A

SUMMER TRAINING REPORT

ON

WATER MANAGEMENT SYSTEM

YEAR-2015-16

Submitted by

Prabhat Kumar

Enroll No.: 13/ICS/031

Gautam Buddha University

Greater Noida

Certificate

Date:

This is to certify that the project work entitled Water Management System" carried out by **Mr. Prabhat Kumar**, B.Tech, 2nd year, ***Gautam Buddha University, Greater Noida*** is an authentic record of work carried out by him during May 22, 2015 to July 22, 2015 under our guidance towards the fulfillment of Summer Training.

We appreciate the sincerity and hard work put by **Mr. Prabhat Kumar** for completing the task in two months period. We wish him success in his future endeavor.

Mr. P. K. Mishra
Scientist- B &OIC,
Computer Centre,
NIH, Roorkee

Dr. Sanjay K. Jain
Scientist - F & Head,
Computer Centre,
NIH, Roorkee

ACKNOWLEDGMENT

I take this opportunity with much pleasure to thanks all who have helped me through the course of my summer training.

I sincerely thanks ***Dr. Sanjay Kumar Jain***, Sc 'F' & Head, Computer Centre, NIH, Rookee and ***Mr. P.K. Mishra***, Sc 'B' & Officer In-charge, Computer Center, NIH, Rookee for their guidance, help and motivation.

I sincerely thanks **Mr. Abhishek Agarwal**, Resource Person (Junior) for constant help in coding the programme.

I would like to express my deep gratitude to **Mr. Manoj Goel, SRA,** and **Mr. Alok Kumar Sharma**, Computer Center (NIH, Rookee) for their support and help during the training programme.

(Prabhat Kumar)

ABSTRACT

Water, now-a-days, is very crucial interms of availability, distribution and management. With increased urbanization and population growth, the water demand will further rise in future. To cope up with water stress, sound water management is need of the hour.

Since past, water is managed traditionally with limited technological interventions. These days improved techniques have been developed resulting in efficient water management. Computational efficiency has also improved with advanced Computer/Work stations and complex Computer programmes.

The present work is an attempt to re-code FORTRAN programme into C# to characterize the stream flow of the Satluj river. Many programmes have been developed in FORTRAN which is not so much user friendly. In the present study, *IDE Visual Studio 2010 technology and, C# language* has been used to characterize the stream flow of the Satluj river which is user friendly.

Contents

Introduction

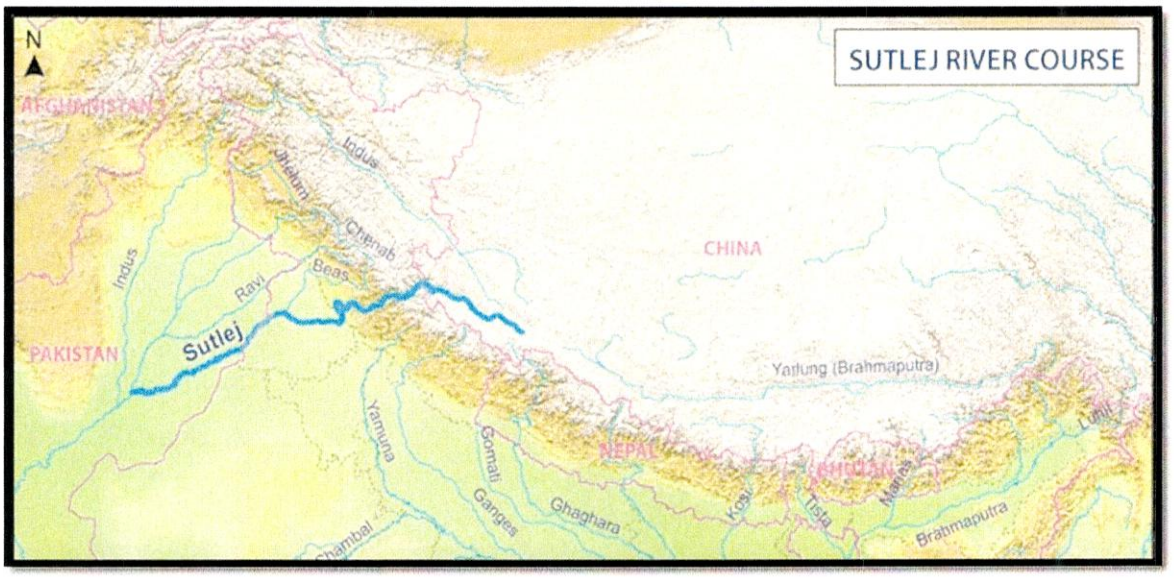
Streamflow in the Himalayan rivers is generated from rainfall, snow and ice. The distribution of runoff produced from these sources is such that the streamflow may be observed in these rivers throughout the year, i.e. they are perennial in nature. Snow and glacier melt runoff contributes substantially to the annual flows of these rivers and its estimation is required for the planning, development and management of the water resources of this region. The average contribution of snow and glacier melt runoff in the annual flows of the Satluj River at Bhakra Dam has been determined. Keeping in view the availability of data for the study basin, a water balance approach was used and a water budget period of 10 years (October 1986-September 1996) was considered for the analysis. The rainfall input to the study basin over the water budget period was computed from isohyets using rainfall data of 10 stations located at different elevations in the basin. The total volume of flow for the same period was computed using observed flow data of the Satluj River at Bhakra Dam. A relationship between temperature and evaporation was developed and used to estimate the évapotranspiration losses. The snow-covered area, and its depletion with time, was determined using satellite data. It was found that the average contribution of snow and glacier runoff in the annual flow of the Satluj River at Bhakra Dam is about 59%, the remaining 41% being from rain.

The Study Basin

The Satluj River has its source area in the Mansarovar and Rakastal lakes in the Tibetan Plateau at an elevation of about 4572 m (Fig. 1).

It is one of the main tributaries of the Indus River and flows through different areas having different climatic and topographic features. The Tibetan Plateau receives hardly any rainfall and thus has a cold desert type of climate. Therefore, very little flow is observed in this river when it passes through this dry region. After entering Indian territory, the Satluj River flows through Himachal Pradesh and receives runoff from snow and glaciers as well as rain. In the high altitude region, numerous glaciers drain directly into the Satluj River at various points along its course. The lower part of the basin experiences a considerable amount of rainfall. The location map of the study area has given in **Fig. 1.**

Fig. 1 Location map of the Satluj River.



The total catchment area of the Satluj River up to Bhakra Dam is about56 874 km^2. The Indian part of the Satluj basin, for which the present study is carried out, covers an area of about 22 305 km^2, including the whole catchment of the Spiti basin (a major tributary of the Satluj).

Streamflow Characteristics of The Satluj River

The streamflow of the Satluj River consists of the contribution from rain, snow and glaciers. These contributions from each components varies with time of the year. Generally, the snowmelt contribution starts from March and lasts until June/July, Generally, high discharges and floods are observed in the months of July and August and these are essentially due to heavy rain in the lower part of the basin. Glaciers contribute to their maximum in the months of July and August. As such, glacier melt runoff contribution lasts till September/October.

Based on analysis of 10 years (October 1986-September 1996) of flow data, the monthly distribution of streamflow at Bhakra Dam is shown in **Fig. 2** and quarterly distribution is given in **Table 1.**

It may be seen that maximum flow is observed in the month of July, followed by August, and the minimum in the month of February. About 83% of the total annual flow is observed during the pre-monsoon and monsoon seasons, because a high contribution from all the sources of runoff is produced in these two seasons. In the other two quarters, flow is only 17% of the total annual flows.

Fig. 2 Monthly average discharge of the Indian part of the Satluj River observed at Bhakra Dam.

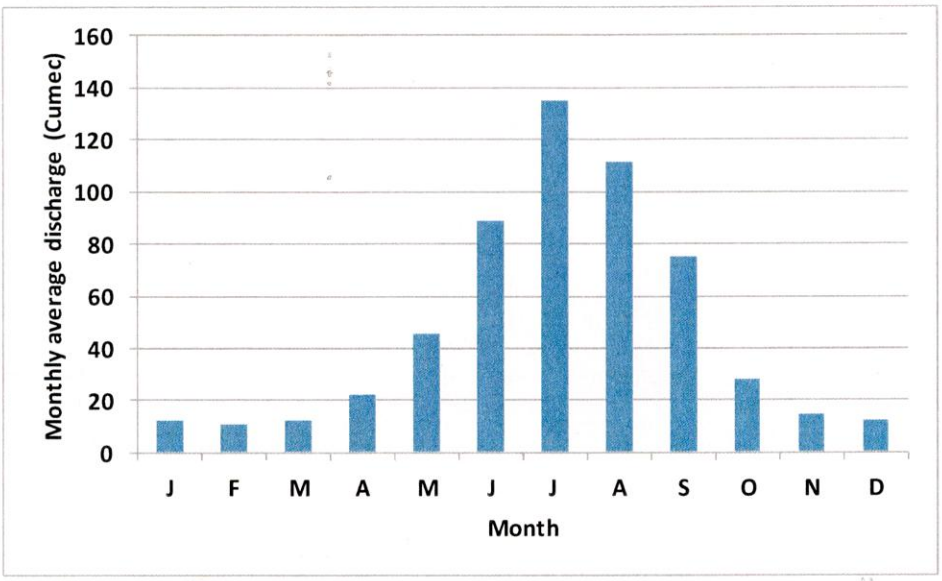


Table:1 Quarterly distribution of the annual flows of Satluj River at Bhakra Dam.

Period	**Average runoff (mm)**	**Contribution to annual flow (%)**
October-December	55.0	9.9
January-March	38.1	6.9
April-June	150.9	26.8
July-September	317.4	56.4
Year (October-September)	**561.4**	**100.0**

Methodology

Because of the rugged terrain and inaccessibility to the higher reaches, a poor snow gauge network exists in the high altitude region of the Himalayas, particularly where heavy snowfall is experienced. Therefore, assessment of snowfall over the whole basin becomes very difficult under such conditions. Thus, rainfall input along with streamflow and evapotranspiration information can be used for assessing the snow and glacier melt contribution. To assess the contribution of snow and glacier melt runoff into annual flows of the Satluj River, the following water balance approach was used:

$$SGC = Q - (R - E) \quad \ldots(1)$$

Where, SGC is the contribution from snow and glacier melt runoff, Q is the observed flow, R is the rainfall input to basin and E is the evapotranspiration loss from the basin.

Application of the water balance approach for estimating the average contribution of snow and glacier melt runoff to annual flows requires a long water budget period for the analysis. Table 2 represent the Snow cover area in the Indian part of the Satluj basin.

Table 2: Maximum and minimum snow-covered area in the Indian part of Satluj basin up to Bhakra Dam (total area: 22 305 km^2).

Year	Month	Snow-covered area as % of total basin	Month	Snow-covered % of total basin area
1986	March	64	September	15
1987	March	59	September	12
1988	March	71	September	35
1989	March	63	September	20
1990	March	70	September	17
1991	March	63	September	30
1992	March	72	September	16
1993	March	58	September	17
Average		**65**		**20.3**

Advantages of using C#

In the existing module, the programme is written in "Fortran". As FORTRAN is console based, it is very difficult for the users who doesn't know how to work on it.
But on the other hand, C# is both console based as well as application based, and is user friendly.

