WRITEQ(1,3,17,CF)$
         WRITE(' ')$  WRITE(' ')$
         WRITEQ(0,7,8,CTC)$
         WRITE(PEJT)$
         WRITEQ(0,7,8,CFC)$
         WRITE(' ')$  WRITE(' ')$
         WRITEQ(0,6,8,CDC)$

         CG= 0.0174532925$ C0= 39.6296148$ C1= 21.47*32.808399$
         P= C0*SIN(31.2*CG)/COS(31.2*CG)$
 RUN..
         READ(PCF('NASAD'),  FLUR, TAXI, GRW, RESERVE, TOW,
         MAXTOW,MAXLW,DATE,IO,TTT,ROUTE,DONE, ERR)$
         WRITE(' ')$  WRITE(' ')$  WRITE(' ')$
         WRITE('      *** INPUT PARAMETERS ***')$
         WRITE(FO2, 11, FLUR,TAXI,GRW,RESERVE,TOW,
               MAXTOW,MAXLW,DATE,IO,TTT,ROUTE)$
```

```
IF ROUTE GEQ 1111 THEN
BEGIN
  READ(PCF('NASAD'),  Q, DONE, ERR)$
  WRITE(F02, 1, Q)$
     Q=Q-1$
END
ELSE
BEGIN
  READ(PCF('NASAD'),  ST,ST1, DONE, ERR)$
  WRITE(F02, 2, ST, ST1)$
   G1Q= ZONEI(ST)$ G2Q= ZONEI(ST1)$ Q= ABS(G2Q-G1Q)
END$
TT = TTT$

BEGIN
   INTEGER ARRAY STOREI,STOREIQ,STOREG,STOREGQ,E(0..Q),
   F(-1..Q)$
   FORMAT F04(X63, I6, A2),
          F05('TRIP FUEL', I6, A1),
          F06('COST', D8.2, A1)$
   REAL ARRAY DECLTIME,DECLFUEL,DECLDIST(1..W)$
   INTEGER PROCEDURE SQ(H1,H2)$ INTEGER H1,H2$
   SQ= A(G1Q+IO*H1-1)+H2$
   INTEGER PROCEDURE JQ(H1,H2)$ INTEGER H1,H2$
   JQ= F(H1-1)+H2-STOREI(H1)+1$

   PROCEDURE DESCLIMB(S)$ VALUE S$ INTEGER S$
   COMMENT COMPUTATION OF PERFORMANCE IN CLIMB OR DESCENT$
   BEGIN
      INTEGER ALT$
      LOCAL LABEL JUM$
      ALT= 31+(S-1)*4$
      DECLTIME(S)=  IF MQ EQL 0 THEN 60*TABLEQ(3,S,CT,CTC)
      /1000 ELSE 3.1+0.41*ALT$
      DECLDIST(S)=  IF MQ EQL 0 THEN TABLEQ(2,S,CD,CDC)
      ELSE 9+3*ALT$
      DECLFUEL(S)=  IF MQ EQL 0 THEN TABLEQ(4,S,CF,CFC)
      ELSE 940+8.5*ALT$ IF DECLTIME(S) LSS 0 THEN GOTO JUM$

      AID4=  DECLDIST(S)*60/(DECLTIME(S)*
      SQRT(TAX(S)**2+TAY(S)**2))$
      DECLDIST(S)= SQRT((0.75*WIX(S)+TAX(S)*AID4)**2+
      (0.75*WIY(S)+TAY(S)*AID4)**2)*(DECLTIME(S)-2)/60$
      AIRDIST(S)= (1-DECLDIST(S)/LE)*AIRDIST(S)$
      TIME(S)= AIRDIST(S)/SQRT(TAX(S)**2+TAY(S)**2)*
      36000$ LE= LE-DECLDIST(S)$
JUM..
    END$  COMMENT DECLIMB$

    REAL PROCEDURE ARCTANQ(Q1,Q2)$ REAL Q1, Q2$
    ARCTANQ=  ARCTAN(Q2/Q1)/CG+(IF Q1 LSS 0 THEN 180
    ELSE IF Q2 LSS 0 THEN 360 ELSE 0)$

    PROCEDURE LO(Z1, Z2)$ REAL Z1, Z2 $
```

```
          COMMENT COMPUTATION OF LONGITUDE FROM EUCLIDEAN COORDINATES
                  IN RECTANGULAR GRIDS$
          LO1= 80-ARCTAN(Z1/(19-Z2))/CG$

          INTEGER PROCEDURE HEADING(L1,L2)$ REAL L1,L2$
          COMMENT COMPUTATION OF HEADING IN FLIGHTS$
          BEGIN
             REAL M1,M2,ZX,ZY$
             XSTER= (X1+X2)/2$ YSTER= (Y1+Y2)/2$
             LO(XSTER,YSTER)$ M1= SIN((80-LO1)*CG)$
             M2= COS((80-LO1)*CG)$
             HEADING= ARCTANQ(-L1*M1+L2*M2,L1*M2 + L2*M1)
          END$  COMMENT HEADING$

          PROCEDURE  LINE(N1,N2)$ BOOLEAN N1,N2$
          COMMENT PRINT-OUT PROCEDURE$
          BEGIN
             INTEGER I,J,K,L,M,N,P,Q,R,S,T    $
             FORMAT P1('TOC', 2I5, X12, I4, X10, I4, 3I6, I8,  A2),
                    P2(I3, X22, I4, X10, I4, 3I6, I8, A2),
                    P3('TOD', 4I5, I6, I8, I6, 3I6, I8, A2),
                    P4(I3, 4I5, I6, I8, 4I6, I8, A2)$
             LOCAL LABEL LAB30,LAB31,LAB32$
             IF N1 THEN GOTO LAB30$
             IF N2 THEN GOTO LAB31$
             I=G3$  S=MG$  T=B3$
LAB30..      GEOMGRID(G4, G3)$
             J= HEADING (TAX(G), TAY(G))$
             K= 310 + (G-1)*40$
           IF NOT N1 AND NOT N2 THEN BEGIN
             L= TDEV(G)$
             M= SQRT(TAX(G)**2 + TAY(G)**2)$
           END$
LAB31..
             N= (IF N1 OR N2 THEN .75 ELSE 1)*
                  (WIX(G)*E1 + WIY(G)*E2)$
             M3= IF N1 OR N2 THEN
                  DECLTIME(G)*600 ELSE TIME(G)$
             P= 100*ENTIER(M3/36000) + MOD(M3, 36000)/600$
             Q= 100*ENTIER(FLTIME/36000) + MOD(FLTIME, 36000)/600$
             R= BURN - GRWQ$
           IF N1 THEN
             BEGIN
                WRITE(P1,J,K,N,DISTANCE,P,Q,R,GRWQ)$
                GOTO LAB32
             END$
           IF N2 THEN
             BEGIN
                WRITE(P2,ST1,N,DISTANCE,P,Q,R,GRWQ)$
                GOTO LAB32
             END$
             IF S EQL T THEN
                WRITE(P3,J,K,L,M,N,LE,DISTANCE,P,Q,R,GRWQ)
              ELSE WRITE(P4,I,J,K,L,M,N,LE,DISTANCE,P,Q,R,GRWQ)$
```

