User's guide to OSFIC. A computer program for oilspill fires in compartments

Karlsson, Björn

1989

LUND UNIVERSITY · SWEDEN
INSTITUTE OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF FIRE SAFETY ENGINEERING
CODEN: SE-LUTVDG/TVBB-3051
ISSN 0284-933X

BJÖRN KARLSSON

USER'S GUIDE TO OSFIC

A computer program for oilspill fires in compartments

LUND 1989

Table of contents

1.0 Introduction

This document is a description of the computer code OSFIC. The development of the code was sponsored by the Swedish Nuclear Power Inspectorate as an effort to develop useful tools for fire analyses in nuclear power plants. The name OSFIC stands for "OilSpill Fires In Compartments".

The purpose of this paper is to describe the code and to facilitate the use of the computer program. A more detailed description of the model, it's purpose and the physics involved is given in a seperate paper (1).

Chapter 2 gives a short description of the subroutines involved and the program flow. Chapter 3 gives a list of the main variables used in the program. Finally, an example of an interactive run is given in chapter 4.

A printout of the code itself is given in Appendix A. The code of another computer program, HYDRO, is given in Appendix B. This is a very short program, developed to analyse hydrogen leakage scenarios.

2.0 Main features of the program.

2.1 General

OSFIC is written in ANSI '77 FORTRAN, it consists of 1 main program, 11 subroutines and 2 real functions. It has around 1500 lines of code, including comments. The compiled program is 295 kilobytes long when compiled with the RM FORTRAN compiler on an IBM PC computer.

2.2 Main program, subroutines and functions.

The the main program is called OSFIC. It calculates the heat release rate and gastemperatures and controls the calls to the other subroutines. Fig. 1 shows a simplified flow diagram of the program.

The following is a list of the subroutines and functions with short comments on their purpose:

INPUT:
This is the input subroutine of the program. It reads data interactively or from a file specified by the user. It is called once by the main program

VIEWGL:
This routine calculates the geometric view factor from the gaslayer to the pump under consideration. It is called once by the main program.

CLOSED:
This routine calculates the time to smoke filling and the maximum gastemperature in a closed compartment. It is called once for every oilspill size treated by the main program.

GASLAY:
This routine calculates the time to critical surface temperature of the pump and the critical core temperature of the cable as a result of the heat transfer from the gaslayer. It is called once for every oilspill size treated by the main program.

PILEAK:
This routine is used when the oilspill is due to a pipe rupture. It divides the room into a certain number of coordinates, places the oilspills there and calls other subroutines which calculate the heat transfer to the components for every spill

position. This routine is called once for every oilspill size treated by the main program.

VIEW:
This routine provides data to real function CYL allowing configuration factors from the flame to be calculated. It is called by PILEAK for every oilspill position treated.

CYL:
This function calculates the configuration factor from a cylindrical flame to a component. Called by VIEW.

SUTEM:
This routine calculates the time to critical surface temperature of the pump as a result of radiation from a cylindrical flame. Called by PILEAK for every oilspill position considered.

NEWT:
This function solves the heat transfer equation used in routine SUTEM by the Newton-Raphson method. Called by SUTEM.

FLOCAB:
This routine is used when a cable emerging from the floor is present in the room. It calculates the time to critical core temperature of the cable as a result from radiation from the flame. Called by PILEAK.

CABLES:
This routine is used when the cables enter through a wall and are suspended from the ceiling. It checks whether the cables are within the plume radius and consequently decides which mode of heat transfer is assigned to the cables. Called by PILEAK.

DATA:
This routine sorts the data and finds the characteristic time to damage for each oilspill size. Called once by the main program.

OUTPUT:
This is the output routine. It it writes the output to a file specified by the user. Called once by the main program.

3.0 Program variables and defined constants.

3.1 Input variables

A list of the input variables is given in Table 1. The cartesian coordinate system used by the program is: When entering through the door the origin is in the left hand side corner, the x-axes is along the wall with the door. The length of this wall is the breadth of the room. However, if there is a separating wall in the room, splitting the room in two parts, the origin must be in the same part as the oilspill. In the following, when coordinates are required, this refers to the x- and the y-axes coordinates. The z-axes (or the height) is treated separately, by asking questions on the objects height.

The length, breadth and height of the room are required as well as the size of the opening, if there is one. The room is assumed to be of a rectangular shape. A separating wall can be inserted, splitting the room into two parts. It is assumed to confine the oilspills to one part of the room and to protect the components in the other part against radiation from the flames. However, the separating wall does not reach the ceiling and these components are therefore subjected to radiation from the hot layer. Two wall coordinates are required, the separating wall is assumed to be at right angles to the walls of the room.

The cable can enter the room from the floor or through the wall. In the former case the coordinates of the floor cable are required, in the latter the hight at which the cable enters and three corner coordinates of the cable are required. These three coordinates are assumed to be the point at which the cable enters the room, the point at which it turns towards the pump and the point where it enters the pump. The cable is assumed to be a straight line at right angles to the walls of the room between these three coordinates. The core temperature at which the cable stops functioning is also required.

Two corner coordinates of the pump are required, the first is the point nearest to the origin and the second is the opposite corner coordinate, i.e. the point farthest from the origin. The pump is assumed to be at right angles to the walls of the room. The height of the pump from the floor is also required. The total amount of oil contained in the pump is assumed to be the maximum oilspill, this is entered in liters. The critical pump surface temperature is entered in degrees Celcius.

The room is divided into a net of coordinates for the oilspill positions. The differential size of this net is required input.

3.2 Constants defined in the program.

Table 2 gives a list of the constants defined in the DATA-statements in the program. These values can be changed but then the code needs to be re-compiled. The first five values in Table 2 refer to hydrocarbon transformer oil. The data is from reference [2], except for the oilspill thickness. The thickness of the oilspill is difficult to predict and should really be treated in a probabilistic way since it is directly proportional to the area of the oilspill, thereby the energy output and the time to burn-out.

The 6 next entries in Table 2 refer to the cable. Here only one type of cable is treated, i.e. a power cable named AKKJ 3*400 + 50, produced by ASEA KABEL AB. This cable can easily be replaced, by a smaller or larger cable of a similar type, by changing the radius of the core and the radius of the cable.

The walls of the compartment are assumed to be of concrete, hence the value of thermal inertia for the walls. The thermal inertia of the pump refers to that of steel.

3.3 Output variables

Table 3 gives a list of the output variables. Some of these variables are only dependent on the oilspill size, other variables are dependent on both oilspill size and oilspill position. For a further definition of these variables, especially the time related ones, see (1).

4 An example of a run

The following two pages are an example of a session at the terminal. The **bold** letters represent the user response, all other text is generated by the program. The user is required to press "carrage return" after each line of input, this user response is not indicated in the example.

The numerical input can be entered in exponential, floting point or integer form. Sometimes when numerical input is required the program asks for this to be separated by a comma. However, each entry can be separated by a comma, a "carrage return" or by pressing the space bar. In the following example the first numerical input is seperated by a comma and at the end of this line the "carrage return" is pressed.

The example here refers to a compartment in Barsebäck Nuclear Power Station, a more detailed description of the room is given in the main report (1).

```
C:\> OSFIC

WELCOME TO THE INPUT PORTION OF THE PROGRAM.  THIS SECTION ALLOWS
YOU TO INITIALIZE YOUR VARIABLES EITHER INTERACTIVELY OR BY A
PREEXISTING INPUT FILE.  IT THEN GIVES YOU THE OPTION TO SAVE THE
INPUT DATA IN A FILE (NAMED BY YOU). ALL LENGTHS AND COORDINATES
ARE IN METERS.
THE CARTESIAN COORDINATE SYSTEM USED IS: WHEN ENTERING THROUGH
THE DOOR, THE ORIGIN IS IN THE CORNER ON YOUR LEFT, THE X-AXES IS
ALONG THE WALL WITH THE DOOR. THE LENGTH OF THIS WALL IS THE
BREADTH OF THE ROOM.
HOWEVER, IF A WALL SPLITS THE ROOM IN TWO PARTS, THE ORIGIN MUST
BE IN THE SAME PART AS THE OILSPILL.


WOULD YOU LIKE ONE OF YOUR  INPUT DATA FILES TO BE READ IN AND
INITIALIZE THE INPUT VARIABLES? (Y/N):

N

TYPE IN ROOM LENGTH,BREADTH AND HEIGHT (IN METERS) SEPERATED BY A
COMMA.

22,8,6

IS THE DOOR TO THE COMPARTMENT OPEN ? (Y/N):

Y

TYPE IN THE HEIGHT AND THE BREADTH OF THE DOOR, SEPERATED BY A
COMMA.

2,1

ENTER THE CRITICAL SURFACE TEMPERATURE OF THE CABLE (CELCIUS):

150
```

```
ENTER THE CRITICAL SURFACE TEMPERATURE OF THE PUMP (CELCIUS):

200

ENTER THE MAXIMUM AMOUNT OF OIL (LITERS) THAT CAN BE SPILLED
IN THIS ROOM:

10

DOES A WALL SPLIT THE ROOM IN TWO PARTS ? (Y/N):

Y

INPUT THE X AND Y COORDINATES OF THE WALL (X1,Y1,X2,Y2):

0,10
8,10

DO THE CABLES ENTER THE ROOM THROUGH A WALL OR FROM THE FLOOR ?
(W/F):

W

INPUT THREE COORDINATES OF CABLE POSITION. FIRST COORDINATE IS
WHERE CABLE ENTERS THE ROOM, SECOND WHERE TURNS 90 DEGREES
TOWARDS PUMP AND THIRD WHERE CABLE ENTERS PUMP. TYPE IN THE
COORDINATES WITH COMMAS BETWEEN THEM (X1,Y1,X2,Y2,X3,Y3):

0.2,0
0.2,2
4,2

INPUT THE HEIGHT FROM THE FLOOR TO THE CABLES:

3.5

INPUT TWO OPPOSITE CORNER COORDINATES OF PUMP STARTING WITH THE
ONE NEAREST TO THE DOOR: X1,Y1,X2,Y2

3,1
5,3

ENTER THE HEIGHT OF THE PUMP (FROM THE FLOOR)

1.9


THE ROOM IS DIVIDED INTO A NET OF COORDINATES FOR OILSPILL
POSITIONS. INPUT THE DIFFERENTIAL NET SIZE (DX,DY):

1,1

DO YOU WANT TO SAVE YOUR INPUT DATA IN A FILE? (Y/N):

Y

CHOOSE A NAME FOR THIS INPUT FILE:

T9916.IN

CALCULATING........
```

```
GIVE YOUR OUTPUT FILE A NAME:

T9916.UT

CHOOSE YOUR OUTPUT OPTION: SHORT FORM OR LONG FORM (S/L):

L

WRITING TO FILE.........

C:\>
```

The input file can be saved in a file named by the user. Under the MS-DOS operating system this name can be fourteen characters long, of these, the last four can be a period plus a three-character extension. The first two may be drive specifiers.

The following shows the input file generated by the above run, named T9916.IN.

```
 22.00000    8.000000    6.000000
T
 2.000000    1.000000
 150.0000
 200.0000
 20.00000
T
0.0000000E+00  10.00000    8.000000    10.00000
F
0.2000000    0.0000000E+00 0.2000000    2.000000    2.000000
 4.000000
 2.000000
 3.100000    1.100000    4.900000    2.900000
 1.900000
 0.500000    0.500000
```

If several cases for the same compartment are to be run, the quickest way of doing it is to edit the input file, change the desired values and run the program by just specifying the input file. By comparing the interactive example and the input file above the user can identify the position of the variables he wants to change, and thus, run the program in a less time consuming way.

5 The resulting output file

The user must specify a name for the output file, this is done in a similar way as described for the input file. The output file can be viewed either by printing it on the screen or a printer or by reading it into an editor. It can either be in short form or in long form.