Requirements for Microsoft Visual Studio 2010

1. **Supported Operating System**

Windows 7, Windows Server 2003 R2 (32-Bit x86), Windows Server 2003 R2 x64 editions, Windows Server 2003 Service Pack 2, Windows Server 2008 R2, Windows Server 2008 Service Pack 2, Windows Vista Service Pack 2, Windows XP Service Pack 3

- Windows XP (x86) with Service Pack 3 - all editions except Starter Edition
- Windows Vista (x86 & x64) with Service Pack 2 - all editions except Starter Edition
- Windows 7 (x86 and x64)
- Windows Server 2003 (x86 & x64) with Service Pack 2 - all editions
- Users will need to install MSXML6 if not already present
- Windows Server 2003 R2 (x86 and x64) - all editions
- Windows Server 2008 (x86 and x64) with Service Pack 2 - all editions
- Windows Server 2008 R2 (x64) - all editions

2. **Supported Architectures:**

- 32-Bit (x86)
- 64-Bit (x64) (WOW)

3. Hardware Requirements:

- 1.6GHz or faster processor
- 1024 MB RAM (1.5 GB if running in a virtual machine)
- 5.5 GB of available hard-disk space
- 5400 RPM hard drive
- DirectX 9 capable video card running at 1024 x 768 or higher-resolution display
- DVD-ROM drive

Program Code

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO;
using Microsoft.VisualBasic;

namespace SnowMelt
   {
    public partial class Form1 : Form
    {

    public Form1()
    {
    InitializeComponent();
    }
    public void ZONEMELT(double[,] TMELT, double[,] SCAP, double NZ)
    {

        double[,] ZSMCUM = new double[1500, 10];
        double NYR = 0.0;
        NYR = 3;
        int N1, N2;

         N1 = 1;
         N2 = 365;

         double SUM = 0.0;

         for (int N = 1; N <= NYR; N++)
         {
          for (int J = 3; J <= NZ; J++)
          {
                SUM = 0.0;
                for (int I = N1; I <= N2; I++)
                {
                 SUM = SUM + TMELT[I, J];
                 ZSMCUM[I, J] = SUM;
                 //FileSystem.Print(17," ",N.ToString("F3")," ", J.ToString("F3")," ",
                 //  SCAP[I,J].ToString("F3")," ",TMELT[I,J].ToString("F3")," ",
                 //ZSMCUM[I,J].ToString("F3"));
```

```
                //L30:   continue;
            }
                // L20: continue;
            }
                N1 = N2 + 1;
                N2 = N1 + 364;
                //L10: continue;
            }


                // WRITE(17,1400)

                int K1 = 1;
                int K2 = 365;

        double NC = 0.0;
        for (double N = 1; N <= NYR; N++)
        {
        for (int I = K1; I <= K2; I++)
        {
              NC = NC + 1;
         FileSystem.Print(17, "  ", NC.ToString("F3"));
         for (int J = 1; J <= 4; J++)
         {
         FileSystem.Print(17, "  ", SCAP[I, J].ToString("F3"), "  ", ZSMCUM[I,
J].ToString("F3"), "  ", SCAP[I, J].ToString("F3"), "  ", ZSMCUM[I, J].ToString("F3"));
         }
         FileSystem.Print(17, "  ", N.ToString("F3"), "  ", I.ToString("F3"));
         for (int J = 4; J <= NZ; J++)
         FileSystem.Print(17, "  ", ZSMCUM[I, J].ToString("F3"), "  ", SCAP[I,
J].ToString("F3"));

                K1 = K2 + 1;
                K2 = K1 + 364;

        }
        }
        }


    public void TLAPSE(double NDAYS, int N, int NN, double[] T, double[] TLR, double
TSELEV, double[] ZELEV, double[,] TZ, int NZ)
        {
        double[,] TZZ = new double[1500, 10];
        double[,] DELT = new double[1500, 10];

        for (int J = 0; J <= N-1; J++)
        {
         for (int I = 0; I <= NDAYS-1; I++)
        {
```

```
            DELT[I, J] = (TLR[I] * (TSELEV - ZELEV[J])) / 100;
            TZZ[I, J] = T[I] + DELT[I, J];
            if (TZZ[I, J] < 0)
               TZZ[I, J] = 0.0;
            NZ = NN - N + J;
            TZ[I, NZ] = TZZ[I, J];
        }
        }
        }


    public void PPTDIST(int NDAYS, int N, int NN1, double[] P, double[] PDIST, double
PSELEV, double[] ZELEV, double[,] PP, int NZ)
        {

        double[,] DELP = new double[1500, 10];
        double[,] PZZ = new double[1500, 10];

        for (int J = 0; J <= N-1; J++)
        {
         for (int I = 0; I <= NDAYS-1; I++)
        {
            DELP[I, J] = (PDIST[I] * (PSELEV - ZELEV[J])) / 100;
            PZZ[I, J] = P[I] + DELP[I, J];
            if (PZZ[I, J] < 0)
               PZZ[I, J] = 0.0;
            NZ = NN1 - N + J;
            PP[I, NZ] = PZZ[I, J];
        }
        }
        }

    public void PCONT(double NDAYS, double NZ, double[,] IPR, double[,] AN, double[,]
PP, double[,] TMZ, double[,] TC, double[,] SCAP, double[,] PPP)
        {
        double[,] PRECIP = new double[1500, 10];
        for (int I = 1; I <= NDAYS; I++)

        for (int J = 1; J <= NZ; J++)
        {
            PPP[I, J] = 0.0;
        }

        for(int J=1; J<=NZ; J++)
        {
         double SNOW=0.0;
         double R=0.0;

         for(int I=1; I<=NDAYS;I++)
         {
```

```
double SNOW0=0.0;
double SNOW1=0.0;

if(IPR[I,J]==0)
{
  SNOW0=SCAP[I,J];
}


if(IPR[I,J]==1)
{
  SNOW1=SCAP[I,J];
}

R=TMZ[I,J]*AN[I,J];

PRECIP[I,J]=PP[I,J];

if(PRECIP[I,J]<0.00001)
  goto L12;
if(TMZ[I,J]>TC[I,J])
  goto L11;
SNOW=SNOW+PRECIP[I,J];

PPP[I,J]=0.0;
goto L12;

L11: PPP[I,J]=PRECIP[I,J];

L12:
if(SNOW<0.00001)
      goto L90;

if(TMZ[I,J]<0.0)
      goto L90;
double EP=0.0;
EP=R*(1-SNOW1);

if(R>SNOW)
  goto L13;
PPP[I,J]=PPP[I,J]+EP;
SNOW=SNOW-R;

goto L90;

L13: PPP[I,J]=PPP[I,J]+(SNOW*(1-SNOW1));
SNOW=0.0;

L90: PPP[I,J]=PPP[I,J]*(1-SNOW0);
}
```

```
//L80:    continue;

}
}

public void CHKDATA(double[,] X, int NZ, int NHYEAR, int[] IYR)
{
double[] NDAYSS = new double[12] { 7, 4, 7, 6, 7, 6, 7, 7, 6, 7, 6, 7 };

FileSystem.Print(16, "  ",NZ.ToString("F3"));

int IYEAR=0;
int IYEAR1=0;

//WRITE(16,100)


for(int J=1; J<=NZ; J++)
{
   int N1 = 0;
   int N2 = 0;
   int NC = 0;


for(int NY=1; NY<=NHYEAR; NY++)
{
      IYEAR1 = IYR[NY] - 1;



for(int M=11; M<=12; M++)
{
for(int K=1;K<=4; K++)
{
   if(N1==0)
        goto L222;
   if(NC>1 &&K==1)
        goto L444;

     N1=N1+8;
     N2=N2+8;
     goto L555;


     L444: N1=N2+1;
          N2=N1+7;

     L555:   IYEAR=IYR[NY];
```

```
            if(M==2&&K==4)
        {
            LEAPYR(IYEAR1,NDAYSS);
        }

            //FileSystem.Print(16, "  ", IYEAR1.ToString("F3")," ",
NDAYSS[2].ToString("F3"));

            if(K==4)
              N2=N2-8+Convert.ToInt32(NDAYSS[M]);

            NDAYSS[2]=4;
            goto L333;

          L222:   N1=1;
              N2=8;

          L333:   FileSystem.Print(16," ",J.ToString("F3")," ",IYEAR1.ToString("F3")," 
",M.ToString("F3")," ",K.ToString("F3")," ",N1.ToString("F3")," ",N2.ToString("F3"));
          for(int I=N1; I<=N2; I++)
          {
          FileSystem.Print(16," ",X[I,J].ToString("F3"));

          }
          /*
          L333:   FileSystem.Print(16," ",IYEAR1.ToString("F3")," ",J.ToString("F3")," 
",M.ToString("F3")," ",K.ToString("F3")," ",N1.ToString("F3")," ",N2.ToString("F3"));


          for(int I=N1; I<=N2; I++)
          {
          FileSystem.Print(16," ",X[I,J].ToString("F3"));

          }
          */

          }
          NC=NC+1;
          }


///page11///


          for(int M=1; M<=10; M++)
          {
          for(int K=1; K<=4; K++)
          {
              if(NC>1 && K==1)
                goto L666;
```

```
                    N1=N1+8;
                    N2=N2+8;

                    goto L777;


                L666:   N1=N2+1;
                        N2=N1+7;

                L777:   IYEAR=IYR[NY];


                    if(M==2 && K==4)
                       LEAPYR(IYEAR,NDAYSS);

                    //FileSystem.Print(16,"  ",IYEAR.ToString("F3")," 
",NDAYSS[2].ToString("F3"));

                    if(K==4)
                       N2=N2-8+Convert.ToInt32(NDAYSS[M]);
                    NDAYSS[2]=4;

                    FileSystem.Print(16,"  ",J.ToString("F3"),"  ",IYEAR.ToString("F3")," 
",M.ToString("F3"),"  ",K.ToString("F3"),"  ",N1.ToString("F3"),"  ",N2.ToString("F3"));
                    for(int I=N1; I<= N2; I++)


                    FileSystem.Print(16,"  ",X[I,J].ToString("F3"));

                    }

                    NC=NC+1;
                    }

                    // WRITE(16,700)
                    //L20:  continue;
                    }

                    //  L10: continue;
                    }
                    }

    public void LEAPYR(int iyear, double[] ndayss)
                    {
                    int Ipyr=0;
                    Ipyr = iyear % 4;
                    if (Ipyr == 0)
                    ndayss[2] = 5;
                    }
```

```
public void OPTIMIZE()
        {
   double[] TIME= new double[1500];
   double[] X = new double[5];
   double[] E = new double[5];
   double[,] V = new double[5,5];
   double[] SA = new double[5];
   double[] D = new double[5];
   double[] H = new double[5];
   double[] AL = new double[5];

   double[] PH = new double[5];
   double[,] B = new double[5,5];
   double[] BX = new double[5];
   double[] DA = new double[1];
   double[,] VV = new double[5,5];
   double[] EINT = new double[5];

   double[] UH = new double[1500];
   double[] QEST = new double[1500];
   double[] BFLOW = new double[1500];
   double NRUN=0.0;
  // double DUH = 0.0;
   double[] QESTR = new double[1500];


   double[] QESTS = new double[1500];
   double NS=0.0;
   double NR=0.0;
   double NB=0.0;
   double[] BASEIN = new double[1500];
   double[] QESTB = new double[1500];

   double[] QO = new double[1500];
   double[] QSFA = new double[1500];
   double[] QSCA = new double[1500];
   double[] AREAR = new double[1500];
   double[] AREAS = new double[1500];
   double KOUNT=0.0;

   int P=0;
   int PR=0;
   double BMAG = 0.0;

   double BBMAG = 0.0;
   double PW=0.0;

   double LC=0.0;
```

```
        FileSystem.FileOpen(7, "Opt8891.DAT", OpenMode.Output, OpenAccess.Write,
OpenShare.Shared, 1000);
        FileSystem.FileOpen(8, "OPTRES.OUT", OpenMode.Input, OpenAccess.Write,
OpenShare.Shared, 1000);


        double DLT=0.0;
        FileSystem.Input(7, ref DLT);

        // WRITE(*,*)NRUN

        for(int I=2; I<=NRUN;I++)
            TIME[I]=TIME[I-1]+DLT;

          double SUM1=0.0;
          double SUM2=0.0;
          double SUM3=0.0;


        for(int I=1; I<=NRUN; I++)
        {
            SUM1=SUM1+QO[I];
            SUM2=SUM2+QSFA[I];
            SUM3=SUM3+QSCA[I];
        }


       double M=0.0;
        int L=0;
        int LOOPY=0;
        int ND=0;
        int NDATA=0;
        double NSTEP=0.0;

        FileSystem.Input(7,ref M);
        FileSystem.Input(7,ref P);
        FileSystem.Input(7,ref L);
        FileSystem.Input(7,ref LOOPY);
        FileSystem.Input(7,ref PR);
        FileSystem.Input(7,ref ND);
        FileSystem.Input(7,ref NDATA);
        FileSystem.Input(7,ref NSTEP);

        FileSystem.Print(16, "  ", M.ToString("F3")," ",P.ToString("F3")."
",L.ToString("F3")," ",LOOPY.ToString("F3")," ",PR.ToString("F3")."
",ND.ToString("F3")," ",NDATA.ToString("F3")," ",NSTEP.ToString("F3")," ");

        //L10:
        for(int K=1; K<=P; K++)
        {
```

```
        FileSystem.Input(7, ref X[K]);


       //FileSystem.Print(16,"  ",X[K].ToString("F3"));
        //L100:   continue;
         }


         for(int J=1; J<=P; J++)
         {
             FileSystem.Input(7, ref E[J]);
             //FileSystem.Print(16, "  ", E[J].ToString("F3"));

         //L200:  continue;

         }

         FileSystem.Input(7,ref NR);
         FileSystem.Input(7,ref NS);
         FileSystem.Input(7,ref NB);

         // FileSystem.Print(16,"  ", NR.ToString("F3"),"  ",NS.ToString("F3"),"
",NB.ToString("F3"),"  ");

         //WRITE(8,013)

         if((ND-1)<0 || (ND-1)>0)
            goto L30;
         else if((ND-1)==0)
            goto L20;