```
LAB32..
           ENDS COMMENT LINES

           PROCEDURE SPACEOPT(FBQ,FB,BI,BG,RR)$ INTEGER FB,RR$
           BOOLEAN FBQ,BI,BG$
           COMMENT FBQ DETERMINES WETHER THE FLIGHTPLAN
           COMPUTATION WILL BE PERFORMED BACKWARDS(FALSE) OR FOR
           WARDS(TRUE).
           FB DETERMINES WETHER COSTS(+1) OR FUEL(0) OR FLIGHT
           TIME(-1) WILL BE OPTIMIZED. RR IS TAKE OFF WEIGHT OR
           LANDINGWEIGHT.
           BI DEFINES WETHER THE NAVIGATION REGIME IS FREE IN THE
           HORIZONTAL(FALSE) OR BOUNDED BY ONE POINT(TRUE).
           BG DEFINES WETHER THE CRUISING ALTITUDE IS FREE(FALSE)
           OR BOUNDED(TRUE)$
           BEGIN
              LOCAL LABEL ITER, ENDG, CI, REPEAT, EINDS
              INTEGER AA,QUANT,II,JJ,GG,GRWP, DDG, DTG, TMG$
              BOOLEAN ITERATION, BCLIMB,BNQ,BN$
              INTEGER ARRAY ROW,ROWQ,QUANTQ(0..F(Q))$

              PROCEDURE PREP(A1,B1,AA,BB)$ INTEGER A1,B1$
              COMMENT COMPUTATION OF DISTANCE AND FLIGHT TIME, AND
                      PRINT-OUT OF THE FLIGHT PLAN$
              BOOLEAN AA,BB$
              COMMENT AA REFERS TO CLIMB, BB TO DESCENT$
              BEGIN
                 DISTANCE= DISTANCE+A1$ FLTIME= FLTIME+B1$
                 LINE(AA,BB)$ BURN= GRWQ
              END$  COMMENT PREP$

              PROCEDURE EDITING$
              BEGIN LOCAL LABEL ENDED$
                 FORMAT F03(X44, 'CH', I4, I6, D6.2, A2)$

                 PROCEDURE QP$
                 BEGIN
                 IF FB EQL 1 THEN
                    WRITE('COST')
                 ELSE IF FB EQL 0 THEN
                    WRITE('FUEL')
                 ELSE IF FB EQL -1 THEN
                    WRITE('FLTIME')$
                 GOTO ENDED$
                 END$  COMMENT QP$


                 WRITE(PEJT)$
 WRITE('                 *** DETAILED FLIGHT PLAN ***')$
                 WRITE(' ')$  WRITE(' ')$
                 IF ROUTE EQL 1111 THEN
                 BEGIN
                    WRITE('NON SPECIAL MIN TRACK')$
                    QP
                 ENDS
                 IF ROUTE EQL 4444 THEN
                 BEGIN
                    WRITE('ATC I')S   QPS
                 ENDS
                 IF ROUTE EQL 5555 THEN
                 BEGIN
                    WRITE('ATC II')S   QPS
                 ENDS
                 IF ROUTE EQL 7777 THEN
                 BEGIN
                    WRITE('GRC')S   QPS
                 ENDS
                 IF ROUTE EQL 9999 THEN
                 BEGIN
                    WRITE('SPECIAL TRACK')S   QPS
                 ENDS
                 WRITE('MIN TRACK IN SPACE')S   QPS
ENDED..
                 WRITE(F03, FLUR, DATE, TT)S
                 WRITE(' ')S
WRITE(
'NO. HEAD   FL  TMP  TAS  WIND   DIST  ACCD  TIME  ACCT  BURN  WEIGHT'
    )S
              ENDS  COMMENT EDITINGS

              PROCEDURE QQ(O)S INTEGER OS
              COMMENT COMPUTATION OF SEGMENT CONTRIBUTION, COST,
                      TIME, FUEL, WEIGHT ETC.S
              BEGIN
                 LOCAL LABEL JM1, JM2, JM3, JM, ENDOS

                 PROCEDURE CL(M1,M2)S REAL M1,M2S
                 COMMENT M1 DENOTES SEGMENT FUEL, M2 SEGMENT TIMES
                 BEGIN
                    LOCAL LABEL JUS REAL MMS
                    IF BCLIMB THEN
                    BEGIN
                       MM=  0S GOTO JU
                    ENDS
                    IF ITERATION THEN
                    BEGIN
                       MM=  IF G GEQ GG THEN -80
                       ELSE -100S GOTO JU
                    ENDS
                    MM= IF (O EQL -1 AND G GEQ GG) OR (O EQL 1 AND
                            G LSS GG) THEN 100 ELSE 80S
JU..
                    MM=  M1+ O*(GG-G)* (IF MQ NEQ B1 THEN MM ELSE 0)S
                    GRWQ= GRWQ + O*MMS
                    QUANT= QUANT+(IF FB EQL 1 THEN 1083.3*M2+15*MM
                   ELSE IF FB EQL 0 THEN MM ELSE 600*M2)S
                 ENDS  COMMENT CLS

```

```
          QUANT = QUANTQ(I)$
          ITERATION= IF NOT FBQ AND MQ EQL 0 AND 0 EQL -1
             THEN TRUE
          ELSE FALSE$ IF ROW(I) EQL  0 THEN GOTO JM$
          GRWQ=  IF ITERATION THEN ROW(0) ELSE ROW(I)$
          IF MQ EQL  B1 OR ITERATION THEN
          BEGIN
             IF DECLFUEL(G)  LSS  0 THEN GOTO JM$
             BCLIMB= TRUE$ CL(DECLFUEL(G), DECLTIME(G))$
             BCLIMB=  FALSE$
             DDG= DECLDIST(G)$  DTG= DECLTIME(G)*600 $
             IF ED THEN PREP(DDG, DTG, TRUE, FALSE)$
          END$
          IF NOT FBQ THEN GOTO JM2$
JM1..
          AID4=  IF TDEV(G) LSS -10 THEN 64.0&+6
          ELSE TABLE(1,G,MW)$
          IF GRWQ GEQ AID4 THEN GOTO JM$
          IF NOT FBQ THEN GOTO JM3$
JM2..
          GRWP=  GRWQ$ GRWQ=  GRWQ + 0* AIRDIST(G)/ 2*
          (IF G EQL 1 THEN 4.94 +0:0000693* GRWQ ELSE
          IF G EQL 2 THEN 0.54 + 0¡0001043* GRWQ
          ELSE -0.65+ 0.000117* GRWQ)$
          AA= TABLE(0,G,SR)$ GRWQ=  GRWP$
          IF AA  LSS  0 THEN GOTO JM$
          CL(AIRDIST(G)*10**4/AA,TIME(G)/600)$
          TMG= TIME(G)$
          IF ED THEN PREP(LE,   TMG ,FALSE,FALSE)$
          IF NOT FBQ THEN GOTO JM1$
JM3..
          IF MQ EQL B3 AND NOT ITERATION THEN
          BEGIN
             BCLIMB=
             TRUE$ CL(DECLFUEL(G),DECLTIME (G))$
             BCLIMB=  FALSE$
             DDG= DECLDIST(G)$  DTG= DECLTIME(G)*600 $
              IF ED THEN PREP(DDG, DTG, FALSE, TRUE)$
          END$
          GOTO ENDQ$
JM..
          GRWQ= 0$ QUANT= 64.0&+6$
ENDQ..
       END$
    COMMENT QQ$


       ED=  IF BI AND BG THEN TRUE ELSE FALSE$
       FOR AA= 0 STEP 1 UNTIL F(Q) DO
       BEGIN
          ROW(AA)= ROWQ(AA)= 0$   QUANTQ(AA)= 64.0&+6
       END$
       QUANT= 0$ BCLIMB=  FALSE$
       IF FBQ THEN
```

```
BEGIN
   ROW(0)= RR$ QUANTQ(0)= 0
END
ELSE
BEGIN
   ROW(F(Q))= RR$ QUANTQ(F(Q))= 0
END$
IF ED THEN
BEGIN
   DISTANCE= FLTIME= 0$ BURN= ROW(0)$
   EDITINGS$   WRITE(F04, RR)$
END$
IF FBQ THEN
BEGIN
   B1= 0$ B2= 1$ B3= Q-1
END
ELSE
BEGIN
   B1= Q-1$ B2= -1$ B3= 0
END$