A small part of the long form output file from the run discussed in chapter 4 is shown below. The input variables for the run are written at the top of this file:

```
INPUT VARIABLES:

ROOM LENGTH, BREADTH AND HEIGHT = 22.00   8.00   6.00

DOOR HEIGHT AND BREADTH = 2.00  1.00

CRITICAL SURFACE TEMP. OF CABLE AND PUMP =  150.0   200.0

MAXIMUM AMOUNT OF OILSPILL (LITERS) =  20.0

A WALL SPLITS THE ROOM IN TWO PARTS.

WALL COORDINATES ARE:
 0.00  10.00
 8.00  10.00

X AND Y CABLE COORDINATES:
 0.20   0.00
 0.20   2.00
 2.00   4.00

HEIGHT FROM FLOOR TO CABLE =  2.00

X AND Y CORNER-COORDINATES OF PUMP:
 3.10   1.10
 4.90   2.90

PUMP HEIGHT =  1.90

DIFFERENTIAL NET SIZE DX AND DY :  0.50   0.50
```

After this the output for each oilspill size is displayed. The following only shows a part of the long form for one (10 liters) oilspill size. The short form output only contains the variables listed within the two lines below (*** and ---).

Displayed below the dashed line (---) are the times to damage due to different modes of heat transfer. The time to damage as a result of radiation from the flames depends on the position of the oilspill, these times are therefore written for each oilspill position. Time to damage due to heat transfer from the gaslayer is not dependent on the position of the oilspill and is therefore a single number for each oilspill size. The critical time to damage is the shortest of these times. The number 1666.67 stands for infinity. This value is displayed when, for example, the maximum gastemperature is lower than the critical cable temperature, consequently it takes an infinitely long time for the cable to reach it's critical temperature.

The resulting probability of damage for each position is also given. If the time to extinction is shorter than the critical time to damage the probability of damage is said to be zero. Otherwise the probability is given the value on the normalized frequency distribution curve for the oilspill position. In this example we assume that there is a 60% probability that the oilspill will be within a 1 meter radius from any pump in the room. There are 125 positions in the room that satisfy this criteria, each of these positions therefore gets a probability of 0.48%. All other positions divide the remaining 40% between them, resultin in a probability of 0.17% each.

```
**************************************************

SPILL NO 5              SPILL AMOUNT (LITERS) =  10.0

OILSPILL AREA = 2.00      FLAME HEIGHT = 3.56

ENERGY OUTPUT(KW) = 2413.8  MAX. GASTEMPERATURE (DEG C) =  242.9

TIME TO FLAME EXTINCTION (MINUTES) =  4.76

PROBABILITY OF DAMAGE FOR THIS OILSPILL % =  29.68
--------------------------------------------------

TIME TO DAMAGE AS A RESULT OF HEAT TRANSFER FROM:

GASLAYER TO CABLE (MINUTES) =   31.68

GASLAYER TO PUMP        =  1666.67

PLUME TO CABLES         =   3.12
```

POS. NO	FLAME TO PUMP	CRITICAL TIME	PROBABILITY
1	100.39	3.12	0.17
2	55.23	3.12	0.17
3	29.94	29.94	0.00
4	16.19	16.19	0.00
5	8.96	8.96	0.00
6	5.39	5.39	0.00
7	4.10	4.10	0.17
8	4.06	4.06	0.17
9	4.06	4.06	0.17
10	4.06	4.06	0.17
11	4.10	4.10	0.17
12	5.39	5.39	0.00
13	8.96	8.96	0.00
14	16.19	16.19	0.00
15	29.94	29.94	0.00
16	55.23	31.68	0.00
17	100.39	31.68	0.00
18	85.05	3.12	0.17
19	44.97	3.12	0.17
20	23.02	23.02	0.00
21	11.40	11.40	0.00
22	5.39	5.39	0.00
23	2.36	2.36	0.17
24	2.08	2.08	0.48
25	2.08	2.08	0.48
26	2.08	2.08	0.48

etc
etc
etc

Having written these tables up for each oilspill size the program displayes the overall probability of damage at the bottom of the output file:

```
##########################################

THE PROBABILITY THAT A FIRE WILL CAUSE DAMAGE IS = 28.50

##########################################
```

Bibliography:

(1) Karlsson, B., "A Deterministic and probabilistic Model for Oilspill Fires in Nuclear Power Plants". Dept. of Fire Safety Eng., Lund Inst. of Tech. Sept. 1989.

(2) Babrauskas, V., "Estimating Large Pool Fire Burning Rates". Fire Technology, Vol. 19, No. 4, pp 251-261; November 1983.

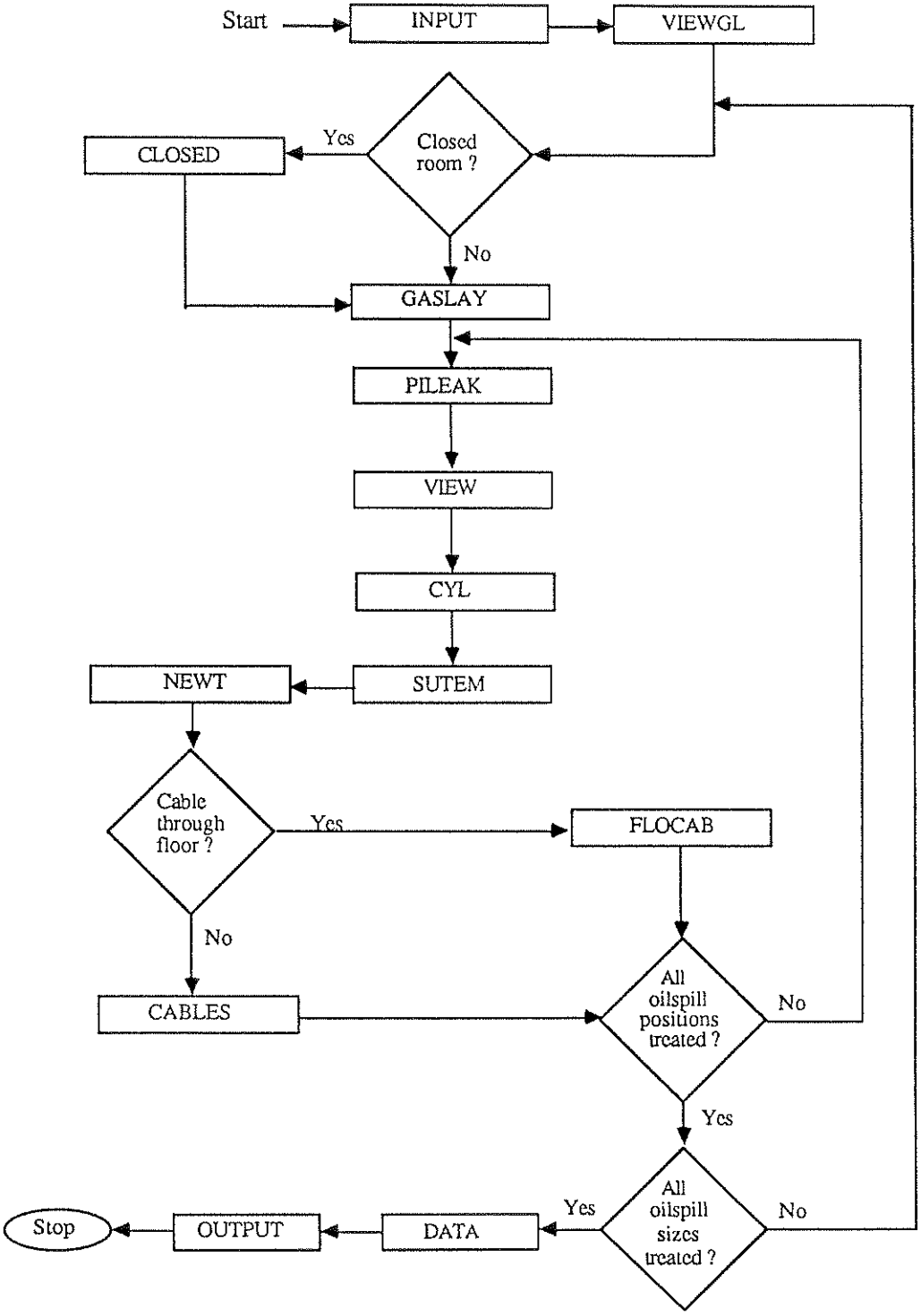

Figure 1. Program flow diagram.

Table 1: Input variables:

L	=	Length of the room (m)
B	=	Breadth of the room (m)
H	=	Height of the room (m)
HO	=	Height of the opening (m)
BO	=	Breadth of the opening (m)
X1C, Y1C,	=	A corner coordinate of a cable (also X2C,Y2C,X3C,Y3C) (m)
XCFLO, YCFLO	=	Coordinates of floor cable (m)
CABHGT	=	Height of the cables from the floor (m)
PX1, PY2	=	Corner coordinate of pump (also PX2, PY2) (m)
PMPHGT	=	Height of the pump from the floor (m)
DX,DY	=	The differential size of the net the room is divided into
OPEN	=	True if the room has an opening
CABFLO	=	True if the cable emerges from the floor but not through the wall
SCREEN	=	True if the room is split in two parts by a separating wall
TCCRIT	=	Critical core temperature of the cable (° C)
TPCRIT	=	Critical surface temperature of the pump (° C)
MAXLIT	=	Maximum amount of oil that can be spilled in a room (liters)
X1W,Y1W	=	Coordinates of separating wall (also X2W,Y2W) (m)

Table 2: Data constants:

DH	=	The heat of combustion of the oil (kJ/kg)
DENS	=	Density of the oil (kg/m^3)
THICKN	=	The thickness of the oilspill (m)
KBETA	=	Extinction coefficient times the mean beam length corrector (m^{-1})
MFREE	=	Infinite-diameter pool mass loss rate ($kg/m^2/s$)
ROCORE	=	Core density of the cable
CPCORE	=	Heat capacity of the core of the cable
RADCAB	=	Outer radius of the cable
RADCOR	=	Core radius of the cable
KPVC	=	Thermal conductivity of cable's insulation material (W/m.°K)
T0CAB	=	Initial temperature of the cable (°C)
T0PMP	=	Initial temperature of the pump (°C)
KROC	=	Thermal inertia of the compartment wall material ($W^2s/m^4{}^\circ K^2$)
KRCPMP	=	Thermal inertia of the pump material ($kW^2s/m^4{}^\circ K^2$)
RO	=	Density of air (kg/m^3)
CP	=	Heat capacity of air
CONV	=	Convective heat transfer coefficient ($W/m^2{}^\circ K$)

Table 3: Output variables:

K	=	Oilspill size number
N	=	Oilspill position number
AREA(K)	=	Area of oilspill no. K (m^2)
LITERS(K)	=	Amount of oil for oilspill no. K (liters)
Q(K)	=	Energy output of oilspill no. K (kW)
TSLUT(K)	=	Time to burn-out (s)
GASTEM(K)	=	Maximum gastemperature of the hot layer for oilspill no. K (°C)
FLH(K)	=	Flame height of oilspill no. K (m)
TIME90(K)	=	Characteristic time to damage (s)
TIMEGC(K)	=	Time to critical core temperature as a result of heat transfer from gaslayer to cable for oilspill no.K (s)
TIMEPC(K)	=	Time to critical core temperature as a result of heat transfer from the plume to the cable for oilspill no.K (s)
TIMEGP(K)	=	Time to critical surface temperature as a result of heat transfer from the gaslayer to the pump for oilspill no. K.
TIMEFP(N,K)	=	Time to critical surface temperature as a result of heat transfer from the flame to the pump for oilspill size no. K and oilspill position no. N (s)
TIMEFC(N,K)	=	Time to critical core temperature as a result of heat transfer from the flame to the cable for oilspill size no. K and oilspill position no. N (s)
TIME(N,K)	=	The shortest time to critical temperature (of the five above mentioned times) for oilspill size no. K and oilspill position no. N (s)
PROB(K)	=	The probability of damage to a component for oilspill size no. K
PROBAB	=	The overall probability of damage to components

```
C ******************************************************************
C THIS PROGRAM (OSFIC) CALCULATES THE RISK FOR FAILURE OF A
C CRITICAL COMPONENT (CABLE OR PUMP) AS A RESULT OF AN OILSPILL
C FIRE IN A COMPARTMENT.
C ******************************************************************
C PROGRAMMER: KARLSSON,B
C
C REFERENCE:  MAGNUSSON,S E
C         DEPT. OF FIRE SAFETY ENGINEERING
C         LUND INSTITUTE OF TECHNOLOGY
C         LUND, SWEDEN
C
      INTEGER I,NOVAL,MAXPOS,K,MAXNO,P,OUT,IR,IW,MAX,IGNORE
      PARAMETER (MAXPOS=1000,MAXNO=10,MAX=20)
      REAL TIME(1:MAXPOS,1:MAXNO),SORT(1:MAXPOS,1:MAXNO)
      REAL TIMEFP(1:MAXPOS,1:MAXNO),TIMEFC(1:MAXPOS,1:MAXNO)
      REAL TSLUT(1:MAXNO),GASTEM(1:MAXNO),TIME90(1:MAXNO),AREA(1:MAXNO)
      REAL TPLUME(1:MAXNO),FLH(1:MAXNO),Q(1:MAXNO),PROBAB(1:MAXNO),TOTAL
      REAL TIMEC(1:MAXPOS,1:MAXNO),LITERS(1:MAXNO)
      REAL TIMEGC(1:MAXNO),TIMEGP(1:MAXNO),TIMEPC(1:MAXNO),TOPMP
      REAL X1,X2,OPFACT,MAIR,ATOT,MP,HK,QFO,R,LIM,MBURN,FINISH,TOCAB
      REAL DH,DENS,PI,KROC,ROOTG,CP,RO,TO,KELVIN,MFREE,THICKN,DX,DY
      REAL L,B,H,HO,BO,PX1,PX2,PY1,PY2,TPCRIT,TCCRIT,MAXLIT,KBETA
      REAL CABHGT,PMPHGT,GASHGT,PLURAD,X,Y,F,DLIT,PXL1,PXL2,PLY1,PLY2
      REAL X1C,X2C,X3C,Y1C,Y2C,Y3C,XCFLO,YCFLO,PROB,J,X1W,Y1W,X2W,Y2W
      REAL LREAL,BREAL,ROCORE,CPCORE,RADCAB,RADCOR,KPVC,KRCPMP,CONV
      REAL SPPOS(1:MAXPOS),SPPROB(1:MAXPOS,1:MAXNO),B1,L1
      LOGICAL CABFLO,OPEN,PLEAK,YAXES,OUTPT,SCREEN,OK,SAME
      CHARACTER*60 NAME,NAMN
      CHARACTER*3 ANS
C
C DATA FOR OIL
C
      DATA DH/46000/,DENS/900/,THICKN/0.005/,KBETA/0.7/,MFREE/0.039/
C
C DATA FOR CABLES
C
      DATA ROCORE/2957./CPCORE/875.5/,RADCAB/0.0365/,RADCOR/0.02/
      DATA KPVC/0.09/,TOCAB/50.0/
C
C OTHER DATA
C
```

```
      DATA LIM/0.001/,IR/5/,IW/6/,PI/3.14159/,KROC/2.0/,KRCPMP/1.6E8/
      DATA ROOTG/3.132/,CP/1.0/,RO/1.2/,TO/20.0/,KELVIN/273.15/
      DATA CONV/30.0/,TOPMP/50/
C
C  CALLING INPUT SUBROUTINE
C
      CALL INPUT(L,B,H,HO,BO,X1C,Y1C,X2C,Y2C,X3C,Y3C,CABHGT,XCFLO,YCFLO
     &  ,PX1,PY1,PX2,PY2,PMPHGT,PLX1,PLY1,PLX2,PLY2,DX,DY,OPEN,CABFLO
     &  ,PLEAK,SCREEN,TCCRIT,TPCRIT,MAXLIT,IGNORE,X1W,Y1W
     &  ,X2W,Y2W)
      WRITE(IW,100)
  100 FORMAT(/,' CALCULATING.......')

C
C NUMBER OF DIFFERENT OILSPILL POSITIONS, AND ORIENTATION OF PUMPS IF
C PUMP IS LEAKING
C
      PLEAK=.FALSE.
      LREAL=L
      BREAL=B
      L1=L
      B1=B
      TOTAL=0
C
C IF THERE IS A SCREEN THEN TAKE ACCOUNT OF THIS
C
      IF (SCREEN) THEN
        IF (ABS(X1W-X2W).LT.LIM) THEN
          B=X1W
          IF (PX1.GT.X1W) THEN
            SAME=.FALSE.
            B1=BREAL-B
          ELSE
            SAME=.TRUE.
          ENDIF
        ELSEIF (ABS(Y1W-Y2W).LT.LIM) THEN
          L=Y1W
          IF (PY1.GT.Y1W) THEN
            SAME=.FALSE.
            L1=LREAL-L
          ELSE
            SAME=.TRUE.
          ENDIF
```

```
      ELSE
        WRITE(*,*) ' SOMETHING IS WRONG WITH WALL POSITION'
        GOTO 800
      ENDIF
    ELSE
      SAME=.TRUE.
    ENDIF
C
C  CALCULATING CONFIGURATION FACTOR FROM CEILING LAYER TO PUMP
C
    IF(OPEN) THEN
      GASHGT=H0
      D=GASHGT-PMPHGT
      IF(D.GT.0) THEN
        CALL VIEWGL(LREAL,BREAL,L,B,L1,B1,D,PX1,PY1,PX2,PY2,F,SAME)
      ELSE
        F=1.0
      ENDIF
    ELSE
      F=1.0
    ENDIF
C
C THE NEXT FEW LINES READ IN PROBABILITIES FOR OILSPILL POSITION
C
    OPEN(UNIT=10,FILE='PROB.DAT',STATUS='OLD')
    DO 17 K=1,NOVAL
      READ(10,*) SPPOS(K)
 17 CONTINUE
    CLOSE(UNIT=10)
C
C THIS LOOP TAKES US THROUGH 10 DIFFERENT OILSPILL SIZES
C
 5  CONTINUE
    J=0
    DLIT=MAXLIT/10
    DO 50 K=1,10
      LITERS(K)=DLIT*K
      R=SQRT(0.001*LITERS(K)/(THICKN*PI))
      AREA(K)=PI*R**2
C
C IF ROOM HAS OPENINGS THEN:
C
```

```
      IF (OPEN) THEN
       GASHGT=HO
       GOTO 10
      ELSE
       GOTO 20
      ENDIF
C
C  CALCULATING ENERGY RELEASE, PYROLISES RATE ETC. FOR ROOM WITH OPENING
C
 10  CONTINUE
      OPFACT=HO*BO*SQRT(HO)
      MAIR=0.5*OPFACT
      MBURN=MFREE*(1-EXP(-KBETA*R*2))
C
C  CHECK IF BURNING IS FUEL- OR VENTILATION CONTROLLED.
C  EFFICIENCY OF COMBUSTION IS (CONSERVATIVELY) ASSUMED TO BE 1.0
C  BUT, TO CALCULATE HOW LONG THE FIRE LASTS, EFFIFIENCY OF COMBUSTION IS
C  ASSUMED TO BE 0.6
C
      IF ((MAIR*0.232/(MBURN*PI*R**2)).GE.(DH/13200)) THEN
       Q(K)=1.0*MBURN*(R**2)*PI*DH
       TSLUT(K)=0.001*LITERS(K)*DENS/(0.6*MBURN*(R**2)*PI)
      ELSE
       Q(K)=MAIR*0.232*13200
       MP=MAIR*0.232*13200/DH
       TSLUT(K)=DENS*0.001*LITERS(K)/(MP*0.6)
      ENDIF
      FLH(K)=-1.02*2*R+0.23*(Q(K)**0.4)
C
C CALCULATING QFO, TEMP ETC FOR ROOM WITH OPENING
C
      ATOT=LREAL*BREAL*2+H*LREAL*2+BREAL*H*2
      HK=SQRT(KROC/TSLUT(K))
      QFO=610*SQRT(HK*ATOT*OPFACT)
C
C  QUINTIERE FORMULA FOR GASTEMPERATURES
C
      X1=Q(K)/(ROOTG*CP*RO*(TO+KELVIN)*OPFACT)
      X2=HK*ATOT/(ROOTG*CP*RO*OPFACT)
      GASTEM(K)=(TO+KELVIN)*1.6*(X1**0.6667)*(X2**(-0.333))+TO
      GOTO 30
C
C  FOR ROOM WITH NO OPENING
```

```
C
 20 CONTINUE
    MBURN=MFREE*(1-EXP(-KBETA*R*2))
    FINISH=DENS*0.001*LITERS(K)/(MBURN*(R**2)*PI*0.6)
    Q(K)=1.0*MBURN*(R**2)*PI*DH
    CALL CLOSED(K,MAXNO,Q,LREAL,BREAL,H,TSLUT,GASTEM,FINISH)
    FLH(K)=-1.02*2*R+0.23*(Q(K)**0.4)
 30 CONTINUE
C
C CALL SUBROUTINE GASLAY WHICH CALCULATES HEAT TRANSFER FROM GASLAYER
C TO CABLES AND PUMP.
C
    CALL GASLAY(GASTEM,TIMEGC,TIMEPC,TIMEGP,TPCRIT,TCCRIT,R,Q,CABHGT
   &  ,PLURAD,K,MAXNO,TPLUME,CABFLO
   &  ,ROCORE,CPCORE,RADCAB,RADCOR,KPVC,KRCPMP,CONV,TOCAB,TOPMP,F)
C
C CALL SUBROUTINE PILEAK THIS ROUTINE PLACES
C THE OILSPILL AT POSSIBLE POSITIONS IN THE ROOM AND
C CALCULATES THE TIME TO PUMP AND CABLE FAILURE AS A FUNCTION OF
C OILSPILL POSITIONS.
C
     CALL PILEAK(L,B,X,Y,DX,DY,PX1,PX2,PY1,PY2,R,FLH,TPCRIT
   & ,TIME,TIMEGP,X1C,Y1C,X2C,Y2C,X3C,Y3C,PLURAD,TIMEGC,TIMEPC,
   & TIMEC,CABFLO,XCFLO,YCFLO,K,MAXNO,P,NOVAL,TIMEFC,TIMEFP,TCCRIT
   & ,PLEAK,SCREEN,SAME,ROCORE,CPCORE,RADCAB,RADCOR,KPVC,CONV,KRCPMP
   & ,TOCAB,TOPMP)
C
C TO SORT THE DATA CALL SUBROUTINE DATA
C
    CALL DATA(NOVAL,TIME,SORT,TIME90,K,MAXNO,PLEAK,IGNORE
   &     ,PROBAB,SPPOS,TSLUT,SPPROB)
C
C FIND PROBABILITY
C
 45   CONTINUE
     IF (TSLUT(K).LT.TIME90(K)) THEN
      J=J+1.0
     ENDIF
    TOTAL=PROBAB(K)+TOTAL
 50  CONTINUE
    PROB=1.0-J/10.0
    TOTAL=TOTAL/10.0
```

```
C
C OUTPUT (TIMES ARE IN MINUTES), FIRST CHECK FILE STATUS
C
 580 WRITE(IW,590)
 590 FORMAT(/,' GIVE YOUR OUTPUT FILE A NAME:',/)
    READ(IR,595) NAME
 595 FORMAT(A60)
    INQUIRE (FILE=NAME,EXIST=OK)
    IF (OK) THEN
     WRITE(IW,710)
 710   FORMAT(/,' THIS FILE ALREADY EXISTS, DO YOU WANT TO',
   .      ' WRITE OVER IT? (Y/N):',/)
     READ(IR,720) ANS
 720   FORMAT(A1)
     IF (ANS.EQ.'Y') THEN
      OPEN(UNIT=10,FILE=NAME,STATUS='OLD',ERR=597)
      CLOSE(UNIT=10,STATUS='DELETE')
      GOTO 730
     ELSEIF (ANS.NE.'Y' .AND. ANS.NE.'N') THEN
      GOTO 580
     ELSE
      GOTO 580
     ENDIF
    ENDIF
C
C  CHOOSING OUTPUT OPTION
C
 730 CONTINUE
C
 740   WRITE(IW,750)
 750   FORMAT(/,' CHOOSE YOUR OUTPUT OPTION: SHORT FORM OR LONG',
   .      ' FORM (S/L):',/)