         L20:
         for(int KA=1; KA<=NDATA; KA++)
         {
         FileSystem.Print(7,"  ",DA[KA].ToString("F3"));
        //L300:continue;

         }

         int LAP=0;

         L30: LAP=PR*1;
         int LOOP=0;
         int ISW=0;
         int INIT=0;
         KOUNT=0.0;
         double TERM=0.0;
         double DELY=0.0000001;
         double F1=0.0;
         double F0=0.0;
```

```
int NPAR=0;
NPAR=NDATA;
int N=0;
N=L;


for(int K=1; K<=L; K++)
{

//L40:  AL[K]=(CH(X,DA,N,NPAR,K)-CG(X,DA,NDATA,NPAR,K))*0.0001;
   AL[K] = (CH(X, DA, N, NPAR, K) - CG(X, DA, NDATA, NPAR, K)) * 0.0001;
}


for(int I=1; I<=P; I++)
{
for(int J=1; J<=P; J++)
{
      V[I,J]=0.0;
      if((I-J)<0 || (I-J>=0))
         goto L60;
      else if((I-J)==0)
         goto L61;

      L61: V[I,J]=1.0;
      L60: continue;

}
}

for(int KK=1; KK<=P; KK++)
{
EINT[KK]=E[KK];
//L65:    continue;
}

double FBEST=0.0;

L1000:
for(int J=1; J<=P; J++)
{
   if(NSTEP==0)
      E[J]=EINT[J];
   SA[J]=2.0;

   //L70: D[J]=0.0;
   D[J] = 0.0;
}

FBEST=F1;
```

```
L80:    int W=1;

if(INIT==0)
   goto L120;


L90:
for(int K=1; K<=P; K++)

{
X[K]=X[K]+E[W]*V[W,K];

}

for(int K=1; K<=L; K++)
     H[K]=F0;

//double[] XP,double DA,double NP, double NPAR
L120: F1 = F(X,DA,N,NPAR);
F1=M*F1;

if(ISW==0)
   F0=F1;


ISW=1;

double[] ABS= new double[1500];

if((ABS[Convert.ToInt32(FBEST-F1)]-DELY)<0)
     goto L122;

else if((ABS[Convert.ToInt32(FBEST-F1)]-DELY)>0)
      goto L125;


L122:    TERM=1.0;
   goto L450;


//L125: continue;
L125:
int Z=1;

double UC=0.0;
double XC=0.0;

L130:    XC=CX(X,DA,N,NPAR,Z);
```

```
   LC=CG(X,DA,N,NPAR,Z);
   UC=CH(X,DA,N,NPAR,Z);

if(XC<LC)
   goto L420;
if(XC>UC)

   goto L420;
if(F1<F0)
   goto L420;
if(XC<(LC+AL[Z]))
   goto L140;
if(XC>(UC-AL[Z]))
   goto L140;

H[Z]=F0;
goto L210;

//L140: continue;
L140:
double BW=0.0;

BW=AL[Z];

if(XC<LC || UC<XC)
   goto L150;
if(LC<XC && XC<(LC+BW))
   goto L160;
if((UC-BW)<XC && XC<UC)
   goto L170;

L150:  PH[Z]=1.0;
goto L190;

L160:   PW=(LC+BW-XC)/BW;
goto L180;

L170:   PW=(XC-UC+BW)/BW;

L180:   PH[Z]=1.0-3.0*PW+4.0*PW*PW-2.0*PW*PW*PW;

L190:   F1= H[Z]+(F1-H[Z])*PH[Z];

// L210:   continue;
L210:
 if(Z==L)
   goto L220;
 Z+=1;
 goto L130;
```

```
L220:    INIT=1;

if(F1< F0)
   goto L420;

D[W]=D[W]+E[W];
E[W]=3.0*E[W];
F0=F1;

if(SA[W]>1.5)
   SA[W]=1.0;

L230:
   for(int JJ=1; JJ<=P; JJ++)
   {
      if(SA[JJ]>0.5)
         goto L440;
      //L240:    continue;

   }

 for(int R=1; R<=P; R++)
 {

 for (int C = 1; C <= P; C++)
  VV[C, R] = 0.0;
 }

 int KR=0;

 for (int R = 1; R <= P; R++)
{
KR = R;
for (int C = 1; C <= P; C++)
{    for (int K = KR; K <= P; K++)
      VV[R, C] = D[K] * V[K, C] + VV[R, C];
       B[R, C] = VV[R, C];
}

}


for(int C=1; C<=P; C++)
{
BMAG = BMAG + B[1, C] * B[1, C];

//L280: continue;

}
```

```
  BMAG = Math.Sqrt(BMAG);

  BX[1] = BMAG;

for (int C = 1; C <= P; C++)
   V[1, C] = B[1, C] / BMAG;

int IR = 0;
double SUMAV = 0.0;
double SUMVM = 0.0;

for(int R=2; R<=P; R++)
{
   IR = R - 1;
for(int C=1; C<=P; C++)
{
      SUMVM = 0.0;

 for (int KK = 1; KK <= IR;KK++ )
{
      SUMAV = 0.0;

 for(int KJ=1; KJ<=P;KJ++)
{
      SUMAV = SUMAV + VV[R, KJ] * V[KK, KJ];
 }

      SUMVM = SUMAV * V[KK, C] + SUMVM;
 }
      B[R, C] = VV[R, C] - SUMVM;

 }
 }


for(int R=2; R<=2; R++)

{
   BBMAG = 0.0;
   for (int K = 1; K <= P; K++)
      BBMAG = BBMAG + B[R, K] * B[R, K];

   BBMAG = Math.Sqrt(BBMAG);
   for (int C = 1; C <= P; C++)
      V[R, C] = B[R, C] / BBMAG;
}

LOOP = LOOP + 1;
LAP = LAP + 1;
```

```
            if (LAP == PR)
               goto L450;

            goto L1000;

         L420:
            if (INIT == 0)
               goto L450;

         for (int IX = 1; IX <= P; IX++)
            X[IX] = X[IX] - E[W] * V[W, IX];

         E[W] = -05 * E[W];

         if (SA[W] < 1.5)
            SA[W] = 0.0;
         goto L230;

         //L440: continue;
            L440:

            if (W == P)
               goto L80;

            W =W+ 1;
               goto L90;

           // L450: WRITE(8,003)
            L450:


            FileSystem.Print(8, "  ", LOOP.ToString("F3"), "  ", F0.ToString("F3"), "  ",
BMAG.ToString("F3"), "  ", BBMAG.ToString("F3"));

            FileSystem.Print(8, "  ", KOUNT.ToString("F3"));

            //WRITE(8,005)


            for(int JM=1; JM<=P; JM++)
            FileSystem.Print(8, "  ", JM.ToString("F3"), X[JM]);

            LAP = 0;

            if (INIT == 0)
               goto L470;
            if (TERM == 1.0)
               goto L480;
            if (LOOP > LOOPY)
```

```
            goto L480;

          goto L1000;
          L470:
         // L470:    WRITE(8,007)

          //L480: continue;
          L480:
          //L490: WRITE(8,008)


          for (int J = 1; J <= P; J++)
          for (int I = 1; I <= P; I++ )
          FileSystem.Print(8, "  ", J.ToString("F3"), "  ", I.ToString("F3"), "  ", V[J,
I].ToString("F3"));
          // WRITE(8,011)

          for (int J = 1; J <= P; J++ )
          FileSystem.Print(8, "  ", J.ToString("F3"), "  ", E[J].ToString("F3"));

          }

       public double CX(double[] X, double[] DA, int N, int NPAR, int K)
         {

          return X[K];

          }

       public double CG(double[] X, double[] DA, int N, int NPAR, int K)
         {
          double A = 0.0;
          switch(K)
          {
             case 1:
             case 2:
             case 3:
             case 4: A= 0.0; break;

             case 5: A= 80.0; break;

          }
          return A;
          }

       public double CH(double[] X,double[] DA,int N, int NPAR, int K)
          {
          double A = 0.0;
          switch (K)
          {
```

```
            case 1: A= 1.0; break;

            case 2: A= 0.15; break;

            case 3: A= 1.0; break;


            case 4: A= 0.15; break;

            case 5: A= 120.0; break;


        }
        return A;
        }

    public double F(double[] XP, double[] DA, double NP, double NPAR)
        {
        double[] QQSFA=new double[1500];
        double[] QQSCA = new double[1500];
        double[] QQBASE = new double[1500];
        double[] QBARB = new double[1500];


        double[] UH = new double[1500];
        double[] QEST = new double[1500];
        double[] BFLOW = new double[1500];

        int NRUN = 0;
        //double DUH = 0.0;
        double DLT = 0.0;

        double[] QESTR = new double[1500];
        double[] QESTS = new double[1500];

        double NS = 0.0;
        double NR = 0.0;
        double NB = 0.0;

        double[] BASEIN = new double[1500];
        double[] QESTB = new double[1500];
        double[] QO = new double[1500];
        double[] QSFA = new double[1500];
        double[] QSCA = new double[1500];
        double[] AREAR = new double[1500];
        double[] AREAS = new double[1500];

        double KOUNT = 0.0;
        double AR = 0.0;
        double BR = 0.0;
```

```
        double AS = 0.0;
        double BS = 0.0;
        double AKB = 0.0;


        AR=XP[1];
        BR=XP[2];
        AS = XP[3];
        BS = XP[4];
        // AKB=XP[5];

        AKB = 110.0;

        if (AR < 0.0)
           goto L111;
        if (BR < 0.0)
           goto L111;
        if (AS < 0.0)
           goto L111;
        if (BS < 0.0)

           goto L111;
        if (AKB < 0.0)
           goto L111;









        for(int I=1; I<= NRUN; I++)
        {
           QQSFA[I] = QSFA[I];
           QQSCA[I] = QSCA[I];
           QQBASE[I] = BASEIN[I];
           //QQBASE[I] = 0.50 * BASEIN[I];

           //FileSystem.Print(16, "  ", I.ToString("F3"), "  ", QQSFA[I].ToString("F3"), "  ",
QQSCA[I].ToString("F3"), "  ", QQBASE[I].ToString("F3"));

        }

        double AKS = 0.0;
        double CS0 = 0.0;
        double CS1 = 0.0;
        double CS2 = 0.0;

        FileSystem.Print(8, "  ", AKS.ToString("F3"), "  ", CS0.ToString("F3"), "  ",
CS1.ToString("F3"), "  ", CS2.ToString("F3"));
```

```
QESTR[1] = 0.01;
QESTS[1] = 0.01;
QEST[1] = QO[1];
QESTB[1] = QO[1];

double AKR = 0.0;
double CR0=0.0;
double CR1=0.0;
double CR2=0.0;

for(int I=1; I<=NR; I++)
{
  // WRITE(*,*)I,AKR,CR0,CR1,CR2

  for(int N=2; N<=NRUN; N++)
  {
    AKR = AR * Math.Pow(AREAR[N],BR);
    CR0=(0.5*DLT)/(AKR+0.5*DLT);
    CR1 = CR0;
    CR2=1-(CR0+CR1);

    if (CR2 < 0)

      return 1.0E+10;
    if(I==1)
      QESTR[N]=2*CR1*QQSFA[N]+CR2*QESTR[N-1];
    else
      QESTR[N]=CR0*QQSFA[N-1]+CR1*QQSFA[N]+CR2*QESTR[N-1];

    continue;
  }

  for(int K=2; K<=NRUN; K++)
    QQSFA[K]=QESTR[K];
  continue;
  }


  for(int I=1; I<=NS; I++)
  {
  //WRITE(*,*) I,AKS,CS0,CS1,CS2

  for (int N = 2; N <= NRUN; N++)
  {
  AKS = AS * Math.Pow(AREAS[N],BS);

      if(AKS==0)
    {
```

```
            CS0 = 0;
            CS1 = 0;
            CS2 = 1.0;
            goto L120;
         }

      CS0 = (0.5 * DLT) / (AKS + 0.5 * DLT);
      CS1 = CS0;
      CS2 = 1 - (CS0 + CS1);

//L120: continue;
      L120:
if (CS2 < 0)
      return 1.0E+10;

QESTS[N] = CR0 * QQSCA[N - 1] + CS1 * QQSCA[N] + CS2 * QESTS[N - 1];

if (QQSCA[N] == 0 && QQSCA[N - 1] == 0)


      QESTS[N] = 0;

continue;
      }

      for (int K = 2; K <= NRUN; K++)
         QQSCA[K] = QESTS[K];
}


double CB0 = 0.0;
double CB1 = 0.0;
double CB2 = 0.0;
for(int I=1; I<=NB; I++)
{
      //WRITE(*,*)I,AKB,CB0,CB1,CB2

      for(int N=2; N<=NRUN; N++)
      {
         CB0=(0.5*DLT)/(AKB+0.5*DLT);
         CB1 = CB0;
         CB2 = 1 - (CB0 + CB1);

         if (CB2 < 0)
            return 1.0E+10;