COMMENT ZONE CYCLE BEGINS$
FOR MQ= B1 STEP B2 UNTIL B3 DO
BEGIN
   G1=  IF FBQ THEN MQ+1 ELSE MQ$
   G2=  IF FBQ THEN MQ ELSE MQ+1$
   M= IO*MQ+G1Q+(IF IO EQL -1 THEN -1 ELSE 0)$

   COMMENT A CYCLE FOR GRAPH POINTS ALONG MERIDIAN
           BEGINS$
   FOR JJ= STOREI(G1) STEP 1 UNTIL STOREIQ(G1) DO
   BEGIN
      G3= SQ(G1,JJ)$ J= JQ(G1,JJ)$
      COMMENT A CYCLE FOR GRAPH POINTS
              ALONG NEXT MERIDIAN BEGINS$
      FOR II= STOREI(G2) STEP 1 UNTIL STOREIQ(G2) DO
      BEGIN
         IF NOT BI THEN
         BEGIN
            IF(M GEQ 3 AND M LEQ 8) AND ABS
            (II-JJ) GEQ 10 THEN GOTO CI$
            BNQ= (FBQ AND IO EQL -1) OR (NOT FBQ AND
            IO EQL 1)$
            BN= (FBQ AND IO EQL  1) OR (NOT FBQ AND
            IO EQL -1)$
            IF M LSS 3 THEN
            BEGIN
               IF(BNQ AND BK
               (M,JJ,II)) OR (BN AND BK(M,II,
               JJ)) THEN GOTO CI
            END$
            IF M GTR 8 THEN
            BEGIN
               IF (BNQ AND
            BKQ(M,JJ,II)) OR (BN AND BKQ
            (M,II,JJ)) THEN GOTO CI
        END
      END$
      G4= SQ(G2,II)$ I= JQ(G2,II)$
      IF FBQ THEN GEOMGRID(G4,G3)ELSE
      GEOMGRID(G3,G4)$
      PART2GEOM$
      IF MQ NEQ B1 THEN GG=STOREG(MQ - B2)$

      COMMENT A CYCLE FOR FLIGHT LEVELS BEGINS$
      FOR G= STOREG(MQ) STEP 1 UNTIL
      STOREGQ(MQ) DO
      BEGIN
        IF NOT FBQ AND MQ EQL 0 THEN
        BEGIN
          TEL = 0$ IF G NEQ GG THEN
          GOTO ENDG$
          N1=  ROW(0)$ ROW(0)=  GRWQ=
          MAXTOW-20000$
REPEAT..
          TEL = TEL +1$ IF G  LSS  1
          THEN
          BEGIN
            G=  GG$ GRWQ=  0$
            GOTO ENDG
          END$
          METPROC(G)$ DESCLIMB(G)$
          IF DECLFUEL(G)  LSS  0 THEN
          BEGIN
            TEL= 0$ G=  G-1$ GOTO REPEAT
          END$
          QQ(-1)$  IF GRWQ EQL 0 THEN GOTO ENDG$
          N2=  ROW(I) - GRWQ$ IF N2  GTR  10
          AND ROW(0)  EQL  MAXTOW THEN GOTO
          ENDG$
          ROW(0)=  ROW(0) +  (IF TEL  GTR  10
          THEN N2/2 ELSE N2)$
          GRWQ=  ROW(0)$ IF ABS(N2)  LSS  10
          OR TEL  GTR  10 THEN
          BEGIN
            ROW(0)=  N1$ GOTO ITER
          END
          ELSE GOTO REPEAT$
 ITER..
        END
        ELSE
        BEGIN
          METPROC(G)$
          IF MQ EQL 0 OR MQ EQL Q-1 THEN
          BEGIN
            GRWQ= RR$ DESCLIMB(G)
          END$
          IF FBQ THEN QQ(-1) ELSE QQ(1)$
```

```
                        IF GRWQ EQL  0 THEN GOTO ENDG
                      END$
                      IF QUANT LEQ QUANTQ(J) THEN
                      BEGIN
                        ROW(J)= GRWQ$ QUANTQ(J)= QUANT$
                        ROWQ(J)= 1$ IF BI AND NOT BG
                        THEN STOREG(MQ)= G$
                        IF ITERATION THEN G=  GG
                      END$
ENDG..
                    END$  COMMENT FLIGHT-LEVEL CYCLE$
CI..
                  END$  COMMENT ALONG-NEXT-MERIDIAN CYCLE$

                END$  COMMENT ALONG-CURRENT-MERIDIAN CYCLE$

                IF BI AND NOT BG THEN
                  STOREGQ(MQ) = STOREG(MQ)
              END$  COMMENT ZONE CYCLE$


                IF ROW(0) EQL 0 THEN GOTO EIND$
                IF BI THEN GOTO EIND$  J=IF FBQ THEN F(Q) ELSE F(0)$
              FOR MQ= B3 STEP -B2 UNTIL B1 DO
              BEGIN
                K=  IF FBQ THEN MQ ELSE MQ+1$ I= ROWQ(J)$
                STOREIQ(MQ +(IF B3 EQL 0 THEN 1 ELSE 0))= I-F(K-1)+
                STOREI(K)-1$ J= I$
              END$
              FOR MQ= 0 STEP 1 UNTIL Q DO STOREI(MQ)= STOREIQ(MQ)$
EIND..
              TOWQ= ROW(0)$ M1=  IF FBQ THEN QUANTQ(F(Q)) ELSE
              QUANTQ(0)$
              LW= ROW(F(Q))$ IF ED THEN
              BEGIN
                  WRITE(' ')$
                  WRITE(F05, TOWQ-LW)$
                  WRITE(F06, IF FB EQL 1 THEN M1/100 ELSE
                ((TOWQ-LW)*15+1083.3*FLTIME/600)/100)$
              END$
            END$  COMMENT SPACEOPT$

            PROCEDURE FF(UU,VV)$   INTEGER ARRAY UU,VV$
            BEGIN
              F(-1)= -1$ FOR K= 0 STEP 1 UNTIL Q DO
              F(K)= F(K-1)+VV(K)-UU(K)+1
            END$  COMMENT FF$

            PROCEDURE PROCES(S1,S2,S3,PR,FFBB,BB,FACTORISATION)$
            INTEGER S1,S2,S3,FFBB$ REAL PR$
            BOOLEAN BB,FACTORISATION$
            COMMENT S1 AND S2 DETERMINE THE LIMITS OF FLIGHT
            LEVELS.
            S3 IS A DUMMY,EXCEPT WHEN
```

```
      FACTORISATION IS TRUE,
      THEN S3 IS A DATUM FOR THE FLIGHT
      LEVEL
      IN WHICH THE M,F,P, IS COMPUTED,
      PR IS PERCENTAGE TRIP FUEL
      FFBB DETERMINES WETHER COSTS(+1),
      FUEL(0) OR TIME(-1) IS OPTIMIZED,
      BB..USE LANDING WEIGHT(TRUE),USE
      TAKE OFF WEIGHT(FALSE)
      FACTORISATION(TRUE) INITIALISES
      PROCES IN THE HORIZONTAL ON THE
      BASIS OF FLIGHT TIME,
      FOLLOWED BY COMPILATION OF FLIGHT
      PLAN IN THE VERTICAL ON THE BASIS
      OF COSTS,FUEL OR TIME$
      BEGIN
         INTEGER NN$
         LOCAL LABEL ENDPR,ING$

         PROCEDURE AA$
         COMMENT SPECIFICATION OF LIMITS IN HORIZONTAL OF
                 GRAPH POINTS ALONG MERIDIAN$
         BEGIN
         LOCAL LABEL UTRECHT$
            STOREI(0)=STOREIQ(0)=ST - A(G1Q-1)$
            STOREI(Q)= STOREIQ(Q)= ST1-A(G2Q-1)$
            IF ROUTE GEQ 1111 THEN GOTO UTRECHT$
            FOR K= 1 STEP 1 UNTIL Q-1 DO
            BEGIN
               STOREI(K)= 1$ STOREIQ(K)= E(K)
            END$