     READ(IR,760) ANS
 760   FORMAT(A1)
     IF (ANS.EQ.'S')THEN
      OUTPT=.TRUE.
     ELSEIF (ANS .NE. 'S' .AND. ANS .NE. 'L') THEN
      GOTO 740
     ELSE
      OUTPT=.FALSE.
     ENDIF
C
C   WRITING TO FILE
```

```
C
   WRITE(IW,599)
599 FORMAT(/,' WORKING.........',/)
   OUT=10
   OPEN(UNIT=10,FILE=NAME,STATUS='NEW',ERR=597)
   WRITE(OUT,600)
600 FORMAT(/,' INPUT VARIABLES:')
   WRITE(OUT,605)LREAL,BREAL,H
605 FORMAT(/,' ROOM LENGTH, BREADTH AND HEIGHT = ',3F6.2)
   IF (OPEN) THEN
    WRITE(OUT,610)HO,BO
610   FORMAT(/,' DOOR HEIGHT AND BREADTH = ',2F5.2)
   ENDIF
   WRITE(OUT,650)TCCRIT,TPCRIT
650 FORMAT(/,' CRITICAL SURFACE TEMP. OF CABLE AND PUMP = ',2F7.1)
   WRITE(OUT,660)MAXLIT
660 FORMAT(/,' MAXIMUM AMOUNT OF OILSPILL (LITERS) = ',F6.1)
   IF (SCREEN) THEN
    WRITE(OUT,670)
670   FORMAT(/,' A WALL SPLITS THE ROOM IN TWO PARTS.')
    WRITE(OUT,672)X1W,Y1W,X2W,Y2W
672   FORMAT(/,' WALL COORDINATES ARE: ',/2F6.2,/,2F6.2)
   ENDIF
   IF (CABFLO) THEN
    WRITE(OUT,625)XCFLO,YCFLO
625   FORMAT(/,' X AND Y COORDINATE OF FLOOR CABLE: ',2F6.2)
   ELSE
    WRITE(OUT,615) X1C,Y1C,X2C,Y2C,X3C,Y3C
615   FORMAT(/,' X AND Y CABLE COORDINATES: ',/,2F6.2,/,2F6.2,/,2F6.2)
    WRITE(OUT,620)CABHGT
620   FORMAT(/,' HEIGHT FROM FLOOR TO CABLE = ',F6.2)
   ENDIF
   WRITE(OUT,630)PX1,PY1,PX2,PY2
630 FORMAT(/,' X AND Y CORNER-COORDINATES OF PUMP: ',/,2F6.2,/,2F6.2)
   WRITE(OUT,635)PMPHGT
635 FORMAT(/,' PUMP HEIGHT = ',F6.2)
   IF (PLEAK) THEN
    WRITE(OUT,640) PLX1,PLY1,PLX2,PLY2
640   FORMAT(/,' X AND Y CORNER-COORDINATES OF LEAKING PUMP: ',/,
     .       2F6.2,/,2F6.2)
   ELSE
    WRITE(OUT,645) DX,DY
```

```
645   FORMAT(/,' DIFFERENTIAL NET SIZE DX AND DY : ',2F5.2)
      ENDIF
      WRITE(OUT,655) IGNORE
655   FORMAT(/,' IGNORING ',I3,'% OF THE SHORTEST CRITICAL TIMES')
      GOTO 700
597   WRITE(IW,598)
598   FORMAT(/,' ERROR OPENING OUTPUT FILE. ')
      GOTO 580
C
C  CALLING OUTPUT ROUTINE TO WRITE OUT TABLES
C
700   CONTINUE
      CALL OUTPUT(MAXNO,NOVAL,LITERS,AREA,FLH,Q,GASTEM,TSLUT,TIME90
     & ,TIME,TIMEGC,TIMEGP,TIMEPC,TIMEFP,TIMEFC,OUTPT,OUT,CABFLO,PROB
     & ,PROBAB,SPPOS,TOTAL,SPPROB)
      CLOSE (UNIT=10)
      DO 789 K=1,MAXNO
        WRITE(*,*) LITERS(K),PROBAB(K)
789   CONTINUE
800   CONTINUE
      STOP
      END
```

```
C ***************************************************************
  SUBROUTINE CABLES(X1C,Y1C,X2C,Y2C,X3C,Y3C,X,Y,PLURAD,TIMEGC,
  &      TIMEPC,TIMEC,K,MAXNO,P,NOVAL)
C ***************************************************************
C THIS SUBROUTINE CHECKS WHETHER THE CABLES LIE WITHIN THE PLUME RADIUS
C AND IF SO THEN IT FINDS THE HEAT TRANSFERE FROM THE PLUME TO THE CABLE
C AND THE CRITICAL TIME IS TIMEPC. ELSE THE CRITICAL TIME IS TIMEGC
C (= GASLAYER TO CABLE).
C
  INTEGER K,MAXNO,NOVAL,P
  REAL X1C,Y1C,X2C,Y2C,X3C,Y3C,X,Y,PLURAD
  REAL TIMEPC(1:MAXNO),TIMEC(1:NOVAL,1:MAXNO),TIMEGC(1:MAXNO)
  REAL XCFLO,YCFLO,XPLUSR,XMINR,YPLUSR,YMINR,LIM
  LOGICAL CABFLO
  DATA LIM/0.001/
C
  XPLUSR=X+PLURAD
  XMINR=X-PLURAD
  YPLUSR=Y+PLURAD
  YMINR=Y-PLURAD
C
C TO FIND IF CABLES ARE WITHIN THE PLUME RADIUS. FIRST IF CABLE LIES
C PARALELL WITH THE LENGTH OF THE ROOM WHEN IT ENTERS THE ROOM.
C
  IF (ABS(X1C-X2C).LE.LIM) THEN
   IF (XMINR.LT.X1C .AND. XPLUSR.GT.X1C) THEN
    GOTO 50
   ELSEIF (YMINR.LT.Y2C .AND. YPLUSR.GT.Y2C) THEN
    GOTO 50
   ELSE
    GOTO 60
   ENDIF
C
C IF CABLES ARE PARALELL TO THE BREADTH OF THE ROOM WHEN THEY ENTER IT.
C
  ELSEIF (ABS(Y1C-Y2C).LE.LIM) THEN
   IF (YMINR.LT.Y1C .AND. YPLUSR.GT.Y1C) THEN
    GOTO 80
   ELSEIF (XMINR.LT.X2C .AND. XPLUSR.GT.X2C) THEN
    GOTO 80
   ELSE
    GOTO 60
```

```
      ENDIF
     ELSE
      WRITE(*,*) ' CABLE POSITION WRONG !'
      GOTO 100
     ENDIF
C
C FOR CABLES PARALELL TO THE LENGTH OF THE ROOM WHEN ENTERING IT
C
 50 CONTINUE
     IF (Y1C.GT.Y2C .AND. X2C.LT.X3C) THEN
      IF (XMINR.GT.X3C .OR. XPLUSR.LT.X2C .OR. YMINR.GT.Y1C
    &            .OR. YPLUSR.LT.Y2C) THEN
       GOTO 60
      ELSE
       GOTO 70
      ENDIF
     ELSEIF (Y1C.GT.Y2C .AND. X2C.GT.X3C) THEN
      IF (XMINR.GT.X2C .OR. XPLUSR.LT.X3C .OR. YMINR.GT.Y1C
    &            .OR. YPLUSR.LT.Y2C) THEN
       GOTO 60
      ELSE
       GOTO 70
      ENDIF
     ELSEIF (Y1C.LT.Y2C .AND. X2C.LT.X3C) THEN
      IF (XMINR.GT.X3C .OR. XPLUSR.LT.X2C .OR. YMINR.GT.Y2C
    &            .OR. YPLUSR.LT.Y1C) THEN
       GOTO 60
      ELSE
       GOTO 70
      ENDIF
     ELSEIF (Y1C.LT.Y2C .AND. X2C.GT.X3C) THEN
      IF (XMINR.GT.X2C .OR. XPLUSR.LT.X3C .OR. YMINR.GT.Y2C
    &            .OR. YPLUSR.LT.Y1C) THEN
       GOTO 60
      ELSE
       GOTO 70
      ENDIF
     ELSE
      WRITE(*,*) ' SOMETHING IS WRONG WITH CABLE POSITIONS ERROR 1'
      GOTO 100
     ENDIF
C
C FOR CABLES PERPENDICULAR TO THE LENGTH OF THE ROOM WHEN ENTERING IT
```

```
C
 80 CONTINUE
   IF (X1C.GT.X2C .AND. Y2C.LT.Y3C) THEN
    IF (YMINR.GT.Y3C .OR. YPLUSR.LT.Y2C .OR. XMINR.GT.X1C
  &            .OR. XPLUSR.LT.X2C) THEN
     GOTO 60
    ELSE
     GOTO 70
    ENDIF
   ELSEIF (X1C.GT.X2C .AND. Y2C.GT.Y3C) THEN
    IF (YMINR.GT.Y2C .OR. YPLUSR.LT.Y3C .OR. XMINR.GT.X1C
  &            .OR. XPLUSR.LT.X2C) THEN
     GOTO 60
    ELSE
     GOTO 70
    ENDIF
   ELSEIF (X1C.LT.X2C .AND. Y2C.LT.Y3C) THEN
    IF (YMINR.GT.Y3C .OR. YPLUSR.LT.Y2C .OR. XMINR.GT.X2C
  &            .OR. XPLUSR.LT.X1C) THEN
     GOTO 60
    ELSE
     GOTO 70
    ENDIF
   ELSEIF (X1C.LT.X2C .AND. Y2C.GT.Y3C) THEN
    IF (YMINR.GT.Y2C .OR. YPLUSR.LT.Y3C .OR. XMINR.GT.X2C
  &            .OR. XPLUSR.LT.X1C) THEN
     GOTO 60
    ELSE
     GOTO 70
    ENDIF
   ELSE
    WRITE(*,*) ' SOMETHING IS WRONG WITH CABLE POSITIONS ERROR 2'
    GOTO 100
   ENDIF
C
C FOR CABLES OUTSIDE PLUME RADIUS
C
 60 CONTINUE
   TIMEC(P,K)=TIMEGC(K)
   GOTO 100
C
C FOR CABLES INSIDE PLUME RADIUS
```

```
C
 70 CONTINUE
    IF (TIMEPC(K).LT.TIMEGC(K)) THEN
     TIMEC(P,K)=TIMEPC(K)
    ELSE
     TIMEC(P,K)=TIMEGC(K)
    ENDIF
    GOTO 100
C
 100 CONTINUE
    RETURN
    END
```

```
C   ****************************************************************
    SUBROUTINE CLOSED(K,MAXNO,Q,L,B,H,TSLUT,GASTEM,FINISH)
C   ****************************************************************
C  THIS ROUTINE CALCULATES THE SMOKE FILLING TIME AND GASTEMPERATURE IN
C  A CLOSED COMPARTMENT.  REFERENCE:  ZUKOSKI, "DEVELOPMENT OF A STRATIFIED
C  CEILING LAYER IN THE EARLY STAGES OF A CLOSED-ROOM FIRE", FIRE AND
C  MATERIALS, VOL 2 NO 2, 1978.
C
    REAL Q(1:MAXNO),TSLUT(1:MAXNO),GASTEM(1:MAXNO)
    REAL ROAIR,CP,HELP,L,B,H,TO,FINISH,Y
    INTEGER K
    DATA HELP/-0.851/,ROAIR/1.2/,CP/1.0/,G/9.81/,TO/293/
C
C TIME TO FILL THE ROOM WITH SMOKE CALCULATED FROM ZUKOSKI'S EQUNS.
C
    QSTAR=Q(K)/(ROAIR*CP*TO*SQRT(G*H)*H**2)
    TAU=1.045*(QSTAR**(HELP))
C
C TIME TILL SMOKE LAYER REACHES FLOOR IS CALCULATED. AS IN THE CASE
C FOR OPEN COMPARTMENTS THE RESULT IS (CONSERVATIVELY) DIVIDED BY 0.6.
C ONE MINUTE ADDED FOR FILLING TIME
C
    TSLUT(K)=60.0+(TAU/(SQRT(G/H)*(H**2/(L*B))))/0.6
    IF (TSLUT(K).GT.FINISH) THEN
     TSLUT(K)=FINISH
    ENDIF
    GASTEM(K)=TO/(1-QSTAR*TAU)
C
C  GASTEMPERATURE RETURNED IN CELSIUS
C
    GASTEM(K)=GASTEM(K)-273.15
    RETURN
    END
```

```
C ******************************************************************
  REAL FUNCTION CYL(A,B,C)
C ******************************************************************
C THIS IS REAL FUNCTION CYL. IT CALCULATES THE CONFIGURATION FACTOR
C FROM A CYLENDRICAL OBJECT TO A SURFACE ELEMENT PARALELL TO THE
C CYLENDER AXES.
C     REFERENCE:  SIEGEL.R , HOWELL.J.R
C             "THERMAL RADIATION HEAT TRANSFER"
C              MCGRAW-HILL BOOK COMPANY
C              NEW YORK, 1981.
C
  REAL A,B,C,X,Y,A1,B1,R1,R2,R3,R4,F,PI
  DATA PI/3.14159/
C
C DEFINING HELP VARIABLES TO MAKE FINAL EXPRESSION SIMPLE
C
  IF (C.LT.B) THEN
   F=1.0
   WRITE(*,*) ' WARNING, CHECK VARIABLES C AND B IN ROUTINE CYL! '
   GOTO 10
  ENDIF
  X=A/B
  Y=C/B
  A1=(1+Y)**2+X**2
  B1=(1-Y)**2+X**2
  R1=X/SQRT(Y**2-1)
  R2=(A1-2*Y)/(Y*SQRT(A1*B1))
  R3=SQRT(A1*(Y-1)/(B1*(Y+1)))
  R4=SQRT((Y-1)/(Y+1))
C
C EXPRESSION FOR CONFIGURATION FACTOR
C
   F=(1/(PI*Y))*ATAN(R1)+(X/PI)*(R2*ATAN(R3)-(1/Y)*ATAN(R4))
 10 CONTINUE
  CYL=F
  RETURN
  END
```

```
C ******************************************************************
   SUBROUTINE DATA(NOVAL,TIME,SORT,TIME90,K,MAXNO,PLEAK,IGNORE
  & ,PROBAB,SPPOS,TSLUT,SPPROB)
C ******************************************************************