         QBARB[N] = (QQBASE[N] + QQBASE[N - 1]) / 2;

         if (QBARB[N] < 0)
            QBARB[N] = 0.0;
```

```
            QESTB[N] = 2 * CB1 * QBARB[N] + CB2 * QESTB[N - 1];
            continue;
          }
          continue;
        }


        for (int I = 2; I <= NRUN; I++)
          QEST[I] = QESTR[I] + QESTS[I] + QESTB[I];


        int NNN = 0;

        for (int I = 2; I <= NRUN; I++)


          NNN = I;

        double SUM;

        L111: SUM = 0.0;


        for (int I = 1; I <= NRUN; I++)
          SUM = SUM + Math.Pow((QO[I] - QEST[I]),2);

        double DD = SUM / 1000;

       // FileSystem.Print(16, "  ",F.ToString("F3"),"  ", AR.ToString("F3"), "  ",
BR.ToString("F3"), "  ", AS.ToString("F3"), "  ", BS.ToString("F3"), "  ",
AKB.ToString("F3"));
        FileSystem.Print(16, "  ", AR.ToString("F3"), "  ", BR.ToString("F3"), "  ",
AS.ToString("F3"), "  ", BS.ToString("F3"), "  ", AKB.ToString("F3"));
        KOUNT = KOUNT + 1;
        return DD;
        }

      public double ROUTE()
        {
        double[] XP = new double[5];
        double[] QQSFA= new double[1500];
        double[] QQSCA = new double[1500];
        double[] QQBASE= new double[1500];
        double[] QBARB = new double[1500];

        double[] UH = new double[1500];
        double[] QEST = new double[1500];
        double[] BFLOW = new double[1500];
```

```
        int NRUN = 0;
        //double DUH = 0.0;
        double DLT = 0.0;

        double[] QESTR = new double[1500];
        double[] QESTS = new double[1500];

        double NS = 0.0;
        double NR = 0.0;
        double NB = 0.0;

        double[] BASEIN = new double[1500];

        double[] QESTB = new double[1500];

        double[] QO = new double[1500];
        double[] QSFA = new double[1500];
        double[] QSCA = new double[1500];
        double[] AREAR = new double[1500];
        double[] AREAS = new double[1500];

        double AR = 0.0;
        double BR = 0.0;
        double AS = 0.0;
        double BS = 0.0;
        double AKB = 0.0;


        DLT = 1.0;

        FileSystem.Input(9, ref NR);
        FileSystem.Input(9, ref NS);
        FileSystem.Input(9, ref NB);

        FileSystem.Print(15, "  ", NR.ToString("F3"), "  ", NS.ToString("F3"), "  ",
NB.ToString("F3"));

        FileSystem.Input(9, ref AR);
        FileSystem.Input(9, ref BR);
        FileSystem.Input(9, ref AS);
        FileSystem.Input(9, ref BS);
        FileSystem.Input(9, ref AKB);

        FileSystem.Print(15, "  ", AR.ToString("F3"), "  ", BR.ToString("F3"), "  ",
AS.ToString("F3"), "  ", BS.ToString("F3"), "  ", AKB.ToString("F3"));


        for(int I=1; I<= NRUN; I++)
        {
          QQSFA[I] = QSFA[I];
```

```
            QQSCA[I] = QSCA[I];
            QQBASE[I] = BASEIN[I];

            //QQBASE[I] = 0.50 * BASEIN[I];
          }

          double AKS = 0.0;
          double CS0 = 0.0;


          double CS1 = 0.0;
          double CS2 = 0.0;

          //FileSystem.Print(8, "  ", AKS.ToString("F3"), "  ", CS0.ToString("F3"), "  ",
CS1.ToString("F3"), "  ", CS2.ToString("F3"));

          QESTR[1] = 0.01;
          QESTS[1] = 0.01;
          QEST[1] = QO[1];
          QESTB[1] = QO[1];

          double AKR=0.0;
          double CR0=0.0;
          double CR1=0.0;
          double CR2=0.0;

          for(int I=1; I<= NR; I++)
          {
          for(int N=2; N<=NRUN; N++)
          {
          AKR = AR * Math.Pow(AREAR[N],BR);
          CR0=(0.5*DLT)/(AKR+0.5*DLT);
          CR1 = CR0;
          CR2 = 1 - (CR0 + CR1);
              if(CR2<0)
                return 1.0E+10;
              if (I == 1)
                QESTR[N] = 2 * CR1 * QQSFA[N] + CR2 * QESTR[N - 1];
              else
                QESTR[N] = CR0 * QQSFA[N - 1] + CR1 * QQSFA[N] + CR2 * QESTR[N - 1];
            //  L101: continue;
              }
              for(int K=2; K<=NRUN; K++)
              {
                QQSFA[K] = QESTR[K];

              }
            //L102:    continue;
              }
```

```
for(int I=1; I<= NS; I++)
{
for(int N=2; N<= NRUN; N++)
{

   //if (AREAS[N] < 0)
   //  AKS = 0;
   //if (AREAS[N] > 0)
   //  AKS = AS * Math.Pow(AREAS[N],BS);

   AKS = AS * Math.Pow(AREAS[N],BS);

   //AKS = AS * Math.Pow(QESTS[N - 1],BS);

   if(AKS==0)
   {
      CS0 = 0;
      CS1 = 0;
      CS2 = 1.0;

      goto L120;

   }

   CS0 = (0.5 * DLT) / (AKS + 0.5 * DLT);
   CS1 = CS0;
   CS2 = 1 - (CS0 + CS1);
//L120: continue;

L120:
if(CS2<0)
   return 1.0E+10;

QESTS[N] = CS0 * QQSCA[N - 1] + CS1 * QQSCA[N] + CS2 * QESTS[N - 1];

if (QQSCA[N] == 0 && QQSCA[N - 1] == 0)
   QESTS[N] = 0;
//L202: continue;
}

for (int K = 2; K <= NRUN; K++)
   QQSCA[K] = QESTS[K];
//L201: continue;
}

double CB0=0.0;
double CB1=0.0;
double CB2=0.0;
```

```
for(int I=1; I<=NB; I++)
{
for(int N=2; N<= NRUN; N++)
{
  CB0=(0.5*DLT)/(AKB+0.5*DLT);
    CB1 = CB0;
    CB2 = 1 - (CB0 + CB1);
    if (CB2 < 0)
      return 1.0E+10;

    QBARB[N] = (QQBASE[N] + QQBASE[N - 1]) / 2;

    if (QBARB[N] < 0)
      QBARB[N] = 0.0;

    QESTB[N] = 2 * CB1 * QBARB[N] * CB2 * QESTB[N - 1];
  //  L302: continue;
  }
  //L301: continue;
  }

for (int I = 2; I <= NRUN; I++)
  QEST[I] = QESTR[I] + QESTS[I] + QESTB[I];

double SUM = 0.0;
for (int I = 1; I <= NRUN; I++)
  SUM = SUM + Math.Pow((QO[I] - QEST[I]),2);

double FV = 0.0;
double KOUNT = 0.0;
FV = SUM / 10000;
//WRITE(*,*)F

KOUNT = KOUNT + 1;
return FV;
}

public void QRTANL(double[] QO, double[] QEST, double[] QESTR, double[] QESTS,
double[] QESTB, double[] EVPVOL)
{
//WRITE(18,1100)

int NYR = 3;
int N1 = 1;
int N2 = 120;

double YQO2 = 0.0;

double YQEST2 = 0.0;
double YQESTR2 = 0.0;
```

```
            double YQESTS2 = 0.0;
            double YQESTB2 = 0.0;
            double YEVPVOL2 = 0.0;
            int NJ = 0;
            for(int N=1; N<=NYR; N++)
            {
              if (N > 1)
                 N1 = N2 + 1;
              if (N > 1)
                 N2 = N1 + 119;

              double YQO1 = 0.0;
              double YQEST1 = 0.0;
              double YQESTR1 = 0.0;
              double YQESTS1 = 0.0;
              double YQESTB1 = 0.0;
              double YEVPVOL1 = 0.0;


              for(int J=1; J<=4; J++)
              {
                 double YQO = 0.0;
                 double YQEST = 0.0;
                 double YQESTR = 0.0;
                 double YQESTS = 0.0;
                 double YQESTB = 0.0;
                 double YEVPVOL = 0.0;

                 for(int I=N1; I<= N2;I++)
                 {
                    YQO = YQO + QO[I];
                    YQEST = YQEST + QEST[I];
                    YQESTR = YQESTR + QESTR[I];
                    YQESTS = YQESTS + QESTS[I];
                    YQESTB = YQESTB + QESTB[I];
                    YEVPVOL += EVPVOL[I];
                    //FileSystem.Print(18, "  ", N.ToString("F3"), "  ", J.ToString("F3"), "  ",
YQO.ToString("F3"), "  ", YQEST.ToString("F3"), "  ", YQESTR.ToString("F3"), "  ",
YQESTS.ToString("F3"), "  ", YQESTB.ToString("F3"), "  ", YEVPVOL.ToString("F3"));
                 //L30: continue;

                 }


                 FileSystem.Print(18, "  ", N.ToString("F3"), "  ", J.ToString("F3"), "  ",
YQO.ToString("F3"), "  ", YQEST.ToString("F3"), "  ", YQESTR.ToString("F3"), "  ",
YQESTS.ToString("F3"), "  ", YQESTB.ToString("F3"), "  ", YEVPVOL.ToString("F3"));

                 N1 = N2 + 1;
                 //N2 = N1 + 91;
```

```
            if (J == 1)
               N2 = N1 + 91;
            if (J == 2)
               N1 = N1 + 91;
            if (J == 3)
               N2 = N2 + 60;

            YQO1 = YQO1 + YQO;
            YQEST1 += YQEST;
            YQESTR1 += YQESTR;
            YQESTS1 += YQESTS;
            YQESTB1 += YQESTB;
            YEVPVOL1 += YEVPVOL;
         //L20: continue;

         }

         //WRITE(18,1700)


         FileSystem.Print(18, "  ", N.ToString("F3"), "  ", NJ.ToString("F3"), "  ",
YQO1.ToString("F3"), "  ", YQEST1.ToString("F3"), "  ", YQESTR1.ToString("F3"), "  ",
YQESTB1.ToString("F3"), "  ", YEVPVOL1.ToString("F3"));

         YQO2 += YQO1;
         YQEST2 += YQEST1;
         YQESTR2 += YQESTR1;
         YQESTS2 += YQESTS1;
         YQESTB2 += YQESTB1;
         YEVPVOL2 += YEVPVOL1;

         //WRITE(18,1700)

      //L10: continue;
      }

      //WRITE(18,1700)
      //WRITE(18,1700)
      //WRITE(18,1700)



      FileSystem.Print(18, "  ", NYR.ToString("F3"), "  ", NJ.ToString("F3"), "  ",
YQO2.ToString("F3"), "  ", YQEST2.ToString("F3"), "  ", YQESTR2.ToString("F3"), "  ",
YQESTS2.ToString("F3"), "  ", YQESTB2.ToString("F3"), "  ", YEVPVOL2.ToString("F3"));
      //WRITE(18,1600)


   }
```

```
public void EFF(double[] QC, double[] QO, int NDAYS)
{
   double[] RA = new double[1500];
   double[] RB = new double[1500];

   double SUMA = 0.0;
   double SUMB = 0.0;


   for(int I=1; I<= NDAYS; I++)
   {
      SUMA += QC[I];
      SUMB += QO[I];

   }

   double AVGQC = 0.0;
   double AVGQO=0.0;
   AVGQC = SUMA / Convert.ToDouble(NDAYS);
   AVGQO = SUMB / Convert.ToDouble(NDAYS);

   double SUMC = 0.0;
   double SUMD = 0.0;
   double SUME = 0.0;
   double SUMF = 0.0;
   double SUMG = 0.0;
   double SUMH = 0.0;

   for(int I=1; I<= NDAYS; I++)
   {
      SUMC = SUMC + Math.Pow((QC[I] - AVGQC),2);
      SUMD = SUMD + Math.Pow((QO[I] - AVGQO) ,2);
      SUME = SUME + Math.Pow((QC[I] - AVGQC) ,3);
      SUMF = SUMF + Math.Pow((QO[I] - AVGQO) ,3);
      SUMG = SUMG + Math.Pow((QC[I] - AVGQC) ,4);
      SUMH = SUMH + Math.Pow((QO[I] - AVGQO) ,4);

   }

   double STDQC = 0.0;
   double STDQO = 0.0;

   STDQC = Math.Pow((SUMC / Convert.ToDouble(NDAYS - 1)),0.5);
   STDQO = Math.Pow((SUMD / Convert.ToDouble(NDAYS - 1)),0.5);


   double CSQC = 0.0;
   double CSQO = 0.0;
```

```
            FileSystem.Input(7, ref X[K]);