UTRECHT..   FF(STOREI,STOREIQ)
         END$  COMMENT AA$

         PROCEDURE           AAA(F1,F2)$ INTEGER F1,F2$
         COMMENT SPECIFICATION OF LIMITS IN THE VERTICAL$
         FOR K= 0 STEP 1 UNTIL Q-1 DO
         BEGIN
            STOREG(K) = F1$  STOREGQ(K) = F2$
         END$  COMMENT AAA$

         PROCEDURE EP$
         IF TOWQ GEQ MAXTOW OR TOWQ EQL 0 THEN
         BEGIN
              WRITE('SUPER CRITICAL')$
              GOTO ENDPR$
         END$  COMMENT EP$

         INTEGER PROCEDURE SS$
         SS= GRW+RESERVE+PR*((GRW+RESERVE)*
         (MINTIME*0.08-0.15)-500*MINTIME+7300+TAXI)$

         INTEGER PROCEDURE SSS$
         SSS= GRW+RESERVE+PR*(TOWQ-LW+TAXI)$

         INTEGER PROCEDURE SSSS$
         SSSS= TOWQ*(GRW+RESERVE+PR*TAXI)/((1+PR)*NN-PR*TOWQ)$

         PROCEDURE NNQ(A)$ VALUE A$ INTEGER A$
         NN= IF NOT BB THEN TOW ELSE A$

         PROCEDURE TW$
         COMMENT SAFEGUARDING AGAINST OVERLOADINGS$
         IF TOWQ GEQ MAXTOW OR TOWQ EQL  0 THEN
         BEGIN
            WRITE(PEJT)$  WRITE('DECR LANDING W')$
            GOTO ENDPR
         END$


         COMMENT IF FACTORIZATION IS TRUE, THEN THE OPTIMIZATION
                 TAKES PLACE FIRST IN THE HORIZONTAL AND THEN
                 IS FOLLOWED BY IN THE VERTICALS
         IF FACTORISATION THEN
         BEGIN
            AA$ AAA(S3,S3)$ NNQ(SS)$ ROUTE= 1111$
            SPACEOPT(IF BB THEN FALSE ELSE TRUE,-1,FALSE,
            TRUE,NN)$ TW$
            AAA(S1,S2)$ GOTO ING
         END
         ELSE
         BEGIN
            AA$ AAA(S1,S2)$ NNQ(SS)$
            SPACEOPT(IF BB THEN FALSE ELSE TRUE,FFBB,FALSE,
            FALSE,NN)$ TW$
ING..
            EP$ NNQ(SSS)$
            SPACEOPT (IF BB THEN FALSE ELSE TRUE,FFBB, TRUE,
            FALSE,NN)$ TW$
            EP$ NNQ(SSSS)$
            SPACEOPT(TRUE,FFBB,TRUE,TRUE,NN)$ EP$
            IF FACTORISATION THEN ROUTE=  1000
         END$
ENDPR..
       END$  COMMENT PROCESS$

       IF ROUTE GEQ 1111 THEN
       BEGIN
          READ(PCF('NASAD'),  ST, ST1, DONE, ERR)$
          G1Q= ZONEI(ST)$
          FOR K= 1 STEP 1 UNTIL Q-1 DO
          BEGIN READ(PCF('NASAD'), STOREI(K), DONE, ERR)$
                WRITE(F02, 1, STOREI(K))$
          STOREI(K)= STOREIQ(K)= STOREI(K)-A(G1Q+IO*K-1)$
          END$
          G2Q = ZONEI(ST1)$
          WRITE(F02, 2, ST,ST1)$
          ENDS
          GEOMGRID(ST,ST1)$ MINTIME=
          GEODIST(E3,(LA1+LA2)/2)/465$ CTQ$
          E(0)= E(Q)= 1$ FOR K= 1 STEP 1 UNTIL Q-1 DO
          E(K)= A(G1Q+IO*K)-A(G1Q+IO*K-1)$

          COMMENT
          1. WHEN A FLIGHT PLAN FOR THE OPTIMAL COST TRACK IS NEEDED
          CALL 'PROCES(1, 3, 1, 0.03, 1, TRUE, FALSE)'
          2. WHEN A FLIGHT PLAN FOR THE OPTIMAL FUEL TRACK NEEDED
          CALL 'PROCES(1, 3, 1, 0.03, 0, TRUE, FALSE)'
          3. WHEN A FLIGHT PLAN FOR THE LEAST TIME TRACK NEEDED
          CALL 'PROCES(1, 3, 1, 0.03,-1, TRUE, FALSE)'$

          PROCES(1, 3, 1, 0.03,-1, TRUE, FALSE)$
          PROCES(1, 3, 1, 0.03, 0, TRUE, FALSE)$

          GOTO RUN
        ENDS
 DONE.. ENDS
        WRITE(' ')$  WRITE(' ')$
        WRITE('     *** END OF RUN ***')$
        GO TO EOP$
 ERR..
        WRITE('     *** INPUT DATA ERROR ***')$
 EOP..
```

```
 ELT NWS,1,810227,039147                                      ΔC*+VD<:ΔCΔHΔ ΔX
**********<FGNA%23 *KWBC %160000<<@%¤\.079213@¤\0319310303231036403203751330394 8
3@¤\131931027343103842320465231054460@¤\232029025363004244300545330065443@¤\3321
27035372705245280675428081406@¤\431731024323002842300235329018498@¤\531830027333
0033422903453280384 78@¤\632028034352805143280615327071461@¤\732227053362707845 27
0935527106426@¤\.115213@¤\032325050382506547260785626088378@¤\13252405340240684 8
240765624079362@¤\232723056422307050230775723078352@¤\332922061442207651220825 72
2081356@¤\432526068392609047270995627107371@¤\532826067422608149260845526084331@
¤\633125064442507350250745425070326@¤\733224065452407051240705524066315@¤\.15121
3@¤\033222068452208253220885722084342@¤\133422072472208554220905622084348@¤\2336
210734822084542208755220793 39@¤\333721072492207954220795522071338@¤\43342306346 2
306651230645423060310@¤\533522058472306151230595423054307@¤\633722054482305551 23
0535423049308@¤\733921050492204952230465323042292@¤\.187213@¤\03392207150220765 4
230725423063331@¤\134122067512307254230665423056311@¤\23442305852230645324059532
4050303@¤\334723045542405153240475224043287@¤\434022046502204452230395224035298@
¤\534222042512304151240365225033281@¤\634523035522403652250335126032271@¤\734723
0285325030522603051270312 70@¤\.223213@¤\035024033562403854250375225035286@¤\\
4352240285825031552503052250292 86@¤\23532502558250255525024522502328 6@¤\33532401
757240165524016522401627 5@¤\43502402355260255327027512702927 9@¤\53512502256270 24
54270255128026280@¤\635225020572702154280215128021280@¤\73532601457280145428014 5
229014285@¤\.259213@¤\035320009572000755200085321009283@¤\135314012581401256140 1
053150072 89@¤\235213018591302057130155413010292@¤\.080213@¤\03163001730290294129
0325328031504@¤\131828024322703942270445326047487@¤\232027041342606043260695426 0
764620@¤\332227060362608445260965526106438@¤\431627016302703140270365327040504@¤\
5318260333226051412605754260594 88@¤\632026046342606643260765526082451@¤\73222705
636260784526093552610343 4@¤\.116213@¤\032527070392709347271045627113379@¤\132927
0684227081492708755270933 36@¤\233127063442706850270705527072311@¤\333227054442 80
56502805755270582 98@¤\432428060392707947270955627104375@¤\53272905841290684928 07
```