C THIS SUBROUTINE SORTS OUT THE DATA. CRITICAL TIMES TO DAMAGE HAVE
C BEEN CALCULATED AS A FUNCTION OF OILSPILL POSITION, TIME(P). THIS
C ROUTINE SORTS THESE TIMES BY SIZE AND FINDS TIME90, THAT IS 90% OF
C THE OILSPILL POSITIONS GAVE A HIGHER CRITICAL TIME THAN TIME90.
C
   REAL TIME(1:NOVAL,1:MAXNO),SORT(1:NOVAL,1:MAXNO),TIME90(1:MAXNO)
   REAL A,SPPOS(1:NOVAL),TSLUT(1:MAXNO),PROBAB(1:MAXNO)
   REAL SPPROB(1:NOVAL,1:MAXNO)
   LOGICAL KLAR,PLEAK
   INTEGER NOVAL,I,K1,J,K,MAXNO,IGNORE
C
C THIS TAKES TIME(K) AND SORTS IT, SORT(I) ARE THE TIMES IN ACENDING ORDER
C
   DO 10 K1=1,NOVAL
    I=K1-1
    KLAR=.FALSE.
 20   IF (I.GT.0 .AND. .NOT.KLAR) THEN
      IF (TIME(K1,K).LT.SORT(I,K)) THEN
       SORT(I+1,K)=SORT(I,K)
       I=I-1
      ELSE
       KLAR=.TRUE.
      ENDIF
    GOTO 20
    ENDIF
    SORT(I+1,K)=TIME(K1,K)
 10 CONTINUE
C
C FINDING PROBABILITY OF DAMAGE FOR EACH OILSPILL SIZE, PROBAB(K)
C
   DO 30 I=1,NOVAL
    IF(TIME(I,K).LT.TSLUT(K))THEN
     PROBAB(K)=SPPOS(I)+PROBAB(K)
     SPPROB(I,K)=SPPOS(I)
    ELSE
     SPPROB(I,K)=0
     PROBAB(K)=SPPROB(I,K)+PROBAB(K)
    ENDIF
```

```
 30 CONTINUE
C
C NOW THE TIMES HAVE BEEN SORTED BY SIZE. FIND TIME90.
C
   IF (PLEAK) THEN
    TIME90(K)=SORT(1,K)
   ELSE
    A=(IGNORE/100.0)*NOVAL
    J=NINT(A)
    IF (J.LT.1) THEN
     J=1
    ELSEIF (J.GT.NOVAL) THEN
     J=NOVAL
    ENDIF
    TIME90(K)=SORT(J,K)
   ENDIF
C
   RETURN
   END
```

```
C ***********************************************************************
   SUBROUTINE FLOCAB(TCCRIT,FLH,R,XCFLO,YCFLO,X,Y,TIMEFC,N,NOVAL
  & ,K,MAXNO,PMPHGT,SIDE,PLEAK,ROCORE,CPCORE,RADCAB,RADCOR,KPVC
  & ,CONV,TOCAB)
C ***********************************************************************
C THIS ROUTINE CALCULATES THE TIME TO CRITICAL CORE TEMPERATURE OF
C A CABLE (WHICH LEADS FROM THE FLOOR TO A PUMP) AS A RESULT OF
C RADIATION FROM A CYLENDRICAL FLAME. THE CORE TEMPERATURE IS
C CALCULATED ACCORDING TO THE LUMPED MASS CAPACITY METHOD
C
   INTEGER N,NOVAL,K,MAXNO
   REAL TCCRIT,R,XCFLO,YCFLO,X,Y,TIMEFC(1:NOVAL,1:MAXNO)
   REAL TFLAME,BOLTZM,EMISS,TO,ROCORE,CPCORE,RADCOR,C,A,F1,F,S
   REAL FLH(1:MAXNO),KELVIN,TWOPIR,F2,F3,RCSIDE,RCPMP,RCFLH,RCRAD
   REAL PMPHGT,SIDE,B,CONV,QRAD,TC1,TC2,KPVC,RADCAB,TOCAB
   LOGICAL PLEAK
C
   DATA TFLAME/1300/,BOLTZM/5.67E-8/,EMISS/0.95/,TO/293/
   DATA TWOPIR/0.15915/,KELVIN/273.15/
C
C FIRST, THE DISTANCE FROM FLAME TO CABLE, THEN CONFIGURATION FACTOR
C
   C=SQRT(ABS(XCFLO-X)**2+ABS(YCFLO-Y)**2)
   B=2*R
 20 CONTINUE
    IF(C.LT.R) THEN
     F=1.0
    ELSE
     A=FLH(K)/2
     F1=CYL(A,R,C)
     F=2*F1
    ENDIF
C
C NOW THE TIME TO CRITICAL SURFACE TEMPERATURE
C
 40 CONTINUE
   QRAD=(TFLAME**4-TO**4)*EMISS*BOLTZM*F
   TC1=ROCORE*CPCORE*RADCOR/(2*CONV)
   TC2=(KPVC/(RADCAB-RADCOR)+CONV)/(KPVC/(RADCAB-RADCOR))
   HELP=1-(TCCRIT-TOCAB)*CONV/QRAD
   IF (HELP.LE.0) THEN
    TIMEFC(N,K)=100000
```

```
      ELSE
        TIMEFC(N,K)=-LOG(HELP)*TC1*TC2
      ENDIF
      IF (TIMEFC(N,K).GT.100000) THEN
        TIMEFC(N,K)=100000
      ENDIF
C
      RETURN
      END
```

```
C ******************************************************************
      SUBROUTINE GASLAY(GASTEM,TIMEGC,TIMEPC,TIMEGP,TPCRIT,TCCRIT,R,Q,
     & CABHGT,PLURAD,K,MAXNO,TPLUME,
     & CABFLO,ROCORE,CPCORE,RADCAB,RADCOR,KPVC,KRCPMP,CONV,TOCAB,
     & TOPMP,F)
C ******************************************************************
C THIS SUBROUTINE CALCULATES THE TIMES TO CRITICAL SURFACE-TEMPERATURE
C OF THE CABLES AND PUMP AS A RESULT OF THE HOT GAS LAYER.
C
      INTEGER K,MAXNO
      REAL GASTEM(1:MAXNO),TIMEGC(1:MAXNO),TIMEPC(1:MAXNO)
      REAL TIMEGP(1:MAXNO),Q(1:MAXNO),TPLUME(1:MAXNO),TOPMP,TOCAB,F
      REAL CABHGT,PLURAD,NEWT,TPCRIT,TCCRIT,R,CONV
      REAL ROCORE,FPCORE,RADCOR,RADCAB,KPVC,TO,CPAIR,ROAIR,PI,KRCPMP
      REAL BOLTZM,EMISS,TC,ZO,ROOTDB,ROOTDL,Q1,A,K1,B1,C,KELVIN
      LOGICAL CABFLO
C
      DATA CPAIR/1.0/,ROAIR/1.1/,PI/3.1416/,KELVIN/273./
      DATA BOLTZM/5.67E-8/,EMISS/1.0/,TO/20/
C
C HEAT TRANSFER FROM GASLAYER TO CABLES, TIMEGC IS THE TIME TO CRITICAL
C CABLE SURFACE-TEMPERATURE. IF GASTEM < TCCRIT THEN TIMEGC IS A LARGE
C NUMBER
C
      IF (CABFLO) THEN
        TIMEGC(K)=100000
        TIMEPC(K)=100000
        GOTO 20
      ENDIF
      TC=ROCORE*CPCORE*(RADCOR**2)*(RADCAB-RADCOR)/(2*KPVC*RADCAB)
      IF (GASTEM(K).LE.TCCRIT) THEN
        TIMEGC(K)=100000
      ELSE
        TIMEGC(K)=-TC*LOG(1-(TCCRIT-TOCAB)/(GASTEM(K)-TOCAB))
      ENDIF
C
C HEAT TRANSFERE FROM PLUME TO CABLE. FIRST PLUME-TEMP AND PLUME-RADIUS
C ARE CALCULATED, THEN TIMEPC. IF THERE IS A SCREEN THEN GO PAST THIS
C
      ZO=-1.02*2*R+0.083*(Q(K)**0.4)
      TPLUME(K)=TO+9.1*(((TO+KELVIN)/(9.81*(CPAIR**2)*(ROAIR**2)))**
     &      0.333)*((Q(K)*0.7)**0.667)*((CABHGT-ZO)**(-1.667))
```

```
      PLURAD=0.12*SQRT((TPLUME(K)+KELVIN)/(TO+KELVIN))*(CABHGT-ZO)
      IF (TPLUME(K).LT.TCCRIT) THEN
       TIMEPC(K)=100000
      ELSE
       TIMEPC(K)=-TC*LOG(1-(TCCRIT-TOCAB)/(TPLUME(K)-TOCAB))
      ENDIF
 20   CONTINUE
C
C HEAT TRANSFER FROM THE GASLAYER TO THE PUMP. C THE NEWTON-RAPHSON METHOD
C TIMEGP FOR SEMI-INFINITE SOLID.
C
      IF (GASTEM(K).LT.TPCRIT) THEN
       TIMEGP(K)=100000
      ELSE
       Q1=((GASTEM(K)+KELVIN)**4)*BOLTZM*EMISS*F
       A=CONV/SQRT(KRCPMP)
       K1=1+4/SQRT(PI)
       B1=SQRT(PI)*(A**2)
       C=1-CONV*(TPCRIT-TOPMP)/Q1
C
C IF RADIATION FROM GASLAYER IS VERY LOW THEN TIME IS PUT TO VERY HIGH.
C
       IF (C.LE.0) THEN
        TIMEGP(K)=100000
       ELSE
        TIMEGP(K)=NEWT(A,K1,B1,C)
       ENDIF
       IF (TIMEGP(K).GT.100000) THEN
        TIMEGP(K)=100000
       ENDIF
      ENDIF
C
C RETURN
C
      RETURN
      END
```

```
C   ***************************************************************
    SUBROUTINE INPUT(L,B,H,HO,BO,X1C,Y1C,X2C,Y2C,X3C,Y3C,CABHGT,XCFLO
   & ,YCFLO,PX1,PY1,PX2,PY2,PMPHGT,PLX1,PLY1,PLX2,PLY2,DX,DY,
   & OPEN,CABFLO,PLEAK,SCREEN,TCCRIT,TPCRIT,MAXLIT,IGNORE
   & ,X1W,Y1W,X2W,Y2W)
C   ***************************************************************
C  THIS IS THE INPUT ROUTINE OF THE PROGRAM
C
C
    REAL L,B,H,HO,BO,X1C,Y1C,X2C,Y2C,X3C,Y3C,CABHGT,XCFLO,YCFLO
    REAL PX1,PY1,PX2,PMPHGT,PY2,PLX1,PLY1,PLX2,PLY2,DX,DY
    REAL TCCRIT,TPCRIT,MAXLIT,X1W,Y1W,X2W,Y2W
    INTEGER IR,IW,INP,IGNORE
    LOGICAL OPEN,CABFLO,PLEAK,SCREEN,OK,SAME
    CHARACTER*60 NAME,TEMP
    CHARACTER*3 ANS
    IR=5
    IW=6
    WRITE(IW,10)
 10 FORMAT(/,' WELCOME TO THE INPUT PORTION OF THE PROGRAM.  THIS',
   .    ' SECTION ALLOWS YOU',/,
   .    ' TO INITIALIZE YOUR VARIABLES EITHER INTERACTIVELY OR',
   .    ' BY A PREEXISTING INPUT',/,
   .    ' FILE.  IT THEN GIVES YOU THE OPTION TO SAVE THE INPUT',
   .    ' DATA IN A FILE (NAMED',/,
   .    ' BY YOU). ALL LENGTHS AND COORDINATES ARE IN METERS.',/,
   .    ' THE CARTESIAN COORDINATE SYSTEM USED IS: WHEN ENTERING',
   .    ' THROUGH THE DOOR, THE',/,
   .    ' ORIGIN IS IN THE CORNER ON YOUR LEFT, THE X-AXES IS',
   .    ' ALONG THE WALL WITH',/,
   .    ' THE DOOR. THE LENGTH OF THIS WALL IS THE BREADTH OF',
   .    ' THE ROOM.  HOWEVER, IF A',/,
   .    ' WALL SPLITS THE ROOM IN TWO PARTS, THE ORIGIN MUST BE',
   .    ' IN THE SAME PART AS',/,
   .    ' THE OILSPILL',/)
C
 15 WRITE(IW,30)
C
 30 FORMAT(/,' WOULD YOU LIKE ONE OF YOUR  INPUT DATA FILES',
   .     ' TO BE READ IN AND',/,
   .     ' INITIALIZE THE INPUT VARIABLES? (Y/N):',/)
    READ(IR,40) ANS
```

```
 40   FORMAT(A1)
C
    IF (ANS .EQ. 'Y') THEN
C
C IF YES, THEN FIND OUT ITS NAME, OPEN THE FILE, AND READ IT IN
C AND CLOSE IT
C
      WRITE(IW,50)
 50     FORMAT(/,' PLEASE ENTER THE NAME OF THE FILE TO BE USED',/)
      READ(IR,52) NAME
 52     FORMAT(A60)
      INP=8
      OPEN(UNIT=8,FILE=NAME,STATUS='OLD',ERR=11)
      READ(INP,*,ERR=12) L,B,H
      READ(INP,*,ERR=12) OPEN
      IF (OPEN) THEN
        READ(INP,*,ERR=12) HO,BO
      ENDIF
      READ(INP,*,ERR=12) TCCRIT
      READ(INP,*,ERR=12) TPCRIT
      READ(INP,*,ERR=12) MAXLIT
      READ(INP,*,ERR=12) SCREEN
      IF (SCREEN) THEN
        READ(INP,*,ERR=12) X1W,Y1W,X2W,Y2W
      ENDIF
      READ(INP,*,ERR=12) CABFLO
      IF (CABFLO) THEN
        READ(INP,*,ERR=12) XCFLO,YCFLO
      ELSE
        READ(INP,*,ERR=12) X1C,Y1C,X2C,Y2C,X3C,Y3C
        READ(INP,*,ERR=12) CABHGT
      ENDIF
      READ(INP,*,ERR=12) PX1,PY1,PX2,PY2
      READ(INP,*,ERR=12) PMPHGT
      READ(INP,*,ERR=12) DX,DY
      READ(INP,*,ERR=12) IGNORE
C
 70     CONTINUE
      CLOSE (UNIT=8)
      GOTO 1000
 11     WRITE(IW,55)