            //FileSystem.Print(16,"  ",X[K].ToString("F3"));
            //L100:   continue;
            }


            for(int J=1; J<=P; J++)
            {
                FileSystem.Input(7, ref E[J]);
                //FileSystem.Print(16, "  ", E[J].ToString("F3"));

            //L200:  continue;

            }

            FileSystem.Input(7,ref NR);
            FileSystem.Input(7,ref NS);
            FileSystem.Input(7,ref NB);

            // FileSystem.Print(16,"  ", NR.ToString("F3"),"  ",NS.ToString("F3"),"
",NB.ToString("F3"),"  ");

            //WRITE(8,013)

            if((ND-1)<0 || (ND-1)>0)
               goto L30;
            else if((ND-1)==0)
               goto L20;

            L20:
            for(int KA=1; KA<=NDATA; KA++)
            {
            FileSystem.Print(7,"  ",DA[KA].ToString("F3"));
            //L300:continue;

            }

            int LAP=0;

            L30: LAP=PR*1;
            int LOOP=0;
            int ISW=0;
            int INIT=0;
            KOUNT=0.0;
            double TERM=0.0;
            double DELY=0.0000001;
            double F1=0.0;
            double F0=0.0;
```

```
        double CKQC=0.0;
        double CKQO=0.0;
        CSQC=(NDAYS*SUME)/((Convert.ToDouble(NDAYS-
1))*(Convert.ToDouble(NDAYS)-2)*(Math.Pow(STDQC,3)));
        CSQO=(NDAYS*SUMF)/((Convert.ToDouble(NDAYS-
1))*(Convert.ToDouble(NDAYS)-2)*(Math.Pow(STDQO,3)));
        CKQC=((Math.Pow(NDAYS,2))*(SUMG))/((Convert.ToDouble(NDAYS)-
1)*(Convert.ToDouble(NDAYS)-2)*(Convert.ToDouble(NDAYS)-
3)*(Math.Pow(STDQC,4)));
        CKQO=((Math.Pow(NDAYS,2))*(SUMH))/((Convert.ToDouble(NDAYS)-
1)*(Convert.ToDouble(NDAYS)-2)*(Convert.ToDouble(NDAYS)-
3)*(Math.Pow(STDQO,4)));

        double SUM3=0.0;
        double SUM4=0.0;
        double SUM5=0.0;

        for(int I=1; I<= NDAYS; I++)
        {
          SUM3=SUM3+QC[I]*86400;
          SUM4=SUM4+QO[I]*86400;
          SUM5=SUM5+QO[I];

        }

        FileSystem.Print(15,"  ", SUM3.ToString("F3"),"  ", SUM4.ToString("F3"));


        double DV=0.0;
        DV=((SUM3-SUM4)/(SUM4))*100.0;
        FileSystem.Print(15,"  ",DV.ToString("F3"));
        //WRITE(*,177) DV


        double AVG4=0.0;



        double SUM6=0.0;
        double SUM7=0.0;




        for(int I=1; I<= NDAYS; I++)
        {
          RA[I]=Math.Pow((QO[I]-QC[I]),2);
          RB[I]=Math.Pow((QO[I]-AVG4),2);

          SUM6+=RA[I];
```

```
            SUM7+=RB[I];

        }

        double R1=0.0;
        double R2=0.0;

        R1=SUM6/SUM7;
        R2=1.0-R1;

        FileSystem.Print(15,"  ",R2.ToString("F3"));
        //WRITE(*,166)R2

        double SSE=0.0;
        double AMSE=0.0;
        double RMSE1=0.0;
        double RMSE2=0.0;
        double SSE1=0.0;
        double RMSE=0.0;
        SSE=SUM6*86.4*86.4/(5278.0*5278.0);
        SSE1=Math.Sqrt(SSE);
        AMSE=SSE/Convert.ToDouble(NDAYS);
        RMSE1=Math.Sqrt(SUM6/Convert.ToDouble(NDAYS));
        RMSE2=SUMB/Convert.ToDouble(NDAYS);
        RMSE=RMSE1/RMSE2;

        FileSystem.Print(15,"  ", SSE.ToString("F3"),"  ",AMSE.ToString("F3"),"
",SSE1.ToString("F3"),"  ",RMSE.ToString("F3"));

        ////PAGE19

        FileSystem.Print(15,"  ",AVGQC.ToString("F3"),"  ",AVGQO.ToString("F3"));

        FileSystem.Print(15,"  ",STDQC.ToString("F3"),"  ",STDQO.ToString("F3"));


        FileSystem.Print(15,"  ",CSQC.ToString("F3"),"  ",CSQO.ToString("F3"));

        FileSystem.Print(15,"  ",CKQC.ToString("F3"),"  ",CKQO.ToString("F3"));
    }



    public void MONQRTANL(int NHYEAR, int[] IYR, double[] QO, double[] QEST,
double[] QESTR, double[] QESTS, double[] QESTB, double[] EVP)
    {
        double[] EVPVOL = new double[1500];
        int[] NDAYSS = new int[12] {31,28,31,30,31,30,31,31,30,31,30,31};
        double[,] TMQO = new double[10,12];
        double[,] TMQEST = new double[10,12];
```

```
double[,] TMQESTR = new double[10,12];
double[,] TMQESTS = new double[10,12];
double[,] TMQESTB = new double[10,12];
double[,] TMEVP = new double[10,12];

double[] QRTQO1 = new double[10];
double[] QRTQO2 = new double[10];
double[] QRTQO3 = new double[10];
double[] QRTQO4 = new double[10];
double[] QRTQO5 = new double[10];

double[] QRTQEST1 = new double[10];
double[] QRTQEST2 = new double[10];
double[] QRTQEST3 = new double[10];
double[] QRTQEST4 = new double[10];
double[] QRTQEST5 = new double[10];


double[] QRTQESTR1 = new double[10];
double[] QRTQESTR2 = new double[10];
double[] QRTQESTR3 = new double[10];
double[] QRTQESTR4 = new double[10];
double[] QRTQESTR5 = new double[10];

double[] QRTQESTS1 = new double[10];
double[] QRTQESTS2 = new double[10];
double[] QRTQESTS3 = new double[10];
double[] QRTQESTS4 = new double[10];
double[] QRTQESTS5 = new double[10];

double[] QRTQESTB1 = new double[10];



double[] QRTQESTB2 = new double[10];
double[] QRTQESTB3 = new double[10];
double[] QRTQESTB4 = new double[10];
double[] QRTQESTB5 = new double[10];

double[] QRTEVP1 = new double[10];
double[] QRTEVP2 = new double[10];
double[] QRTEVP3 = new double[10];
double[] QRTEVP4 = new double[10];
double[] QRTEVP5 = new double[10];

double[] SQOMEAN = new double[12];
double[] SQESTMEAN = new double[12];
double[] SQESTRMEAN = new double[12];
double[] SQESTSMEAN = new double[12];
```

```
double[] SQESTBMEAN = new double[12];
double[] SEVPMEAN = new double[12];
double[] IY = new double[10];

IY[1] = IYR[1] - 1;
IY[2] = IYR[2] - 1;
IY[3] = IYR[3] - 1;

int IYEAR=0;

int NC = 0;
int ND = 0;
for(int I=1; I<=NHYEAR; I++)
{
   for(int J=11; J<=12; J++)
   {
      ND = NDAYSS[J];

      double SUMQO1 = 0.0;
      double SUMQEST1 = 0.0;
      double SUMQESTR1 = 0.0;
      double SUMQESTS1 = 0.0;
      double SUMQESTB1 = 0.0;
      double SUMEVP1 = 0.0;

      for(int K=1; K<=ND; K++)
      {
         NC += 1;
         SUMQO1 += QO[NC];
         SUMQEST1 += QEST[NC];

         SUMQESTR1 += QESTR[NC];
         SUMQESTS1 += QESTS[NC];
         SUMQESTB1 += QESTB[NC];
         SUMEVP1 += EVP[NC];
    //  L30: continue;
      }

      TMQO[I, J] = SUMQO1;
      TMQEST[I, J] = SUMQEST1;
      TMQESTR[I, J] = SUMQESTR1;
      TMQESTB[I, J] = SUMQESTB1;
      TMEVP[I, J] = SUMEVP1;

   //L20: continue;
   }

   for(int J=1; J<=10; J++)
   {
      ND = NDAYSS[J];
```

```
            IYEAR = IYR[J];

            if (J == 2)
               LEAPYR2(IYEAR, ND);

            double SUMQO2 = 0.0;
            double SUMQEST2 = 0.0;
            double SUMQESTR2 = 0.0;
            double SUMQESTS2 = 0.0;
            double SUMQESTB2 = 0.0;
            double SUMEVP2 = 0.0;

            for(int K=1; K<=ND; K++)
            {
               NC = NC + 1;
               SUMQO2 += QO[NC];
               SUMQEST2 += QEST[NC];
               SUMQESTR2 += QESTR[NC];
               SUMQESTS2 += QESTS[NC];
               SUMQESTB2 += QESTB[NC];
               SUMEVP2 += EVP[NC];

            //L50: continue;

            }



            TMQO[I, J] = SUMQO2;
            TMQEST[I, J] = SUMQEST2;
            TMQESTR[I, J] = SUMQESTR2;
            TMQESTS[I, J] = SUMQESTS2;
            TMQESTB[I, J] = SUMQESTB2;
            TMEVP[I, J] = SUMEVP2;
          //  L40: continue;
            }
         //L10:    continue;
         }


         //WRITE(18,150)

         for(int I=1; I<= NHYEAR; I++)
         {

            for(int J=11; J<=12; J++)
            {
               FileSystem.Print(18," ",IY[I].ToString("F3")," ",IYR[I].ToString("F3")," ",J.ToString("F3")," ",TMQO[I,J].ToString("F3")," ",TMQEST[I,J].ToString("F3"),"
```

```
",TMQESTR[I,J].ToString("F3")," ",TMQESTS[I,J].ToString("F3"),"
",TMQESTB[I,J].ToString("F3")," ",TMEVP[I,J].ToString("F3"));
            }

            for(int J=1; J<=10; J++)
            {
              FileSystem.Print(18, " ",IY[I].ToString("F3"), " ", IYR[I].ToString("F3"), " ",
J.ToString("F3"), " ", TMQO[I, J].ToString("F3"), " ", TMQEST[I, J].ToString("F3"), " ",
TMQESTR[I, J].ToString("F3"), " ", TMQESTS[I, J].ToString("F3"), " ", TMQESTB[I,
J].ToString("F3"), " ", TMEVP[I, J].ToString("F3"));
            }

            //WRITE(18,199)
        }

        for(int I=1; I<=NHYEAR; I++)
        {
            QRTQO1[I] = TMQO[I, 11] + TMQO[I, 12] + TMQO[I, 1] + TMQO[I, 2];
            QRTQO2[I] = TMQO[I, 3] + TMQO[I, 4] + TMQO[I, 5];
            QRTQO3[I] = TMQO[I, 6] + TMQO[I, 7] + TMQO[I, 8];
            QRTQO4[I] = TMQO[I, 9] + TMQO[I, 10];
            QRTQO5[I] = QRTQO1[I] + QRTQO2[I] + QRTQO3[I] + QRTQO4[I];




            QRTQEST1[I] = TMQEST[I, 11] + TMQEST[I, 12] + TMQEST[I, 1] + TMQEST[I,
2];
            QRTQEST2[I] = TMQEST[I, 3] + TMQEST[I, 4] + TMQEST[I, 5];
            QRTQEST3[I] = TMQEST[I, 6] + TMQEST[I, 7] + TMQEST[I, 8];
            QRTQEST4[I] = TMQEST[I, 9] + TMQEST[I, 10];
            QRTQEST5[I] = QRTQEST1[I] + QRTQEST2[I] + QRTQEST3[I] +
QRTQEST4[I];

            QRTQESTR1[I] = TMQESTR[I, 11] + TMQESTR[I, 12] + TMQESTR[I, 1] +
TMQESTR[I, 2];
            QRTQESTR2[I] = TMQESTR[I, 3] + TMQESTR[I, 4] + TMQESTR[I, 5];
            QRTQESTR3[I] = TMQESTR[I, 6] + TMQESTR[I, 7] + TMQESTR[I, 8];
            QRTQESTR4[I] = TMQESTR[I, 9] + TMQESTR[I, 10];
            QRTQESTR5[I] = QRTQESTR1[I] + QRTQESTR2[I] + QRTQESTR3[I] +
QRTQESTR4[I];


            QRTQESTB1[I] = TMQESTB[I, 11] + TMQESTB[I, 12] + TMQESTB[I, 1] +
TMQESTB[I, 2];