```
83261  82860  82857  81755  83951  84749  85253  87254  88357  88655  88555  88555
82860  82958  83655  84252  85450  86450  87551  88653  89254  89754  89854  90155
82757  84056  85153  86451  87850  88949  89850  90451  90752  90952  90851  90850
84355  85453  86851  88549  90248  91249  91749  91650  91450  91350  91450  91650
84651  86451  88149  89847  91547  92547  92848  92349  91950  91650  91750  91650
86049  86850  88547  90546  92045  92945  93146  92648  92149  91849  92049  92050
85450  86549  88446  90545  92145  93045  93145  92846  92446  92146  92246  92145
85049  85647  86945  88844  90945  92444  92945  92945  92745  92545  92645  92345
85352  85048  85646  87045  89344  91344  92444  92945  93245  93245  93145  93345
84952  85049  85146  86044  87944  89745  91345  92345  93045  93345  93045  93045
87254  86050  85646  85944  87144  88544  90145  91445  92345  92645  92345  92145
88152  87351  87048  86746  86845  87645  89146  90546  91546  91646  91645  91645
89352  88752  88351  87949  87748  87948  88949  90149  90848  91048  91148  91048
90152  89853  89453  89252  89150  89050  89650  90550  90850  90950  91050  91050
90851  90451  90251  90051  90050  90050  90650  91150  91150  91350  91250  91150
90451  90451  90851  90350  90348  90250  91750  92150  91650  92150  91750  91750
```

```
 92950  93450  93554  93955  95155  95252  96954  97257  98758  98657 101154 101660
 93750  94351  94953  95855  97155  97955  99157  99559  99959 100159 101258 101760
 95150  95551  96452  98055  99656 100558 101559 101860 101561 101561 101960 102460
 95949  96651  97953  99955 101756 103158 103660 103561 102961 102661 102860 103559
 95950  97351  99054 101155 102955 104257 104657 104357 103758 103359 103259 103359
 96749  97950  99753 101754 103555 104655 105055 104855 104456 103857 103458 103158
 96950  98049  99553 101454 103256 104655 105054 105054 104854 104255 103755 103055
 97450  97648  98751 100553 102555 104055 104754 105054 105054 104655 104455 104155
 98048  97747  98048  99250 100854 102755 104154 104654 104954 104954 104954 104655
 98049  97947  97845  98646  99850 101153 102454 103354 104254 104854 105253 105454
 98952  98449  98045  98144  98847  99850 100753 101953 103554 104654 105153 105453
 99455  99153  98648  98546  98747  99049  99751 101152 102854 104054 104754 105154
101362 100358  99653  99851  99649  99450  99751 100652 101954 103254 104354 105454
100962 101060 100858 100855 100854 100654 100554 100954 101854 102854 103854 104754
101659 101660 101458 101356 101356 101456 101656 101955 102654 103054 103654 104554
101860 101860 101755 101755 101755 101756 102555 102654 103454 103454 103354 103754
```

```
108055 108055 108155 108855 108955 110552 109952 112252 112552 113855 114358 115560
108955 108854 109055 109856 110758 111657 112458 113458 113758 114157 114858 115360
109351 109652 110255 111458 112662 113563 114563 115163 115163 115261 115460 115661
110147 110651 111655 113160 114465 115768 116667 116867 116667 116464 116263 116260
110651 111453 112757 114763 116268 117469 118068 117868 117668 117365 116765 116065
111351 112153 113658 115764 117269 118369 118768 118467 118167 117766 117166 116465
111646 112751 114257 116164 117667 118668 118867 118666 118666 118365 117865 117365
111945 113048 114253 116059 117463 118465 118865 118965 118965 118965 118665 118365
113044 113146 113450 114953 116357 117458 118260 118862 119064 119263 119164 118965
113344 113044 113046 114048 115150 116251 117255 118357 119058 119459 119562 119765
113248 113247 113045 113445 114145 114746 115849 117252 118355 119157 119561 120063
114151 113950 113648 113746 114044 114043 114944 116247 117550 118555 119259 119961
114753 114751 114550 114448 114446 114444 114643 115444 116748 117953 119057 120159
115755 115453 115253 115150 115149 115147 115144 115444 116747 117851 116955 120061
116255 116155 115854 115652 115951 116049 116047 116447 117249 118053 118757 119565
116955 117055 116755 116053 116852 117049 116648 117250 118251 118258 118359 118666
```

```
83261  83162  83261  82857  83357  84753  86050  85250  85350  87250  88650  90150
82559  83558  84458  85056  85454  86351  87550  87750  87850  88650  89750  90350
83356  84256  85555  87052  87948  88647  89549  90050  90251  90550  91150  91950
84152  85053  86652  88649  90046  91046  91648  91950  91851  91650  91750  91650
85150  86151  87750  89648  91245  92245  92548  92750  92451  92150  91750  91550
85249  86050  88250  90048  91645  92645  92947  92949  92650  92250  91750  91551
85050  86350  87950  89648  91245  92544  92944  93046  92847  92348  91950  91551
84847  85548  86448  87946  89944  91643  92543  92944  92945  92647  92348  92148
85253  85450  85846  86746  88444  90143  91442  92343  92645  92746  92946  93246
85653  86051  85949  86247  87346  88745  90044  91144  91944  92545  93044  93243
86956  86954  86852  86649  87048  87947  89046  90045  91045  91945  92845  93445
87356  87855  87954  87652  87650  88150  88749  89348  90248  91247  92246  92945
88753  88754  88955  89054  89051  89150  89250  89350  89950  90849  91748  92445
89852  89652  89853  90252  90351  90350  90250  90350  90650  91150  91550  91548
91246  90948  90950  91250  91450  91550  91250  91350  91450  91550  91550  91650
91749  91650  91950  92250  92550  92450  91750  91750  91750  91750  91650  91650
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 93855  93255  92955  93455  94455  96355  96655  98155  97759  99260 100261 101362
 94355  94455  94655  95455  96655  97755  98455  99556 100259 100860 101361 101762
 95355  95855  96655  97755  99255 100255 101055 101957 102659 102860 102760 102862
 96957  97356  98456  99956 101656 102557 103156 103658 104059 103960 103660 102661
 96554  97855  99456 101157 102957 103957 104456 104656 104657 104258 103560 102760
 96953  98054  99655 101656 103457 104457 104855 104755 104656 104058 103460 102961
 97048  98153  99655 101455 103356 104456 104955 104955 104755 104156 103859 103562
 98049  98151  98952 100453 102254 103756 104556 104855 104854 104655 104357 104058
 98251  97849  98248  99248 100850 102453 103655 104254 104653 104754 104955 105355
 97353  97749  97946  98644  99746 101049 102152 103152 103953 104554 104954 105352
 98255  98452  98548  98746  99345 100047 100849 101750 102851 103953 104754 105355
 99663  99558  99654  99650  99748  99848 100449 101150 101851 102952 104154 104755
100065 100562 100859 100856 100654 100552 100951 101352 101852 102653 103754 104455
101761 101662 101761 101860 101657 101755 101954 102054 102454 102955 103555 104055
102660 102661 102561 102461 102361 102757 102955 102855 102955 103155 103355 103255
103560 103561 103459 102659 102760 103458 103755 103655 103455 103355 103355 103255
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
107655 107555 108055 108354 109053 110857 110457 112059 112352 113456 113757 114958
108055 108655 109355 110157 111059 112060 112660 113561 114060 114860 115260 116161
108655 109755 110657 111859 112962 113965 114866 115666 116167 116365 116364 116165
110755 111355 112356 113662 114966 115968 116869 117569 117669 117568 117167 116562
111655 112356 113659 115064 116369 117269 117868 118367 118566 118267 117667 117164
111650 112754 114257 115762 117166 118067 118566 118764 118763 118464 118065 118065
111744 112649 113852 115557 117161 118263 118863 119061 118860 118461 118063 117764
111342 112245 113146 114651 116255 117657 118659 118960 118960 118560 118360 118360
111544 112244 112743 113845 115247 116750 118054 118657 118959 118861 118761 118460
112245 112545 112744 113443 114443 115645 117048 117852 118455 118859 119061 119262
112945 113045 113145 113644 114244 115044 116144 117046 117751 118456 118860 119263
113750 113648 113947 114346 114545 114944 115443 116144 116948 117954 118658 119060
114455 114453 114852 115150 115247 115546 115945 116145 116647 117453 118357 118759
115955 115755 115854 115953 115950 116147 116346 116547 116849 117452 118154 118454
116455 116655 116455 116455 116353 116450 116649 117150 117453 117854 117955 118055
117455 117255 116755 116755 116455 116455 116452 118055 118055 118055 118055 118055
```