 55     FORMAT(/,' FILE NOT FOUND')
      GO TO 15
```

```
12    WRITE(IW,56)
56    FORMAT(/,' ERROR IN INPUT FILE')
     GOTO 15
    ELSEIF (ANS .NE. 'N' .AND. ANS .NE. 'Y') THEN
     GOTO 15
    ELSE
C
C INTERACTIVE INPUT    1: COMPARTMENT
C
100   WRITE(IW,110)
110   FORMAT(/,' TYPE IN ROOM LENGTH,BREADTH AND HEIGHT (IN METERS)',
  .     ' SEPERATED BY A COMMA.',/)
     READ(IR,*) L,B,H
115   WRITE(IW,120)
120   FORMAT(/,' IS THE DOOR TO THE COMPARTMENT OPEN ? (Y/N):',/)
     READ(IR,130) ANS
130   FORMAT(A1)
     IF(ANS.EQ.'Y') THEN
      OPEN=.TRUE.
      WRITE(IW,140)
140    FORMAT(/,' TYPE IN THE HEIGHT AND THE BREADTH OF THE DOOR,',
  .     ' SEPERATED BY A COMMA.',/)
      READ(IR,*) HO,BO
     ELSEIF (ANS .NE. 'N' .AND. ANS .NE. ' ') THEN
      GOTO 115
     ENDIF
     WRITE(IW,150)
150   FORMAT(/,' ENTER THE CRITICAL SURFACE TEMPERATURE OF THE',
  .     ' CABLE (CELCIUS):',/)
     READ(IR,*) TCCRIT
     WRITE(IW,160)
160   FORMAT(/,' ENTER THE CRITICAL SURFACE TEMPERATURE OF THE',
  .     ' PUMP (CELCIUS):',/)
     READ(IR,*) TPCRIT
     WRITE(IW,170)
170   FORMAT(/,' ENTER THE MAXIMUM AMOUNT OF OIL (LITERS) THAT CAN',
  .     ' BE SPILLED',/,
  .     ' IN THIS ROOM:',/)
     READ(IR,*) MAXLIT
175   WRITE(IW,180)
180   FORMAT(/,' DOES A WALL SPLIT THE ROOM IN TWO PARTS ? (Y/N):',/)
     READ(IR,190) ANS
```

```
 190   FORMAT(A1)
      IF (ANS.EQ.'Y') THEN
       SCREEN=.TRUE.
       WRITE(IW,196)
 196    FORMAT(/,' INPUT THE X AND Y COORDINATES OF THE WALL',
     .        ' (X1,Y1,X2,Y2):',/)
       READ(IR,*) X1W,Y1W,X2W,Y2W
      ELSEIF (ANS .NE. 'N' .AND. ANS .NE. 'Y') THEN
       GOTO 175
      ELSE
       SCREEN=.FALSE.
      ENDIF
C
C                2: CABLES
C
 200   WRITE(IW,210)
 210   FORMAT(/,' DO THE CABLES ENTER THE ROOM THROUGH A WALL OR',
     .    ' FROM THE FLOOR ? (W/F):',/)
      READ(IR,220) ANS
 220   FORMAT(A1)
      IF (ANS.EQ.'F') THEN
       CABFLO=.TRUE.
       WRITE(IW,230)
 230    FORMAT(/,' ENTER THE X AND Y COORDINATE OF THIS CABLE',/)
       READ(IR,*) XCFLO,YCFLO
      ELSEIF (ANS .EQ. 'W') THEN
       CABFLO=.FALSE.
       WRITE(IW,240)
 240    FORMAT(/,' INPUT THREE COORDINATES OF CABLE POSITION. FIRST',
     .    ' COORDINATE IS WHERE ',/,
     .    ' CABLE ENTERS THE ROOM, SECOND WHERE TURNS 90 DEGREES',
     .    ' TOWARDS PUMP AND ',/,
     .    ' THIRD WHERE CABLE ENTERS PUMP. TYPE IN THE COORDINATES',
     .    ' WITH COMMAS BETWEEN ',/,
     .    ' THEM (X1,Y1,X2,Y2,X3,Y3):',/)
       READ(IR,*)X1C,Y1C,X2C,Y2C,X3C,Y3C
       WRITE(IW,250)
 250    FORMAT(/,' INPUT THE HEIGHT FROM THE FLOOR TO THE CABLES:',/)
       READ(IR,*) CABHGT
      ELSEIF (ANS .NE. 'W' .AND. ANS .NE. 'F') THEN
       GOTO 200
      ENDIF
C
```

```
C               3: PUMP DATA
C
 300   WRITE(IW,310)
 310   FORMAT(/,' INPUT TWO OPPOSITE CORNER COORDINATES OF PUMP',
  .     ' STARTING WITH THE ONE ',/,
  .     ' NEAREST TO THE DOOR: X1,Y1,X2,Y2 ',/)
     READ(IR,*) PX1,PY1,PX2,PY2
     WRITE(IW,320)
 320   FORMAT(/,' ENTER THE HEIGHT OF THE PUMP (FROM THE FLOOR)',/)
     READ(IR,*) PMPHGT
     WRITE(IW,370)
 370    FORMAT(/,' THE ROOM IS DIVIDED INTO A NET OF COORDINATES',
  .     ' FOR OILSPILL POSITIONS.',/,
  .     ' INPUT THE DIFFERENTIAL NET SIZE (DX,DY): ',/)
      READ(IR,*) DX,DY
 375   WRITE(IW,380)
 380   FORMAT(/,' DO YOU WANT TO IGNORE A CERTAIN PRECENTAGE OF THE',
  .     ' SHORTEST TIMES TO ',/,
  .     ' CRITICAL DAMAGE ? (Y/N);',/)
     READ(IR,385) ANS
 385   FORMAT(A3)
     IF (ANS.EQ.'Y') THEN
      WRITE(IW,390)
 390    FORMAT(/,' INPUT THE PRECENTAGE OF THE SHORTEST TIMES TO',
  .     ' CRITICAL DAMAGE YOU',/,
  .    ' WOULD LIKE TO IGNORE (EXAMPLE: 10 MEANS YOU IGNORE 10%)'/)
      READ(IR,*) IGNORE
     ELSEIF(ANS.NE.'Y' .AND. ANS.NE.'N') THEN
      GOTO 375
     ELSE
      IGNORE=0
     ENDIF
C
C               4: SAVING INPUT FILE
C
 500   WRITE(IW,510)
 510   FORMAT(/,' DO YOU WANT TO SAVE YOUR INPUT DATA IN A FILE?',
  .     ' (Y/N):',/)
     READ(IR,520) ANS
 520   FORMAT(A1)
     IF (ANS .EQ. 'Y') THEN
 525    WRITE(IW,530)
```

```
530     FORMAT(/,' CHOOSE A NAME FOR THIS INPUT FILE:',/)
      READ(IR,540) TEMP
540     FORMAT(A60)
      INQUIRE (FILE=NAME,EXIST=OK)
      IF (OK) THEN
       WRITE(IW,570)
570      FORMAT(/,' THIS FILE ALREADY EXISTS, DO YOU WANT TO',
     .  ' WRITE OVER IT? (Y/N):',/)
       READ(IR,573) ANS
573      FORMAT(A1)
       IF (ANS.EQ.'Y') THEN
        OPEN(UNIT=9,FILE=TEMP,STATUS='OLD',ERR=550)
        CLOSE(UNIT=9,STATUS='DELETE')
        GOTO 575
       ELSEIF (ANS.NE.'Y' .AND. ANS.NE.'N') THEN
        GOTO 525
       ELSE
        GOTO 525
       ENDIF
      ENDIF
575     CONTINUE
      INP=9
      OPEN(UNIT=9,FILE=TEMP,STATUS='NEW',ERR=550)
      WRITE(INP,*) L,B,H
      WRITE(INP,*) OPEN
      IF (OPEN) THEN
       WRITE(INP,*) HO,BO
      ENDIF
      WRITE(INP,*) TCCRIT
      WRITE(INP,*) TPCRIT
      WRITE(INP,*) MAXLIT
      WRITE(INP,*) SCREEN
      IF (SCREEN) THEN
       WRITE(INP,*) X1W,Y1W,X2W,Y2W
      ENDIF
      WRITE(INP,*) CABFLO
      IF (CABFLO) THEN
       WRITE(INP,*) XCFLO,YCFLO
      ELSE
       WRITE(INP,*) X1C,Y1C,X2C,Y2C,X3C,Y3C
       WRITE(INP,*) CABHGT
      ENDIF
      WRITE(INP,*) PX1,PY1,PX2,PY2
```

```
      WRITE(INP,*) PMPHGT
      WRITE(INP,*) DX,DY
      WRITE(INP,*) IGNORE
C
 545    CONTINUE
      CLOSE (UNIT=9)
      GOTO 1000
C
 550    WRITE(IW,560)
 560    FORMAT(/,' ERROR OPENING FILE FOR NEW INPUT DATA')
     ELSEIF (ANS .NE. 'Y' .AND. ANS .NE. 'N') THEN
      GOTO 500
     ENDIF
    ENDIF
 1000 CONTINUE
    RETURN
    END
```

```
C ******************************************************************
   REAL FUNCTION NEWT(A,K,B,C)
C ******************************************************************
C THIS NUMERICAL ROUTINE FINDS THE TIME (T) REQUIRED FOR AN OBJECT
C TO REACH A CERTAIN SURFACE TEMPERATURE. THE OBJECT IS ASSUMED
C TO BE SEMI-INFINITE. THE EQATION USED IS:
C   TSCRIT-TO = Q/H * (1-EXP(-X**2)*ERFC(X)
C       WHERE X = H**2 * SQRT(T/KROC)
C REFERENCE: "HEAT TRANSFER" HOLMAN, J.P
C         MCGRAW-HILL, 1976, NEW YORK.
C
   REAL FXO,FDASHX,X,XO,A,K,B,C,ERROR,UPPLIM,LOWLIM,FX,HELP1,HELP2
   DATA ERROR/0.001/,UPPLIM/1E15/,LOWLIM/-1000/
C
   XO=0.1
 10 CONTINUE
C
C DEFINING FUNCTION AND ITS DERIVATIVE
C
    FXO=(2+A*SQRT(XO))/(2+K*A*SQRT(XO)+B*XO)-C
    HELP1=A*(1-K)/SQRT(XO)-0.5*A*B*SQRT(XO)-2*B
    HELP2=(2+K*A*SQRT(XO)+B*XO)**2
   FDASHX=HELP1/HELP2
C
C NEWTON-RAPHSON METHOD
C
   X=XO-FXO/FDASHX
C
C CHECKING IF SOLUTION IS WITHIN ERROR LIMITS AND IS CONVERGENT
C
    FX=(2+A*SQRT(X))/(2+K*A*SQRT(X)+B*X)-C
    IF (ABS(FX-FXO).LT.ERROR) THEN
     GOTO 20
    ELSE(X.GT.UPPLIM .OR. X.LT.LOWLIM) THEN
     WRITE(*,*) 'NUMERICAL PROBLEM'
     GOTO 20
    ENDIF
 20 CONTINUE
   NEWT=X
   RETURN
   END
```

```
C ****************************************************************
   SUBROUTINE OUTPUT(MAXNO,NOVAL,LITERS,AREA,FLH,Q,GASTEM,TSLUT
  & ,TIME90,TIME,TIMEGC,TIMEGP,TIMEPC,TIMEFP,TIMEFC,OUTPT,OUT
  & ,CABFLO,PROB,PROBAB,SPPOS,TOTAL,SPPROB)
C ****************************************************************
   INTEGER OUT,MAXNO,NOVAL,K,I
   REAL LITERS(1:MAXNO),AREA(1:MAXNO),FLH(1:MAXNO),Q(1:MAXNO)
   REAL GASTEM(1:MAXNO),TSLUT(1:MAXNO),TIMEGC(1:MAXNO)
   REAL TIMEGP(1:MAXNO),TIMEPC(1:MAXNO),TIME90(1:MAXNO)
   REAL TIME(1:NOVAL,1:MAXNO),TIMEFP(1:NOVAL,1:MAXNO)
   REAL TIMEFC(1:NOVAL,1:MAXNO),PROB,PROBAB(1:MAXNO)
   REAL TOTAL,SPPOS(1:NOVAL),SPPROB(1:NOVAL,1:MAXNO)
   LOGICAL OUTPT,CABFLO
   DO 10 K=1,10
    WRITE(OUT,90)
 90   FORMAT(/,' ***************************************************')
    WRITE(OUT,100) K,LITERS(K)
100   FORMAT(/,' SPILL NO ',I2,'              SPILL AMOUNT',
  .       ' (LITERS) = ',F6.1)
    WRITE(OUT,110) AREA(K),FLH(K)
110   FORMAT(/,' OILSPILL AREA = ',F5.2,'       FLAME HEIGHT = ',
  .       F5.2)
    WRITE(OUT,120) Q(K),GASTEM(K)
120   FORMAT(/,' ENERGY OUTPUT(KW) = ',F7.1,'   MAX. GASTEMPERATURE',
  .       ' (DEG C) = ',F7.1)
    WRITE(OUT,130) TSLUT(K)/60
130   FORMAT(/,' TIME TO FLAME EXTINCTION (MINUTES) = ',F7.2)
    WRITE(OUT,143)PROBAB(K)
143   FORMAT(/,' PROBABILITY OF DAMAGE FOR THIS OILSPILL % = ',F7.2)
    IF (GASTEM(K).GE.1026) THEN
     WRITE(OUT,90)
     WRITE(OUT,145)
145    FORMAT(/,' MODEL NOT VALID FOR FIRES OF THIS SIZE IN A',
  .       ' CLOSED ROOM.',/,
  .       ' RISK OF EXPLOSION DUE TO INCOMPLETE COMBUSTION.',/)
     WRITE(OUT,90)
     RETURN
    ENDIF
     WRITE(OUT,150)