            QRTQESTB2[I] = TMQESTB[I, 3] + TMQESTB[I, 4] + TMQESTB[I, 5];
            QRTQESTB3[I] = TMQESTB[I, 6] + TMQESTB[I, 7] + TMQESTB[I, 8];
```

```
            QRTQESTB4[I] = TMQESTB[I, 9] + TMQESTB[I, 10];
            QRTQESTB5[I] = QRTQESTB1[I] + QRTQESTB2[I] + QRTQESTB3[I] +
QRTQESTB4[I];

            QRTEVP1[I] = TMEVP[I, 11] + TMEVP[I, 12] + TMEVP[I, 1] + TMEVP[I, 2];
            QRTEVP2[I] = TMEVP[I, 3] + TMEVP[I, 4] + TMEVP[I, 5];
            QRTEVP3[I] = TMEVP[I, 6] + TMEVP[I, 7] + TMEVP[I, 8];
            QRTEVP4[I] = TMEVP[I, 9] + TMEVP[I, 10];
            QRTEVP5[I] = QRTEVP1[I] + QRTEVP2[I] + QRTEVP3[I] + QRTEVP4[I];

        }

        //WRITE(18,299)

        //WRITE(18,350)

        int JS1=1;
        int JS2 = 2;
        int JS3=3;
        int JS4 = 4;
        int JS5 = 0;

        for(int I=1; I<=NHYEAR; I++)
        {
            FileSystem.Print(18, "  ", IY[I].ToString("F3"), "  ", IYR[I].ToString("F3"), "  ",
JS1.ToString("F3"), "  ", QRTQO1[I].ToString("F3"), "  ",
QRTQEST1[I].ToString("F3"),QRTQESTR1[I].ToString("F3"), "  ",
QRTQESTS1[I].ToString("F3"), "  ", QRTQESTB1[I].ToString("F3"), "  ",
QRTEVP1[I].ToString("F3"));

            FileSystem.Print(18, "  ", IY[I].ToString("F3"), "  ", IYR[I].ToString("F3"), "  ",
JS2.ToString("F3"), "  ", QRTQO2[I].ToString("F3"), "  ", QRTQEST2[I].ToString("F3"),
QRTQESTR2[I].ToString("F3"), "  ", QRTQESTS2[I].ToString("F3"), "  ",
QRTQESTB2[I].ToString("F3"), "  ", QRTEVP2[I].ToString("F3"));

            FileSystem.Print(18, "  ", IY[I].ToString("F3"), "  ", IYR[I].ToString("F3"), "  ",
JS3.ToString("F3"), "  ", QRTQO3[I].ToString("F3"), "  ", QRTQEST3[I].ToString("F3"),
QRTQESTR3[I].ToString("F3"), "  ", QRTQESTS3[I].ToString("F3"), "  ",
QRTQESTB3[I].ToString("F3"), "  ", QRTEVP3[I].ToString("F3"));

            FileSystem.Print(18, "  ", IY[I].ToString("F3"), "  ", IYR[I].ToString("F3"), "  ",
JS4.ToString("F3"), "  ", QRTQO4[I].ToString("F3"), "  ", QRTQEST4[I].ToString("F3"),
QRTQESTR4[I].ToString("F3"), "  ", QRTQESTS4[I].ToString("F3"), "  ",
QRTQESTB4[I].ToString("F3"), "  ", QRTEVP4[I].ToString("F3"));

            FileSystem.Print(18, "  ", IY[I].ToString("F3"), "  ", IYR[I].ToString("F3"), "  ",
JS5.ToString("F3"), "  ", QRTQO5[I].ToString("F3"), "  ", QRTQEST5[I].ToString("F3"),
QRTQESTR5[I].ToString("F3"), "  ", QRTQESTS5[I].ToString("F3"), "  ",
QRTQESTB5[I].ToString("F3"), "  ", QRTEVP5[I].ToString("F3"));
```

```
    //WRITE(18,199)
  }

  for(int J=11; J<=12; J++)
  {
    double SQO1 = 0.0;
    double SQEST1 = 0.0;
    double SQESTR1=0.0;
    double SQESTS1 = 0.0;
    double SQESTB1 = 0.0;
    double SEVP1 = 0.0;

    for(int I=1; I<=NHYEAR; I++)
    {
      SQO1 += TMQO[I, J];
      SQEST1 += TMQEST[I, J];
      SQESTR1 += TMQESTR[I, J];
      SQESTS1 += TMQESTS[I, J];
      SQESTB1 += TMQESTB[I, J];
      SEVP1 += TMEVP[I, J];
    //L102: continue;
    }

    SQOMEAN[J] = SQO1 / NHYEAR;
    SQESTMEAN[J] = SQEST1 / NHYEAR;
    SQESTRMEAN[J] = SQESTR1 / NHYEAR;
    SQESTSMEAN[J] = SQESTS1 / NHYEAR;
    SQESTBMEAN[J] = SQESTB1 / NHYEAR;
    SEVPMEAN[J] = SEVP1 / NHYEAR;
  //L101:   continue;

  }

  for(int J=1; J<=10; J++)
  {
    double SQO2 = 0.0;
    double SQEST2 = 0.0;
    double SQESTR2 = 0.0;
    double SQESTS2 = 0.0;
    double SQESTB2 = 0.0;
    double SEVP2 = 0.0;

    for(int I=1;I<=NHYEAR; I++)
    {

      SQO2 += TMQO[I, J];
      SQEST2 += TMQEST[I, J];
      SQESTR2 += TMQESTR[I, J];
      SQESTS2 += TMQESTS[I, J];
      SQESTB2 += TMQESTB[I, J];
```

```
            SEVP2 += TMEVP[I, J];
        //L104: continue;

        }

        SQOMEAN[J] = SQO2 / NHYEAR;
        SQESTMEAN[J] = SQEST2 / NHYEAR;
        SQESTRMEAN[J] = SQESTR2 / NHYEAR;


        SQESTSMEAN[J] = SQESTS2 / NHYEAR;
        SQESTBMEAN[J] = SQESTB2 / NHYEAR;
        SEVPMEAN[J] = SEVP2 / NHYEAR;
    //L103:   continue;

    }


    double YQO   = 0.0;
    double YQEST   = 0.0;
    double YQESTR   = 0.0;
    double YQESTS   = 0.0;
    double YQESTB   = 0.0;
    double YEVP   = 0.0;

    for(int J=1; J<=12; J++)
    {
        YQO += SQOMEAN[J];
        YQEST += SQESTMEAN[J];
        YQESTR += SQESTRMEAN[J];
        YQESTS += SQESTSMEAN[J];
        YQESTB += SQESTBMEAN[J];
        YEVP += SEVPMEAN[J];
    }

    //WRITE(18,299)
    //WRITE(18,550)

    for(int J=11; J<=12; J++)
    {

        FileSystem.Print(18, "  ", J.ToString("F3"), "  ", SQOMEAN[J].ToString("F3"), "  ",
SQESTMEAN[J].ToString("F3"), "  ", SQESTRMEAN[J].ToString("F3"), "  ",
SQESTSMEAN[J].ToString("F3"), "  ", SQESTBMEAN[J].ToString("F3"), "  ",
SEVPMEAN[J].ToString("F3"));

    }

    for(int J=1; J<=10; J++)
    {
```

```
            FileSystem.Print(18, "  ", J.ToString("F3"), "  ", SQOMEAN[J].ToString("F3"), "  ",
SQESTMEAN[J].ToString("F3"), "  ", SQESTRMEAN[J].ToString("F3"), "  ",
SQESTSMEAN[J].ToString("F3"), "  ", SQESTBMEAN[J].ToString("F3"), "  ",
SEVPMEAN[J].ToString("F3"));

        }



        FileSystem.Print(18,"  ",YQO.ToString("F3"), "  ", YQEST.ToString("F3"), "  ",
YQESTR.ToString("F3"), "  ", YQESTS.ToString("F3"), "  ", YQESTB.ToString("F3"), "  ",
YEVP.ToString("F3"));


        double FQO1 = 0.0;
        double FQO2 = 0.0;
        double FQO3 = 0.0;
        double FQO4 = 0.0;
        double FQO5 = 0.0;


        FQO1 = SQOMEAN[11] + SQOMEAN[12] + SQOMEAN[1] + SQOMEAN[2];
        FQO2 = SQOMEAN[3] + SQOMEAN[4] + SQOMEAN[5];
        FQO3 = SQOMEAN[6] + SQOMEAN[7] + SQOMEAN[8];
        FQO4 = SQOMEAN[9] + SQOMEAN[10];
        FQO5 = FQO1 + FQO2 + FQO3 + FQO4;

        double FQEST1 = 0.0;
        double FQEST2 = 0.0;
        double FQEST3 = 0.0;
        double FQEST4 = 0.0;
        double FQEST5 = 0.0;

        FQEST1 = SQESTMEAN[11] + SQESTMEAN[12] + SQESTMEAN[1] +
SQESTMEAN[2];
        FQEST2 = SQESTMEAN[3] + SQESTMEAN[4] + SQESTMEAN[5];
        FQEST3 = SQESTMEAN[6] + SQESTMEAN[7] + SQESTMEAN[8];
        FQEST4 = SQESTMEAN[9] + SQESTMEAN[10];
        FQEST5 = FQEST1 + FQEST2 + FQEST3 + FQEST4;


        double FQESTR1 = 0.0;
        double FQESTR2 = 0.0;
        double FQESTR3 = 0.0;
        double FQESTR4 = 0.0;
        double FQESTR5 = 0.0;

        FQESTR1 = SQESTRMEAN[11] + SQESTRMEAN[12] + SQESTRMEAN[1] +
SQESTRMEAN[2];
        FQESTR2 = SQESTRMEAN[3] + SQESTRMEAN[4] + SQESTRMEAN[5];
```

```
        FQESTR3 = SQESTRMEAN[6] + SQESTRMEAN[7] + SQESTRMEAN[8];
        FQESTR4 = SQESTRMEAN[9] + SQESTRMEAN[10];
        FQESTR5 = FQESTR1 + FQESTR2 + FQESTR3 + FQESTR4;


        double FQESTS1 = 0.0;
        double FQESTS2 = 0.0;
        double FQESTS3 = 0.0;
        double FQESTS4 = 0.0;
        double FQESTS5 = 0.0;

        FQESTS1 = SQESTSMEAN[11] + SQESTSMEAN[12] + SQESTSMEAN[1] +
SQESTSMEAN[2];
        FQESTS2 = SQESTSMEAN[3] + SQESTSMEAN[4] + SQESTSMEAN[5];
        FQESTS3 = SQESTSMEAN[6] + SQESTSMEAN[7] + SQESTSMEAN[8];
        FQESTS4 = SQESTSMEAN[9] + SQESTSMEAN[10];
        FQESTS5 = FQESTS1 + FQESTS2 + FQESTS3 + FQESTS4;

        double FQESTB1 = 0.0;
        double FQESTB2 = 0.0;
        double FQESTB3 = 0.0;
        double FQESTB4 = 0.0;
        double FQESTB5 = 0.0;

        FQESTB1 = SQESTBMEAN[11] + SQESTBMEAN[12] + SQESTBMEAN[1] +
SQESTBMEAN[2];
        FQESTB2 = SQESTBMEAN[3] + SQESTBMEAN[4] + SQESTBMEAN[5];
        FQESTB3 = SQESTBMEAN[6] + SQESTBMEAN[7] + SQESTBMEAN[8];
        FQESTB4 = SQESTBMEAN[9] + SQESTBMEAN[10];
        FQESTB5 = FQESTB1 + FQESTB2 + FQESTB3 + FQESTB4;

        double FEVP1 = 0.0;
        double FEVP2 = 0.0;
        double FEVP3 = 0.0;
        double FEVP4 = 0.0;
        double FEVP5 = 0.0;

        FEVP1 = SEVPMEAN[11] + SEVPMEAN[12] + SEVPMEAN[1] + SEVPMEAN[2];
        FEVP2 = SEVPMEAN[3] + SEVPMEAN[4] + SEVPMEAN[5];
        FEVP3 = SEVPMEAN[6] + SEVPMEAN[7] + SEVPMEAN[8];
        FEVP4 = SEVPMEAN[9] + SEVPMEAN[10];
        FEVP5 = FEVP1 + FEVP2 + FEVP3 + FEVP4;

        //WRITE(18,199)
        //WRITE(18,650)

        FileSystem.Print(18," ",JS1.ToString("F3")," ", FQO1.ToString("F3")," ",
FQEST1.ToString("F3")," ", FQESTR1.ToString("F3")," ", FQESTS1.ToString("F3")," ",
FQESTB1.ToString("F3")," ", FEVP1.ToString("F3"));
```