```
  941   931   919   908   896   884   872   858   845   830   815   800   784   767   749   730   711   691   670   648
 1088  1070  1052  1033  1013   992   970   946   921   895   868   839   809   777   744   710   674    -1    -1    -1
 1237  1207  1174  1139  1102  1063  1021   977   930   881   830    -1    -1    -1    -1    -1    -1    -1    -1    -1

*00000*00000*00000*00000*00000141000135800
131800129300126800124800122800120700116900
108200106300104300102600101000 99100 95600

   63    67    71    84    81    86    91    96   101   107   113   119   127   135   144   154   167   181
   75    81    87    94   100   107   113   120   128   136   147   160   177   198   226    -1    -1    -1
   96   104   113   121   131   143   158   179   209   250    -1    -1    -1    -1    -1    -1    -1    -1

  186   198   210   238   235   249   263   277   292   307   324   342   362   384   408   435   466   501
  211   227   245   266   286   306   325   343   362   382   406   435   473   524   590    -1    -1    -1
  263   284   307   330   355   383   418   466   532   624    -1    -1    -1    -1    -1    -1    -1    -1
```

```
2270  2400  2540  2690  2830  2980  3140  3300  3470  3650  3840  4040  4250  4480  4730  5010  5310  5640
2470  2630  2800  2980  3160  3340  3520  3710  3910  4130  4380  4680  5050  5440  5950    -1    -1    -1
2750  2940  3140  3350  3570  3800  4090  4450  4930  5570    -1    -1    -1    -1    -1    -1    -1    -1

  45    46    47    48    50    53    56    59    62
 129   134   140   145   150   158   165   175   185
 205   213   222   236   250   267   285   307   330
 273   287   300   325   350   390   430   475   520
 335   352   375   412   450   512   585   675   820
 387   407   443   493   550   640   767   950  1295
 425   453   504   568   650   782   985  1300    -1
 454   490   556   640   750   933  1245    -1    -1
```

```
     400   470   480   490   500   515   530   545   560
    1430  1445  1460  1480  1500  1550  1600  1645  1690
    2220  2280  2340  2420  2500  2620  2740  2870  3000
    2950  3050  3160  3350  3500  3700  3920  4200  4470
    3660  3850  4050  4280  4500  4850  5150  5670  6320
    4370  4590  4880  5150  5500  6000  6670  7530  8520
    5050  5260  5630  5950  6500  7200  8500  9900 12150
    5630  5900  6280  6720  7500  8700 11050    -1    -1

       0     0     0     0     0     0     0     0     0
      36    40    43    47    50    54    59    68    75
      73    79    84    93   100   110   123   141   157
     105   113   123   136   150   168   190   222   268
     131   144   160   179   200   230   260   322   460
     150   168   191   218   250   296   334   440    -1
     161   187   219   254   300   372   417    -1    -1