150    FORMAT(/,'-----------------------------------------------')
     WRITE(OUT,160)
160    FORMAT(/,' TIME TO CRITICAL DAMAGE AS A RESULT OF RADIATION',
```

```
 .        ' FROM:')
      WRITE(OUT,170) TIMEGC(K)/60
170     FORMAT(/,' GASLAYER TO CABLE (MINUTES) = ',F9.2)
      WRITE(OUT,180) TIMEGP(K)/60
180     FORMAT(/,' GASLAYER TO PUMP          = ',F9.2)
      WRITE(OUT,190) TIMEPC(K)/60
190     FORMAT(/,' PLUME TO CABLES           = ',F9.2).
    IF (OUTPT) THEN
     GOTO 10
    ELSE
     IF (CABFLO) THEN
     WRITE(OUT,205)
200     FORMAT(/,' POS. NO  FLAME TO PUMP  RESULTING TIME',
 .        ' PROB')
     ELSE
     WRITE(OUT,200)
205     FORMAT(/,' POS. NO  FLAME TO PUMP  FLAME TO CABLE',
 .        ' RESULTING TIME  PROB')
     ENDIF
     DO 50 I=1,NOVAL
      IF (CABFLO) THEN
        WRITE(OUT,220)I,TIMEFP(I,K)/60,TIMEFC(I,K)/60,TIME(I,K)/60
 .       ,SPPROB(I,K)
210       FORMAT(TR4,I3,TR6,F9.2,TR6,F9.2,TR6,F5.2)
      ELSE
        WRITE(OUT,210)I,TIMEFP(I,K)/60,TIME(I,K)/60,SPPROB(I,K)
220       FORMAT(TR4,I3,TR6,F9.2,TR6,F9.2,TR6,F9.2,TR6,F5.2)
      ENDIF
 50   CONTINUE
    ENDIF
 10 CONTINUE
   WRITE(OUT,300)
300
 FORMAT(/,'##################################################')
   WRITE(OUT,310) TOTAL
310 FORMAT(/,' THE PROBABILITY THAT A FIRE WILL CAUSE DAMAGE IS',
 .     ' = ', F5.2)
   WRITE(OUT,320)
320
 FORMAT(/,'##################################################')
   RETURN
   END
```

```
C ******************************************************************
      SUBROUTINE PILEAK(L,B,X,Y,DX,DY,PX1,PX2,PY1,PY2,R,FLH,TPCRIT
     &  ,TIME,TIMEGP,X1C,Y1C,X2C,Y2C,X3C,Y3C,PLURAD,TIMEGC,TIMEPC,
     &  TIMEC,CABFLO,XCFLO,YCFLO,K,MAXNO,P,NOVAL,TIMEFC,TIMEFP,
     &  TCCRIT,PLEAK,SCREEN,SAME,ROCORE,CPCORE,RADCAB,RADCOR,KPVC
     &  ,CONV,KRCPMP,TOCAB,TOPMP)
C ******************************************************************
C THIS SUBROUTINE CREATES A NET OF COORDINATES IN A ROOM. THE
C OILSPILL IS THEN PLACED AT A CERTAIN COORDINATE AND THE RADIANT
C HEAT FLUX TO A RECTANGULAR OBJECT (THE PUMP) IN A ROOM IS CALCULATED.
C ALSO, THE HEATING OF THE CABLE IS CALCULATED, DEPENDING ON WHETHER IT
C IS SITUATED IN THE FIRE PLUME OR SUBMERGED IN THE HOT GASLAYER.
C
      INTEGER NOVAL,P,K,MAXNO
      REAL L,B,X,Y,DX,DY,PX1,PX2,PY1,PY2,F,R,FLH(1:MAXNO),TPCRIT
      REAL TIME(1:NOVAL,1:MAXNO),TIMEC(1:NOVAL,1:MAXNO)
      REAL TIMEGP(1:MAXNO),X1C,Y1C,X2C,Y2C,X3C,Y3C,TCCRIT
      REAL PLURAD,TIMEGC(1:MAXNO),TIMEPC(1:MAXNO),XCFLO,YCFLO
      REAL TIMEFP(1:NOVAL,1:MAXNO),TIMEFC(1:NOVAL,1:MAXNO),COUNT
      REAL ROCORE,CPCORE,RADCAB,RADCOR,KPVC,CONV,KRCPMP,TOCAB,TOPMP
      LOGICAL CABFLO,PLEAK,SCREEN,SAME
C
      P=0
      COUNT=0
      DO 10 Y=0,L,DY
       DO 20 X=0,B,DX
        P=P+1
        CALL VIEW(X,Y,PX1,PX2,PY1,PY2,F,R,FLH,K,MAXNO)
        CALL SUTEM(F,TPCRIT,NOVAL,P,TIMEFP,K,MAXNO,KRCPMP,CONV,TOPMP)
        IF (CABFLO) THEN
         CALL FLOCAB(TCCRIT,FLH,R,XCFLO,YCFLO,X,Y,TIMEFC,P,NOVAL,
     &    K,MAXNO,PMPHGT,SIDE,PLEAK,ROCORE,CPCORE,RADCAB,RADCOR,KPVC
     &    ,CONV,TOCAB)
         TIMEC(P,K)=100000
        ELSE
         CALL CABLES(X1C,Y1C,X2C,Y2C,X3C,Y3C,X,Y,PLURAD,TIMEGC
     &    ,TIMEPC,TIMEC,K,MAXNO,P,NOVAL)
         TIMEFC(P,K)=100000
        ENDIF
C
C FIND MINIMUM TIME, FIRST IF SCREEN IS PRESENT
C
```

```
      IF (SCREEN.AND. .NOT.SAME) THEN
       TIMEFP(P,K)=100000
       TIMEFC(P,K)=100000
      ENDIF
      COUNT=COUNT+1
       TIME(P,K)=MIN(TIMEGP(K),TIMEC(P,K),TIMEFP(P,K),TIMEFC(P,K))
 20    CONTINUE
 10  CONTINUE
C
    NOVAL=COUNT
    RETURN
    END
```

```
C ***************************************************************
   SUBROUTINE SUTEM(F,TPCRIT,NOVAL,P,TIMEFP,K,MAXNO,KRCPMP,CONV,
  & TOPMP)
C ***************************************************************
C THIS SUBROUTINE CALCULATES THE CONSTANT HEAT FLUX FROM A CYLENDRICAL
C FLAME TO A PUMP AND THEN THE SURFACE TEMPERATURE OF THE PUMP.
C PUMP IS ASSUMED TO BE OF SEMI-INFINITE THICKNESS. FORMULA FOR
C SURFACE TEMPERATURE IS:
C                TPCRIT-TOPMP = Q/H * (1-EXP(-X**2)*ERFC(X)
C                WHERE X = H*SQRT(T/KROC)
C
   INTEGER NOVAL,P,K,MAXNO
   REAL TFLAME,BOLTZM,EMISS,Q,F,TPCRIT,TO,PI,KRCPMP,CONV,A,B,C,K1
   REAL TIMEFP(1:NOVAL,1:MAXNO),NEWT,TOPMP
C
   DATA TFLAME/1300.0/,BOLTZM/5.67E-8/,EMISS/0.95/
   DATA PI/3.14159/,TO/20.0/
C
C CALCULATING HEAT FLUX AND TIME TO TSCRIT
C
   Q=(TFLAME**4)*BOLTZM*EMISS*F
   A=CONV/SQRT(KRCPMP)
   K1=1+4/SQRT(PI)
   B=SQRT(PI)*A**2
   C=1-CONV*(TPCRIT-TOPMP)/Q
C
C NUMERICAL FUNCTION NEWT SOLVES THE ABOVE EQUATION BY NEWTON-RAPHSON.
C IF RADIATION IS VERY LOW THEN TIME IS PUT TO VERY HIGH VALUE.
C
   IF (C.LT.0) THEN
    TIMEFP(P,K)=100000
   ELSE
    TIMEFP(P,K)=NEWT(A,K1,B,C)
    IF (TIMEFP(P,K).GE.100000) THEN
     TIMEFP(P,K)=100000
    ENDIF
   ENDIF
C
   RETURN
   END
```

```
C ***********************************************************************
      SUBROUTINE VIEW(X,Y,PX1,PX2,PY1,PY2,F,R,FLH,K,MAXNO)
C ***********************************************************************
C  THE ROOM HAS COORDINATES STARTING AT THE LEFT HAND SIDE WHEN STAND-
C  ING IN THE DOOR: (X,Y)=(0,0). TWO CORNERS OF A PUMP HAVE COORD-
C  INATES (PX1,PY1) AND (PX2,PY2). THE ROOM AND THE PUMP ARE CON-
C  SIDERED TO BE OF RECTANGULAR SHAPE AND PARALELL TO EACH OTHER.
C  THE CENTRE OF THE OILSPILL IS AT POINT (X,Y) IN THE ROOM.
C  THIS SUBROUTINE CALCULATES THE VIEW FACTOR FROM A CYLENDRICAL FLAME
C  OF RADIUS R AND HEIGHT FLH TO THE NEAREST POINT ON A RECTANGULAR
C  BOX (PUMP)
C
      REAL X,Y,PX1,PX2,PY1,PY2,F,R,FLH(1:MAXNO)
      REAL A1,A2,A3,A4,A,C,PI,CYL
      DATA PI/3.14159/
C
C  IF FLAME CENTRE IS INSIDE OR ON TOP OF PUMP
C
      IF (X.GE.PX1 .AND. X.LE.PX2 .AND. Y.GE.PY1 .AND. Y.LE.PY2) THEN
       GOTO 10
C
C  IF FLAME CYLENDER IS NOT AT RIGHT ANGLES TO ONE SIDE OF THE PUMP
C
      ELSEIF ((X.LE.PX1.OR.X.GE.PX2) .AND. (Y.LE.PY1.OR.Y.GE.PY2)) THEN
       GOTO 20
C
C  IF FLAME IS AT RIGHT ANGLES TO ONE SIDE OF PUMP
C
      ELSEIF (X.LT.PX1 .OR. X.GT.PX2) THEN
       GOTO 30
      ELSEIF (Y.LT.PY1 .OR. Y.GT.PY2) THEN
       GOTO 40
      ELSE
       WRITE(*,*) 'SOMETHINGS WRONG WITH PUMP POSITION, ROUTINE VIEW'
      ENDIF
 10   CONTINUE
      F=1.0
      GOTO 60
 20   CONTINUE
      A1=SQRT((ABS(X-PX1))**2+(ABS(Y-PY1))**2)
      A2=SQRT((ABS(X-PX2))**2+(ABS(Y-PY1))**2)
      A3=SQRT((ABS(X-PX1))**2+(ABS(Y-PY2))**2)
```

```
      A4=SQRT((ABS(X-PX2))**2+(ABS(Y-PY2))**2)
      C=MIN(A1,A2,A3,A4)
      GOTO 50
 30   CONTINUE
      C=MIN(ABS(X-PX1),ABS(X-PX2))
      GOTO 50
 40   CONTINUE
      C=MIN(ABS(Y-PY1),ABS(Y-PY2))
      GOTO 50
C
C  THE SHORTEST DISTANCE (A) BETWEEN CYLENDER CENTRE AND PUMP HAS BEEN
C  FOUND. NOW FIND VIEW FACTOR FOR 1/2 OF CYLENDER AND MULTIPLY BY 2.
C
 50   CONTINUE

C
C  IF CYLENDER TOUCHES PUMP THEN F=1 ELSE RETURN
C
      IF (C.LE.R) THEN
       F=1.0
      ELSE
       A=FLH(K)/2
       F1=CYL(A,R,C)
       F=2*F1
      ENDIF
C
 60   CONTINUE
      RETURN
      END
```

```
C ***********************************************************************
      SUBROUTINE VIEWGL(LREAL,BREAL,L,B,L1,B1,D,X1P,Y1P,X2P,Y2P,F,SAME)
C ***********************************************************************
C  THIS SUBROUTINE CALCULATES THE CONFIGURATION FROM THE GASLAYER TO A
C  CENTRAL ELEMENT ON THE TOP OF A PUMP. THE LAYER SURFACE IS SPLIT INTO
C  FOUR RECTANGLES, THE VEIW FACTORS FROM THESE ARE CALCULATED AND ADDED
C  TO GIVE THE RESULTING CONFIGURATION FACTOR.
C
C  REFERENCE FOR FORMULA ON CONFIGURATION FACTORS:
C   SIEGEL AND HOWELL: "THERMAL RADIATION HEAT TRANSFER"
C   MCGRAW-HILL, LONDON, 1981
C
      INTEGER N
      LOGICAL SAME
      REAL LREAL,BREAL,L,B,D,X1P,Y1P,X2P,Y2P,F,X,Y,ROOTDS,ROOTDR
      REAL TWOPIR,R,S,B1,L1
      DATA TWOPIR/0.15915/

C
      X=(X1P+X2P)/2
      Y=(Y1P+Y2P)/2
      N=0
      F=0
C
C IF THE OILSPILL AND THE PUMP ARE SEPERATED BY A WALL
C
      IF (SAME) THEN
       L1=L
       B1=B
      ENDIF
C
C  GO THROUGH THE FOUR RECTANGLES AND ADD THEM UP
C
 10   CONTINUE
      N=N+1
      GOTO(1,2,3,4),N
  1   R=X
      S=Y
      GOTO 20
  2   R=X
      S=L1-Y
      GOTO 20
```

```
 3 R=B1-X
   S=Y
   GOTO 20
 4 R=B1-X
   S=L1-Y
   GOTO 20
20 CONTINUE
   ROOTDS=SQRT(D**2+S**2)
   ROOTDR=SQRT(D**2+R**2)
   F=TWOPIR*(S/ROOTDS*ATAN(R/ROOTDS)+R/ROOTDR*ATAN(S/ROOTDR))+F
C
C IF ALL FOUR RECTANGLES HAVE BEEN TREATED THEN RETURN, ELSE GO BACK
C
   IF(N.LT.4) THEN
    GOTO 10
   ENDIF
   RETURN
   END
```