```
        FileSystem.Print(18, "  ", JS2.ToString("F2"), "  ", FQO2.ToString("F2"), "  ",
FQEST2.ToString("F2"), "  ", FQESTR2.ToString("F2"), "  ", FQESTS2.ToString("F2"), "  ",
FQESTB2.ToString("F2"), "  ", FEVP2.ToString("F2"));
        FileSystem.Print(18, "  ", JS3.ToString("F3"), "  ", FQO3.ToString("F3"), "  ",
FQEST3.ToString("F3"), "  ", FQESTR3.ToString("F3"), "  ", FQESTS3.ToString("F3"), "  ",
FQESTB3.ToString("F3"), "  ", FEVP3.ToString("F3"));
        FileSystem.Print(18, "  ", JS4.ToString("F3"), "  ", FQO4.ToString("F3"), "  ",
FQEST4.ToString("F3"), "  ", FQESTR4.ToString("F3"), "  ", FQESTS4.ToString("F3"), "  ",
FQESTB4.ToString("F3"), "  ", FEVP4.ToString("F3"));
        FileSystem.Print(18, "  ", JS5.ToString("F3"), "  ", FQO5.ToString("F3"), "  ",
FQEST5.ToString("F3"), "  ", FQESTR5.ToString("F3"), "  ", FQESTS5.ToString("F3"), "  ",
FQESTB5.ToString("F3"), "  ", FEVP5.ToString("F3"));

        //WRITE(18,1600)

    }

    public void LEAPYR2(int iyear, int nd)
    {
        double Ipyr = 0.0;
        Ipyr = iyear % 4;
        if (Ipyr == 0)
            nd = 29;
    }

    private void button1_Click(object sender, EventArgs e)
    {
        double[] ZELEV = new double[10];
        double[,] AN = new double[1500, 10];
        double[,] CR = new double[1500, 10];
        double[,] CS = new double[1500, 10];
        double[,] TC = new double[1500, 10];
        double[] TLR = new double[1500];
        double[] PDIST = new double[1500];
        double[] ZAREA = new double[10];
        double[,] PPP = new double[1500, 10];
        double[] TM = new double[1500];
        double[] TX = new double[1500];
        double[,] TXZ = new double[1500, 10];
        double[,] TZ = new double[1500, 10];
        double[,] AET = new double[1500, 10];
        double[,] PET = new double[1500, 10];
        double[,] TMZ = new double[1500, 10];
        double[,] SCAP = new double[1500, 10];
        double[,] RMELT = new double[1500, 1500];


        double[] P = new double[1500];
```

```
double[] QC = new double[1500];
double[,] RBASEF = new double[1500, 10];
double[,] RBASEF1 = new double[1500, 10];
int[] IYR = new int[10];
double[,] SMELT = new double[1500, 10];
double[,] RUNOFF = new double[1500, 10];
double[,] IPR = new double[1500, 10];
double[,] PP = new double[1500, 10];
double[,] RBASEF2 = new double[1500, 10];
double[,] SRUNOFF = new double[1500, 10];
double[,] RRUNOFF = new double[1500, 10];
double[,] RMRUNOFF = new double[1500, 10];
double[,] QRSCA = new double[1500, 10];
double[,] SBASEF = new double[1500, 10];
double[,] SBRUNOFF = new double[1500, 10];
double[,] RBRUNOFF = new double[1500, 10];
double[,] BRUNOFF = new double[1500, 10];
double[,] QRSFA = new double[1500, 10];
double[,] RSRUNOFF = new double[1500, 10];
double[,] RRRUNOFF = new double[1500, 10];
double[,] SCAPC = new double[1500, 10];
double[] QTOT = new double[1500];
double[,] DEPSMELT = new double[1500, 10];
double[,] DEPRMELT = new double[1500, 10];
double[,] DEPRAIN = new double[1500, 10];
double[,] SUMMOIST = new double[1500, 10];
double[,] SMOIST = new double[1500, 10];
double[] AREAST = new double[1500];
double[] EVPVOL = new double[1500];
double[] QOD = new double[1500];
double[] QESTD = new double[1500];
double[] QESTRD = new double[1500];
double[] QESTSD = new double[1500];
double[] QESTBD = new double[1500];
double[,] AETVOL = new double[1500, 10];
double[] EVP = new double[1500];
double[,] PETVOL = new double[1500, 10];
double[] PEVPVOL = new double[1500];
double[] PEVP = new double[1500];
double[,] TMELT = new double[1500, 10];
double[] RAINSCA = new double[1500];
double[] BASESCA = new double[1500];
double[] BASESFA = new double[1500];
double[,] RDEPSCA = new double[1500, 10];
```



```
double[,] RDEPSFA = new double[1500, 10];
double[,] SDEPSCA = new double[1500, 10];
```

```
double[,] RSDEPSCA = new double[1500, 10];
double[] WBTRAIN = new double[1500];
double[] WBTRAINS = new double[1500];
double[] WBTMELT = new double[1500];
double[,] DEPRAINS = new double[1500, 10];
double[,] SCAMOIST = new double[1500, 10];
double[,] SFAMOIST = new double[1500, 10];
double[,] RMOIST = new double[1500, 10];
double[,] REPS = new double[1500, 10];
double[,] REPR = new double[1500, 10];



string TMSTN=null;
string TXSTN=null;
// string PSTNAME=null;
string ZONENUM=null;
string DISSTN=null;


double[]  UH=new double[1500];
double[]  QEST=new double[1500];
double[]  BFLOW=new double[1500];
double  NRUN=0.0;
//double  DUH=0.0;
//double  DLT=0.0;
double[]  QESTR=new double[1500];
double[]  QESTS=new double[1500];
//double  NS=0.0;
//double  NR=0.0;
//double  NB=0.0;
double[]  BASEIN=new double[1500];
double[]  QESTB=new double[1500];
double[]  QO=new double[1500];
double[]  QSFA=new double[1500];
double[]  QSCA=new double[1500];
double[]  AREAR=new double[1500];
double[]  AREAS=new double[1500];
//double  AR=0.0;
//double  BR=0.0;
//double  AS=0.0;
//double  BS=0.0;
//double  AKB=0.0;



double PSELEV=0.0;
   int NZPUSE=0;
int NZTUSE = 0;
double TSELEV=0.0;
```

```
int NN1=0;
double RAINREPF=0.0;

   double SMISCA = 0.0;
   double SMISFA = 0.0;
   double AETLIM=0.0;
   double GEOFAC=0.0;

   double SUMS=0.0;
   double SUMR=0.0;
      double SUMB=0.0;
      double SUMQOB=0.0;
      double SUMEV=0.0;
      double SUMPEV=0.0;
      double SUMWBR=0.0;
      double SUMWBRS=0.0;
      double SUMWBM=0.0;
```

```
FileSystem.FileOpen(1, "TM8891.DAT", OpenMode.Input, OpenAccess.Default,
OpenShare.Default, 500);

FileSystem.FileOpen(10, "TX8891.DAT", OpenMode.Input, OpenAccess.Default,
OpenShare.Default, 500);

FileSystem.FileOpen(2, "PPT8891.DAT", OpenMode.Input, OpenAccess.Default,
OpenShare.Default, 500);

FileSystem.FileOpen(3, "SCA8891a.DAT", OpenMode.Input, OpenAccess.Default,
OpenShare.Default, 500);

FileSystem.FileOpen(4, "DIS8891.DAT", OpenMode.Input, OpenAccess.Default,
OpenShare.Default, 500);

FileSystem.FileOpen(5, "para11.DAT", OpenMode.Input, OpenAccess.Default,
OpenShare.Default, 500);

FileSystem.FileOpen(9, "MODE&OPAR8891.DAT", OpenMode.Input,
OpenAccess.Default, OpenShare.Default, 500);
```

```
        FileSystem.FileOpen(15, "COMPQ8891.DAT", OpenMode.Output,
OpenAccess.Write, OpenShare.Shared, 1000);


        //FileSystem.FileOpen(17, "ZMACCU8891.DAT", OpenMode.Output,
OpenAccess.Write, OpenShare.Shared, 1000);


        //FileSystem.FileOpen(18, "QRTVOL8891.DAT", OpenMode.Output,
OpenAccess.Write, OpenShare.Shared, 1000);


        FileSystem.FileOpen(18, "MONQRTANL8891.DAT", OpenMode.Output,
OpenAccess.Write, OpenShare.Shared, 1000);


        FileSystem.FileOpen(16, "CHECK8891.DAT", OpenMode.Output, OpenAccess.Write,
OpenShare.Shared, 1000);



        int NHYEAR = 0;
        int NDAYS = 0;
        FileSystem.Input(1, ref NHYEAR);
        for (int i = 0; i <= NHYEAR-1; i++)
        {
        FileSystem.Input(1, ref IYR[i]);

        }

        FileSystem.Input(1, ref NDAYS);
        NRUN = NDAYS;


        int NZ = 0;
        double BAREA = 0.0;


        FileSystem.Input(1, ref NZ);


        for (int j = 0; j <= NZ-1; j++)
        {
        FileSystem.Input(1, ref ZAREA[j]);
        FileSystem.Input(1, ref ZELEV[j]);
        BAREA = BAREA + ZAREA[j];

        //FileSystem.Print(16,"  ",ZAREA[j].ToString("F3"), "  ", ZELEV[j].ToString("F3"),"
", BAREA.ToString("F3"));
        }
```

```
int NSTATTM = 0;

FileSystem.Input(1, ref NSTATTM);

int NN = 0;


for(int k=1; k<=NSTATTM;k++)
{
 FileSystem.Input(1, ref  TMSTN);


// FileSystem.PrintLine(16,"  ",TMSTN.ToString("F3"));


   FileSystem.Input(1, ref TSELEV);
   FileSystem.Input(1, ref NZTUSE);

   for(int j=0; j<=NZTUSE-1;j++)
   {
      FileSystem.Input(1, ref ZELEV[j]);
   }

   //FileSystem.Print(16,"  ",TSELEV.ToString("F3"));

   //FileSystem.Print(16,"  ",NZTUSE.ToString("F3"));

   //for(int J=1; J<=NZTUSE; J++)
   //{
    //  FileSystem.Print(16,"  ",ZELEV[J].ToString("F3"));
   //}



   NN = NN + NZTUSE;
   for (int i = 0; i <= NDAYS-1; i++)
   {
      FileSystem.Input(1, ref TM[i]);
   }

   for (int i = 0; i <= NDAYS-1; i++)
   {
      FileSystem.Input(1, ref TLR[i]);
   }
```

```
            for(int i=1; i<=NDAYS; i++)
            {

               //TM[i]=TM[i]+1;
               //TM[i]=TM[i]+2;
               //TM[i]=TM[i]+3;
               //TM[i]=TM[i]+4;

            }

             TLAPSE( NDAYS, NZTUSE, NN, TM, TLR, TSELEV, ZELEV, TZ,
NZ);


            }
            for(int j=0; j<=NZ-1; j++)
            {
            for (int i=0; i<=NDAYS-1;i++)
            {
            TMZ[i,j]=TZ[i,j];
            }
            }

        //CHKDATA( TMZ, NZ, NHYEAR, IYR);


            double NSTATTX = 0.0;
           FileSystem.Input(10, ref NSTATTX);



           NN = 0;

           for (int k = 1; k <= NSTATTX; k++ )
           {
              FileSystem.Input(10, ref TXSTN);

              //FileSystem.PrintLine(16, "   ",TXSTN.ToString("F3"));

              FileSystem.Input(10, ref PSELEV);
              FileSystem.Input(10, ref NZTUSE);
           for(int j=0; j<=NZTUSE-1; j++)
           {
              FileSystem.Input(10, ref ZELEV[j]);

           }
```

```
        for(int j=0; j<=NZTUSE-1; j++)
        {
          FileSystem.Print(16,"  ", PSELEV.ToString("F3"),"  ", NZTUSE.ToString("F3"), "
",ZELEV[j].ToString("F3"));
        }

        /*
        FileSystem.PrintLine(16, "   ",TSELEV.ToString("F3"));
        FileSystem.PrintLine(16, "   ",NZTUSE.ToString("F3"));
        for(int j=1; j<=NZTUSE; j++)
        {
          FileSystem.PrintLine(16, "   ",ZELEV[j].ToString("F3"));
        }

        */

        NN=NN+NZTUSE;

        for(int i=0; i<=NDAYS-1; i++)
        {
          FileSystem.Input(10, ref TX[i]);

        }

        for(int i=0; i<=NDAYS-1; i++)
          FileSystem.Input(10, ref TLR[i]);


        for(int i=1; i<=NDAYS;i++)
        {
          /*
          TX[i]=TX[i]+1;
          TX[i]=TX[i]+2;
          TX[i]=TX[i]+3;
          TX[i]=TX[i]+4;
          */
        }
        TLAPSE(NDAYS, NZTUSE, NN, TX, TLR, TSELEV, ZELEV,TZ, NZ);