   *** INPUT PARAMETERS ***

          1   1500  75000  10000      -1 142900   93000   51281          1      0     111

          0    115
 E=
  -5.6346,-01 -8.2619,-01
 MINTIME
   6.5615,+00
 E=
  -9.3467,-01 -3.5552,-01
 DIST=
   1.7445,+02  1.7445,+02
 ENDD
        13305
 TER
```

West bound

ORIGINAL PAGE IS
OF POOR QUALITY

GENERAL.

This program which was written for experiments of three dimensional flight planning for DC8-D3 aircrafts over the North Atlantic has been converted for use in UNIVAC 1108 in ALGOL 60 and will be enhanced to be appropriate for flight planning for DC10 and B747 aircrafts.

Reference is made to pages and formulas in 'Optimal Track Selection and 3-Dimensional Flight Planning' written by H.M. DE JONG, KNMI No.93.

INPUT/OUTPUT DATA DESCRIPTION.

INPUT

METEOROLOGICAL DATA.

There are six data tables for compressed meteorological data (geopotential height(in gpdam)and temperature(centigrade degree)).

See p.126, Fig. 39.

Geopotential values and temperature are depicted from the grid points in a Cartesian grid superposed on a polar stereographic chart projection with standard parallel at 60 degree N.

This grid is part of the well-known octagonal grid prepared by the National Meteorological Center.

The y-axis runs parallel with the 80 degree W meridian.

The area covered is rectangle comprising 12 x 16 grid points.

See output listing.

Those six tables show meteorological data (each table 12 x 16) for 200, 250, and 300 mbar for two validity times 12 hours apart.

PERFORMANCE DATA

See output listing.

TABLE 1 shows specific range for flight levels 310, 350, and 390 (see p.116, table 1). This is an important economic index which is the air distance covered by a turbo-jet aircraft per unit of fuel consumption.

This datum is almost independent of temperature and depends on the aircraft weight only at a prefixed flight level.

TABLE 2 shows maximum weight(kg) per flight levels 310, 350 and 390 (see p. 122, table 9). The maximally allowable weight depends on flight altitude and temperature deviation from standard.

TABLE 3 shows climb distance (nautical mile) as a function of weight (kg) and flight levels 310, 350, and 390.

See P.118, table 5.

TABLE 4 shows climb time (1/1000 hrs) as a function of weight (kg) and flight levels. See p.117, table 3

TABLE 5 shows climb fuel (kg) as a function of weight and flight levels.

See p.119, table 7.

TABLE 6 shows climb time temperature correction as temperature deviation and climb time (1/1000 hrs) at standard temperature.

See p. 117, table 4.

TABLE 7 shows climb fuel temperature correction (kg). See p.119, table 8.

TABLE 8 shows climb distance temperature correction (n.m).

See p.118, table 6

INITIALIZING INPUT PARAMETERS:

FLUR = sequense number.

TAXI = taxi fuel (kg).

GRW = zero fuel weight (kg).

RESERVE = reserve fuel (kg).

TOW = -1 computaion starting in end-point of flight.

MAXTOW = maximum take-off weight.

MAXLW = maximum landing weight.

DATE = date/month/year.

IO = -1; east bound flight, 1; west bound flight.

TTT = time of departure.

ROUTE = route indicator.

ST = starting graph point number.

ST1 = destination graph point number.

Note here that all input data are read from card images of PCF file.

PCF file name is NASAD.

OUTPUT

In the flight plan (see output listing),

NO. = graph point number.

HEAD = heading in flight.

FL = flight level in 100 ft.

TMP = off standard temperature in centigrade degrees.

TAS = true air speed.

WIND = tail (head) wind along the track.

DIST = distance of flight segment.

ACCD = accumulated distance flown(n.m).

TIME = time along segment flown.

ACCT = accumulated time flown (hr, min).

BURN = fuel consumed in segment (kg).

WEIGHT = weight (kg).

TOC = top of climb.

TOD = top of descent.

EXTRA COMMENTS FOR THE PROGRAM.

46 - 49

Procedure INTERPOL computes a linear interpolation.

51 - 69

Procedure SUM1 and SUM2 compute summations.

These procedures are used for computing average values in procedure METPROC.

72 - 73

Procedure WW is a write-procedure.

76 - 77

p1, p2, and p3 are flight levels.

q1, q2, and q3 are pressure altitude for standard pressure levels 300, 250, and 200 mbar.

78 - 84

reads in compressed values for geopotential height and temperature in 200, 250, and 300 mbar, valid for two standard times, 12 hours apart and fills array HT with those data.

85 - 90

writes out the inputed data for checking purposes.

92 - 113

produces the meteorological paramrters (compressed geopotential and temperature data, which apply in grid points at flight levels 310, 350 and 390. Therefore the inputed data for the standard pressure altitude are destroyed

115 - 118

The graph points are labelled by numbers running from 0 to 115. (note that the graph point number shown in Fig. 37, F.110 runs from 1 to 116).

The graph is such that the point sets consist of subsets of points arranged along 'meridians' (index running from 0 to 15).

Array A(index) contains the graph point number assigned to the most northerly point on each meridian.

Array A is used for the computation of zone index (procedure ZONEI).

119 - 133

Array V has values assigned for geographic coordinates for graph points and check points on continents : latitude (4 digits) and longitude (4 digits) in degrees and hundredths. See Fig. 37, P.110.

135 - 142

fills Boolean array BK and BKQ with 'TRUE' ; (initialization).

143 - 156

denotes blocking in continental airways (connectivity in the graph). Boolean array entries of BK and BKQ indicate which graph points are connected or blocked (TRUE=blocked and FALSE=connected).

Array BK is used for the zone index < 3 and array BKQ is used for the zone index > 8.

BK(a,b,c)=FALSE means point'b' on meridian 'a' is connected with point 'c' on meridian 'a+1'.

Here point 'b' denotes the 'b' th point from the bottom on the meridian. Example: see Fig. 37. BK(2,2,2) = FALSE means graph point number 3 on meridian 2 is connected with graph point number 5 on meridian 3

163 - 172

Procedure ZONEI determines the index of the zone associated with a graph point number. Parameter Q = graph point number.

For example, a graph point number 49 gives ZONEI(49)=6.

174 - 183

Procedure DT dissects latitude and longitude from the compressed coordinates

LALAT means latitude and LALONG means longitude.

Parameter K is an index of array V.

185 - 216

Procedure LIS determines actual latitude and longitude value for a graph point. (see procedure DT).

The sign of longitude is changed in east of Greenwhich.

Parameter U=graph point number.

218 - 237

Procedure CTQ prepares a time instant array DD, zonally specified, to be used for the time interpolation later on.

A=0.4 denotes an estimated flight time of 24 min. per continental zone.

There are 15 instant values (index 0 - index 14).

239 - 256

Procedure HH computes grid point values for geopotential or temperature (CHOICE=TRUE computes geopotential value, while CHOICE=FALSE computes temperature value) in flight level index F (index 1 denotes 31000 ft and indices 2 and 3 denote 35000 and 39000 ft respectively).

Parameters V and W denote index numbers of array HT (meteorological).

Interpolation takes place according to zone-assigned time instants DD( ).

258 - 275

Procedure GEOP computes geopotential (when CHOICE=TRUE) or temperature (when CHOICE=FALSE) value in arbitrary point, using adjacent grid point values from array HT (see procedure HH) and bi-linear interpolation.

Parameters V and W will be modified to denote index numbers of array HT. Parameter C = flight level index.

277 - 284

Procedure XYTRANSF converts the geographical coordinates (LAT,LON) of a point on earth into the Cartisan coordinates (Y,X).

See P. 62-63.

286 - 289

Procedure GEODIST computes distance on the sphere from Euclidian distance DDD given in units of the rectangular grid on the map.

LAT denotes latitude.

291 - 301

Procedure GEOMGRID computes Euclidian distance and unit vector pointing along an arc in the graph between two arbitrary graph points with respect to rectangular grid system.

FP and GG denote graph point numbers of end points.

E3 denotes Eucledean distance.

E1 and E2 denote unit vectors for x,y components respectively.

303 - 363

Procedure PART2GEOM determines length of segment by summation of contributions from subsegments (LENGTH(I)).

These subsegments are identified by intersection of an arc with coordinate lines in Euclidean grid.

Normalized coordinates (running from 0 to 1) are stored in array AB.

These values serve also as weighting factors later on.

D denotes distance computed for averaged degree of latitude on the sphere. Array LENGTH(I) contains lengths of subsegments on the sphere. AID2 denotes interpolated degree of latitude.

See P. 84-85.

365 - 410

Procedure METPROC processes meteorological parameters.

Parameter S=flight level index (1,2 or 3).

DETAILS

370 - 371

computes geopotential value at end-points of a flight segment.

372

computes drift angle by using Bellamy's formula (see P. 78).

376 - 381

estimates coordinates in mid point P of segment. P=(Xster, Yster).

384

estimates temperature in mid point P at flight level S.

Temperature has minus sign.

387 - 392

wind components, geostrophic wind. Formulas shown in P.69-70 is not used (reason: not to smooth too drastically from geopotential fields, and to economize in speed of computation by computer).

See formulas 5.22 and 5.23 in p.66.

Array WNDX and WNDY contain x-component and y-component of wind resp.

393

computes true air speed (TAS) for the specific type of aircraft.

For values of constants see Smithsonian tables, Meteorological text books or ICAO Standard Atmosphere.

394 - 395

computes flight time components using vector sum of wind and TAS, and segment length (LENGTH (I))

398 - 409

determines average values for meteorological and other parameters along segment under consideration using previously calculated contributions.

TEM=temperature. TDEV(S)=temperature deviation.

TIME(S)=flight time in flight level index S.

AIRDIST(S)=air distance.

LE=distance of flight segment.

WIX(S) and WIY(S) = wind components.

TAX(S) and TAY(S) = true air speed components.

413 - 429

Procedure TABLE and TABLEQ are for table look-up procedure for performance data. See P.116-119 and output listings.

461 - 497

Procedure READQ reads in performance data and procedure WRITEQ directly writes out those inputed data for checking purpose.

499 - 500

coefficients used in algorithms.

502 - 507

reads in and writes out initializing input parameters for flight plan.

508 - 520

In case of prescribed track flight, the routine reads in Q (number of graph points on the prescribed route).

In another case, it reads in ST(starting point number) and ST1(ending point number) and decides Q value.

In the latter, Q=(number of zones between source and destination) - 1.

529 - 532