The following is the code of a small program which calculates the concentration of a gas in a compartment. It is included here since hydrogen leakage scenarios were also examined in the study.

```
C      ****************************************************************
C     THIS PROGRAM CALCULATES THE CONCENTRATION OF A GAS IN A ROOM
C     AFTER A CERTAIN TIME OF LEAKAGE.
C      ****************************************************************
C     PROGRAMMER:   BJÖRN KARLSSON
C                     DEPT. OF FIRE SAFETY ENGINEERING
C                    LUND INST. OF TECHNOLOGY
C                   LUND, SWEDEN
C
         REAL  C,QA,QG,V,T,HL,L,B,H,HELP,LIM1,LIM2
        INTEGER IW,IR
         DATA  IW/6/,IR/5/,HL/0.0/,  LIM1/4.0/,LIM2/7.0/
C
C     INPUT
C
        WRITE(IW,100)
 100    FORMAT(/,' WELCOME TO THE PROGRAM HYDROGEN. IT CALCULATES THE',
     .            ' CONCENTRATION OF',/,
     .             ' HYDROGEN IN A COMPARTMENT DUE TO A PIPE LEAKAGE OR',
     .            ' PIPE RUPRUTE.',/,/,
     .             ' INSERT THE LENGTH, WIDTH AND HEIGHT OF THE ROOM',
     .            ' (SEPERATED BY',/,
     .             ' A COMMA OR BLANK OR RETURN)',/)
        READ(IR,*) L,B,H
        WRITE(IW,120)
 120    FORMAT(/,' INPUT THE FLOW OF AIR PUSHED INTO THE ROOM BY THE',
     .             ' VENTILATION SYSTEM',/,
     .             ' (IN NORMAL CUBIC METERS PER HOUR)',/)
        READ(IR,*) QA
        WRITE(IW,130)
 130    FORMAT(/,' INPUT THE AMOUNT OF GAS LEAKED (NORMAL CUBIC METERS',
     .            ' PER HOUR)',/)
        READ(IR,*) QG
        WRITE(IW,140)
 140    FORMAT(/,' AT WHAT HEIGHT IN THE ROOM IS THE LEAKAGE ?',/)
        READ(IR,*) HL
        WRITE(IW,150)
 150    FORMAT(/,' FOR HOW LONG TIME DOES THE LEAKAGE CONTINUE ?',
     .            ' (MINUTES)',/)
        READ(IR,*) T
C
C     CALCULATIONS
C
        V=L*B*(H-HL)
         HELP=-(QG+QA)*T/(60*V)
          C=(100*QG/(QG+QA))*(1-EXP(HELP))
        IF (C.LT.LIM1) THEN
         GOTO 200
        ELSEIF (C.LT.LIM2) THEN
         GOTO 210
        ELSEIF (C.GT.LIM2) THEN
         GOTO 220
       ENDIF
C
C     OUTPUT
C
```

```
200   WRITE(IW,300) C
300   FORMAT(/,' HYDROGEN CONCENTRATION = ',F6.2,/,
     .           ' THIS IS UNDER THE FLAMMABILITY LIMIT, THEREFORE NO',
     .          ' RISK OF EXPLOSION.',/)
      GOTO 900
210   WRITE(IW,310) C
310   FORMAT(/,' HYDROGEN CONCENTRATION = ',F6.2,/,
     .           ' THIS IS OVER THE FLAMMABILITY LIMIT, BUT THE BURNING',
     .          ' WILL BE SLOW.',/)
      GOTO 900
220   WRITE(IW,320) C
320   FORMAT(/,' HYDROGEN CONCENTRATION = ',F6.2,/,
     .           ' THIS IS A HIGH CONCENTRATION, RISK OF EXPLOSION.',/,
     .           ' THIS SCENARIO SHOULD BE STUDIED IN DETAIL.',/)
900   CONTINUE
      STOP
      END
```