Procedure SQ and JQ are conversion formulas for indices used in subsegment algorithms (by using entries of array A(index) or array F(index)).

534 - 555

Procedure DESCLIMB computes performance by using the table look-up procedure in climb(if MQ=0) or descend(if MQ not= 0).

Array DECLTIME gets climb time or descent time.

Array DECLDIST gets climb distance or descent distance.

Array DECLFUEL gets climb fuel or descent fuel.

Array AIRDIST gets climb air distance or descent air distance.

Array TIME gets climb time or descent time.

Note that in the descent zone, performance data table is not used.

So specific expressions by company's planning policy are chosen.

See formula 6.31 in P.120-121.

557 - 559

Procedure ARCTANQ is for computation of heading degree by using arc tangent formula.

561 - 564

Procedure LO is for computation of longitude from Euclidean coordinates in rectangular grid. See P.62, formula 5.17 and 5.18.

Z1 and Z2 denote Euclidean coordinates.

566 - 574

Procedure HEADING computes heading in degree as required for flight planning purposes.

L1 and L2 denote x-component and y-component of true air speed.

576 - 617

procedure LINE writes out optimized flight plan.

See the output listing of the flight plan.

619 - 926

Procedure SPACEOPT is a key procedure for optimalization.

In this procedure crucial is that some parameters may not surpass upper bounds, for example aircraft weight.

This requires the inclusion of several protective statements in order to ensure proper functioning of the operational scheme in-flight.

FBQ determines whether the flight plan computation will be performed backward(false) or forward(true).

FB determines whether cost(+1) or fuel(0) or flight time(-1) will be optimized.

BI determines whether the navigation regime is free in the horizontal (false) or bounded by one point(true).

BG determines whether the cruising altitude is free(false) or bounded (true).

RR is take-off weight or landing weight.

DETAILS:

637 - 645

Procedure PREP computes distance and flight time, and prints out one line of flight plan.

A1=segment distance. B1=segment flight time.

AA=true denotes 'climb' and BB=true means 'descent'.

See the output listing of the flight plan.

647 - 694

Procedure EDITING prints out proper heading lines.

696 - 768

Procedure QQ computes segment contribution, fuel, etc.

When o=-1 computation is for in-flight direction.

When o=+1 computation is for opposite flight direction.

702 - 722

Procedure CL computes the aircraft weight(GRWG) and total fuel, flight time or costs at end-point of a flown segment. (the contribution of fuel in steps included).

Parameter M1 denotes segment fuel(kg) and M2 segment time(min).

In the climb phase no step contribution is taken into account as required(BCLIMB=TRUE)

GG denotes flight level index at the graph point when the computation starts. G denotes the flight level index to be investigated.

Step consumption: UP=100 kg/4000ft

DOWN=-80 kg/4000ft.

(surplus of segment fuel)

ITERATION becomes TRUE when (1) back-tracking (2) first zone is reached (3) TOW is unknown and Wb is known (see P. 120).

In array ROW the aircraft weight is stored.

GRWQ denotes aircraft weight at the end point of segment.

QUANT denotes cost, fuel or time at the end point of flown segment.

729 - 736

computation in climb or descent by procedure CL.

If ED is TRUE then it prints out the reselts.

739 - 741

safeguarding statements.

744 - 752

computes segment contribution of changes in aircraft weight.

Note that fuel consumption is expressed as loss of aircraft weight. The table for specific range (P. 116) is used.

Formula 6.27 (P. 115) requires an estimate of weight halfway a flight segment. For that purpose coefficients as shown in table 2, P. 116 are required (r1, r2).

753

jumping back to label JM1 is for safeguarding that the actually derived aircraft weight does not surpass the maximally allowed weight.

755 - 762

computes the contribution for the last segment.

If ED is TRUE then it prints out the results.

765

GRWQ and QUANT attain absurd values for safeguarding operational performance

771

ED becomes TRUE when all degrees of freedom in the horizontal and the vertical are lost, for example when a call of procedure SPACEOPT takes place in the final mode for flight plan computation along the (optimal) track found in previous calls.
(when navigation regimes is bounded by one point and cruising altitude is bounded).

772 - 797

Initialization and preparation of all parameters needed for the algorithmic process of optimization.
FBQ=TRUE means flight plan computation will be performed forwards, while FBQ=FALSE backwards.
Array F is filled in line 928-932 (see procedure FF).
The process can be activated for arbitray begin- and end-points.
In order to reduce storage space a renumbering is made for all subset points. So array F is defined and functions completely analogous to array A for whole graph point set.
Array STOREI and STOREIQ contain the assigned numbers of labelled end-points on meridians in the subgraph.
Zone cycle runs from line 799 to 904.
The zone cycle repeats as many times as the number of zones.
A cycle for graph points along meridian runs from line 806 to 900.
This cycle repeats as many times as the number of graph points in the current meridian.

A cycle for graph points along next meridian runs from line 811 to 898. This cycle repeats as many times as the number of graph points in the next meridian.

A cycle for flight levels runs from line 842 to 896. This cycle repeats as many times as the number of flight levels used.

810

conversion for indices used in subsequent algorithms.

815 - 835

If the navigation is free (BI=FALSE) in the horizontal, it checks whether a segment (II, JJ) is blocked or not.

BK or BKQ = TRUE means airway is blocked.

If blocked, it skips computation for the segment.

836 - 840

conversion for indices (point G4).

determines geographic elements for a segment between points G3 and G4. See procedures GEOMGRID and PART2GEOM.

846 - 875

In case of backtracking in a flight plan computation an iteration process is put into action in the climb zone.

This is done in order to determine the (unknown) take-off weight by iteration in such a way that the climb parameters do match with the parameters found during backtracking when arriving in the climb zone. The iteration starts with a take-off weight 20000 kg below the (known) maximum take-off weight.

Array ROW contains the (optimal) value of aircraft weight found during optimalization, using a zone-cycle.

The iteration finishes when either the weight difference at the point of matching is less than 10 kg or the number of iteration steps exceeds 10 (poor convergence).

In each step take-off weight is adjusted, see line 865-875.

879 - 887

similar computations but not backtracking case or not take-off weight adjustment case.

888 - 894

These statements are crucial optimalization criteria.

909 - 925

Preparation of elements which are required for a subsequent call for procedure SPACEOPT with lessened number of degrees of freedom including a final call for computation and presentation of flight plan data. In this final stage no degrees of freedom are left. Procedure SPACEOPT then merely operates along a predefined track (the solution of previous calls).

928 - 932

Procedure FF determines a zone index array F for a subset of graph points analogous to array A. (array F is a sub-zone array).

934 - 1035

A call for procedure PROCES results in the production of a flight plan along whatever route is desired.

DETAILS

962 - 973

Specification of limits in horizontal of graph points along meridians. Array STOREI and array STOREIQ store lower limits and upper limits respectively for the number of graph points on each meridian.

If route >= 1111, then production of flight plan along a presribed route

In this case the procedure SPACEOPT works in a degenorate mode.

975 - 980

Specification of limits in the vertical.

Array STOREG and array STOREGQ store lower limits and upper limits respectively for the number of flight levels on each meridian.

982 - 987

Procedure EP indicates that the flight planning computation blows up.

989 - 1000

See P. 123-124 for procedures SS, SSS and SSSS.

These procedures compute landing weights.

NN contains landing weight or take-off weight.

1002 - 1008

Procedure TW is for safeguarding against overloading.

1011 - 1035

activates process as follow:

If FACTORIZAION is TRUE, then the optimalization takes place first in the horizontal and is followed by an optimalizaion in the vertical.

The following steps occur:

(a) optimalization in the horizontal based on time.

(b) optimalization in the vertical, using the track solution found in (a) and based on fuel, time or costs.

(c) computation of flight plan along solution found in (b).

If the optimalization takes place in free space, the sequence of calls results in:

(i) optimalization in space based on time, fuel or costs.

(ii) optimalization in the vertical through horizonal track found in (i), based on time, fuel or costs.

(iii) compilation flight plan along track solution found in (ii).

Note that (ii) could be bypassed as the solution is found already in (i). But (ii) can be generated with a slightly different landing weight. The compilation of a flight plan along a prescribed route passes through all three procedure calls.

This means that in fact the computation is repeated threefold, however with properly tuned landing or take-off weights.

In order to protect against subsequent calls of procedure PROCES, ROUTE is assigned by 1000.

1037 - 1048

Read statement in case that a flight plan compilation is desired along a prescribed track.

This track is specified by graph points indicated by their numbers.

1049 - 1052

Computation of an estimate of flight time to be used for the estimation of other parameters.

The same with distance.

Procedure CTQ makes time instant array DD for each zone(meridian).

Array E is for storing the number of graph points used on a meridian.

1062 - 1065

performs a process for the production of a flight plan.

Actual parameters when calling procedure ;

use landing weight (BB=TRUE).

cost optimalization (FFBB=1).

FFBB=0 means fuel opt, FFBB=-1 means time opt.

regularity percentage (PR=0.03).

fligh levels used between S1=1 for 31000 and S2=3 for 39000.

S3=1; the value is immaterial if FACTORIZATION is FALSE.

For required flight plans ;

1. when a flight plan for the optimal cost track is needed,

call 'proces (1,3,1,0.03,1,TRUE,FALSE)'.

2. when a flight plan for the optimal fuel track is needed,

call 'proces (1,3,1,0.03,0,TRUE,FALSE)'.

3. when a flight plan for the least time track is needed,

call 'proces (1,3,1,0.03,-1,TRUE,FALSE)'.

For the flight plan 1, 2 or 3 above, FACTORIZATION=FALSE.

For other flight planning simulations, FACTORIZATION=TRUE.