        }

        for(int j=0; j<=NZ-1; j++)
          for(int i=0; i<=NDAYS-1;i++)
            TXZ[i,j]=TZ[i,j];

        //CHKDATA(TXZ,NZ,NHYEAR,IYR);

        double NSTATP=0.0;
        FileSystem.Input(2, ref NSTATP);
```

```
//FileSystem.PrintLine(16, NSTATP.ToString("F3"));

//double N1=0.0;
//double N2=0.0;

string PSTNNAME=null;


for(int k=0; k<=NSTATP-1; k++)
{
  FileSystem.Input(2, ref PSTNNAME);


  //FileSystem.PrintLine(16, PSTNNAME.ToString("F3"));
  FileSystem.Input(2, ref PSELEV);
  FileSystem.Input(2, ref NZPUSE);
  for(int j=0; j<=NZPUSE-1; j++)
  {
  FileSystem.Input(2, ref ZELEV[j]);

  }
  /*
  FileSystem.Print(16, " ", PSELEV.ToString("F3"));
  FileSystem.Print(16, " ", NZPUSE.ToString("F3"));

  for(int j=1; j<=NZPUSE; j++)
  {
    FileSystem.Print(16, " ", ZELEV[j].ToString("F3"));
  }

 */
```



```
  NN1=NN1+NZPUSE;

  for(int i=0; i<=NDAYS-1; i++)
  {
    FileSystem.Input(2, ref P[i]);

  }

  for(int i=0; i<=NDAYS-1; i++)
  {
    RAINREPF=0.50f;

    if(i>1&& i<181)
    {
      P[i]=P[i]*RAINREPF;
    }
```

```
            if (i > 395 && i < 546)
            {
               P[i] = P[i] * RAINREPF;
            }
            if (i > 760 && i < 911)
            {
               P[i] = P[i] * RAINREPF;
            }
            }


           for(int i=1; i<NDAYS;i++)
              {
              // P[i] = P[i] +0.05*P[i];
              // P[i] = P[i] -0.05*P[i];
              //P[i] = P[i] +0.10*P[i];
              //P[i] = P[i] -0.10*P[i];
              }



           for(int i=0; i<=NDAYS-1; i++)
           {
              FileSystem.Input(2, ref PDIST[i]);

           }

           PPTDIST(NDAYS, NZPUSE, NN1, P,PDIST,PSELEV,ZELEV, PP,NZ);
           }


          //CHKDATA(PP,NZ,NHYEAR,IYR);


          for(int j=0; j<=NZ-1; j++)
          {
          FileSystem.Input(3, ref ZONENUM);
//          FileSystem.PrintLine(16,"  ",ZONENUM.ToString("F3"));

           for (int i = 0; i <= NDAYS-1; i++)
           {
              FileSystem.Input(3, ref SCAP[i,j]);
           }

           for(int i=0; i<=NDAYS-1; i++)
           {
              //SCAP[i,j]=0.95*SCAP[i,j];
              //SCAP[i,j]=0.90*SCAP[i,j];
              //SCAP[i,j]=0.85*SCAP[i,j];
```

```
        }

        for(int i=0; i<=NDAYS-1; i++)
        {
          //FileSystem.PrintLine(16,"  ",SCAP[i,j].ToString("F3"));
        }

      }



//        CHKDATA(SCAP,NZ, NHYEAR,IYR);



      FileSystem.Input(4, ref DISSTN);
      //FileSystem.PrintLine(16,"  ",DISSTN.ToString("F3"));


      for (int i = 0; i <= NDAYS-1; i++)
        FileSystem.Input(4, ref QO[i]);
      /*
      var a="";
      FileSystem.Input(5, ref a);

      int allo = a.IndexOf('*');

      string d = a.Substring(0,2);
      string l = a.Substring(3, 4);
      int dk = Convert.ToInt32(d);
      double lk = Convert.ToDouble(l);

      int p = 0;
      for (int i = 0; i <= NZ-1; i++)
      {
      for (int j = 0; j <= NDAYS-1 ; j++)
      {

        AN[j, i] = lk;
        p++;


        if(p==dk)
        {
          FileSystem.Input(5, ref a);
          //allo = a.IndexOf('*');
```

```
            //d = a.Substring(0, 2);
            //l = a.Substring(3, 4);
            dk = Convert.ToInt32(d);
            lk = Convert.ToDouble(l);
            p = 0;
        }

        if (j > 30)
        {
            MessageBox.Show(Convert.ToString(AN[31, 0]));

            MessageBox.Show(Convert.ToString(AN[32, 0]));

            MessageBox.Show(Convert.ToString(AN[33, 0]));




        }
    }

    }
    */
    for(int j=0; j<=NZ-1; j++)
    {
        for (int i = 0; i <= NDAYS - 1; i++)
            FileSystem.Input(5, ref AN[i, j]);
        for (int i = 0; i <= NDAYS-1; i++)
            FileSystem.Input(5, ref CR[i,j]);
        for (int i = 0; i <= NDAYS-1; i++)
            FileSystem.Input(5, ref CS[i,j]);
        for (int i = 0; i <= NDAYS-1; i++)
            FileSystem.Input(5, ref TC[i,j]);
        for (int i = 0; i <= NDAYS-1; i++)
            FileSystem.Input(5, ref IPR[i,j]);

    }

    //PCONT(NDAYS, NZ,IPR,AN,PP,TMZ,TC,SCAP,PPP);


    if(QC[1] == 187.8)
    {
        SUMS = 0.0;


/////PAGE 4/////

            SUMR = 0.0;
```

```
            SUMB = 0.0;
                SUMQOB = 0.0;
                SUMEV = 0.0;
                SUMPEV = 0.0;
                SUMWBR = 0.0;
                SUMWBRS = 0.0;
                SUMWBM = 0.0;

        }

        for(int i=1; i<=NDAYS;i++)
        {
            //double QZ = 0.0;
            double QZS = 0.0;




            double QZR = 0.0;
            double QZB = 0.0;
            double QZRM = 0.0;
            double QZRSCA = 0.0;
            double QZRSFA = 0.0;
               double QZBSCA = 0.0;
               double QZBSFA = 0.0;
            double ZSCA = 0.0;
            double ZSFA = 0.0;
            double ZSCAE = 0.0;
            double TQSCA = 0.0;
            double TQSFA = 0.0;
            double TQ = 0.0;
            double AETZVOL = 0.0;
               double PETZVOL = 0.0;
               double WZSSCA = 0.0;
               double WZRSSCA = 0.0;
               double WZRSCA = 0.0;
               double WZRSFA = 0.0;
               double CONV=0.0;
               CONV= (1000/86400);


          for (int j = 1; j <= NZ; j++)
          {
             SMELT[i,j]=AN[i,j]*TMZ[i,j]*SCAP[i,j];

             if (IPR[i, j] == 1)
                RMELT[i, j] = (4.2 * TMZ[i, j] * PP[i, j] * SCAP[i, j]) / 333.5;

             if (IPR[i, j] == 0)
                QRSFA[i, j] = PPP[i, j];
```

```
if (IPR[i, j] == 1)
   QRSCA[i, j] = PP[i, j] * SCAP[i, j];
if (TMZ[i, j] < 0)
{
   QRSCA[i, j] = 0.0;
   QRSFA[i, j] = PPP[i, j] * (1 - SCAP[i, j]);

}



//SBASEF[i, j]=(1-CS[i, j]*AN[i, j]*TMZ[i, j]*SCAP[i, j]);




//RBASEF1[i, j]=(1-CR[i, j])*PP[i, j]*SCAP[i, j];
//RBASEF2[i, j]=(1-CR[i, j])*PPP[i, j]*(1-SCAP[i, j]);
//RBASEF[i, j]=RBASEF1[i, j]+RBASEF2[i, j];

SDEPSCA[i, j] = AN[i, j] * TMZ[i, j] * SCAP[i, j];
RSDEPSCA[i, j] = 4.2 * TMZ[i, j] * PP[i, j] * SCAP[i, j] / 333.5;
RDEPSCA[i, j] = PP[i, j] * SCAP[i, j];
RDEPSFA[i, j] = PP[i, j] * (1 - SCAP[i, j]);



SMISCA = 100.0;
SMISFA = 100.0;


DEPSMELT[i, j] = AN[i, j] * TMZ[i, j] * SCAP[i, j];
DEPRMELT[i, j] = 4.2 * TMZ[i, j] * PP[i, j] * SCAP[i, j] / 333.5;
DEPRAINS[i, j] = PP[i, j] * SCAP[i, j];
DEPRAIN[i, j] = PP[i, j] * (1 - SCAP[i, j]);

SCAMOIST[i, j] = DEPSMELT[i, j] + DEPRMELT[i, j] + DEPRAINS[i, j];
SFAMOIST[i, j] = DEPRAIN[i, j];

if(i==1)
   goto L199;


SMOIST[i,j]=SMOIST[i-1,j]+SCAMOIST[i,j];

if(SCAP[i,j]==0.0)
{
   SMOIST[i,j]=0.0;
```

```
        }


        ////////page5/////

        RMOIST[i,j]=RMOIST[i-1,j]+SFAMOIST[i,j];


          goto L299;

        if(SCAP[i,j]==0.0)
            SMOIST[i,j]=0.0;
          RMOIST[i,j]=SFAMOIST[i,j]+SMISFA/2;


          L299: continue;


        //double BETA=4.0f;
        REPS[i,j]=Math.Pow((SMOIST[i,j]/SMISCA),4);
        REPR[i, j] = Math.Pow((RMOIST[i, j] / SMISFA),4);

        double REPSMAX=0.6f;
        double REPRMAX=0.6f;


        if(REPS[i,j]>REPSMAX)
          REPS[i,j]=REPSMAX;
        if(REPR[i,j]>REPRMAX)
          REPR[i,j]=REPRMAX;

      // FileSystem.Print(16," ", j.ToString("F3")," ",i.ToString("F3")," ",SMOIST[i,j].ToString("F3")," ",SCAMOIST[i,j].ToString("F3")," ",1REPR[i,j].ToString("F3");
        //FileSystem.Print(16," ",RMOIST[i,j].ToString("F3")," ",SFAMOIST[i,j].ToString("F3"));

              SRUNOFF[i,j]=SMELT[i,j]*REPS[i,j]*ZAREA[j]*CONV;
          RSRUNOFF[i,j]=QRSCA[i,j]*REPS[i,j]*ZAREA[j]*CONV;
          RMRUNOFF[i,j]=RMELT[i,j]*REPS[i,j]*ZAREA[j]*CONV;
          RRRUNOFF[i,j]=QRSFA[i,j]*REPR[i,j]*ZAREA[j]*CONV;


RUNOFF[i,j]=SRUNOFF[i,j]+RSRUNOFF[i,j]+RRRUNOFF[i,j]+RMRUNOFF[i,j];


          TMELT[i,j]=SRUNOFF[i,j]+RMRUNOFF[i,j];
```

```
SBASEF[i,j]=0.5*(1-REPS[i,j]*(SMELT[i,j]+QRSCA[i,j]+RMELT[i,j]));



RBASEF[i,j]=0.5*(1-REPR[i,j])*QRSFA[i,j];

SBRUNOFF[i,j]=SBASEF[i,j]*ZAREA[j]*CONV;
RBRUNOFF[i,j]=RBASEF[i,j]*ZAREA[j]*CONV;

BRUNOFF[i,j]=SBRUNOFF[i,j]+RBRUNOFF[i,j];



  if(i==1)
   goto L399;

RMOIST[i,j]=RMOIST[i,j]+(1-REPR[i,j]*QRSFA[i,j]-RBASEF[i,j]);


   L399: PET[i,j]=0.70*(0.2252*Math.Exp(0.0938*TXZ[i,j]));

   PET[i,j]=PET[i,j]*(1-SCAP[i,j]);
if(TXZ[i,j]<1)
   PET[i,j]=0.0;

      AETLIM=100.0;
         if(RMOIST[i,j]<AETLIM)
   AET[i,j]=PET[i,j]*(RMOIST[i,j]/SMISFA);

if(RMOIST[i,j]>AETLIM)
   AET[i,j]=PET[i,j];

RMOIST[i,j]=RMOIST[i,j]-AET[i,j]-0.25;

if(RMOIST[i,j]>SMISFA)
   RMOIST[i,j]=SMISFA;

if(SMOIST[i,j]>SMISCA)
   SMOIST[i,j]=SMISCA;

FileSystem.Print(16," "+SCAMOIST[i,j].ToString("F3")," ",SMOIST[i,
j].ToString("F3")," ",SFAMOIST[i, j]," ",RMOIST[i, j].ToString("F3"));
   FileSystem.Print(16," ",AET[i, j].ToString("F3")," ",PET[i, j].ToString("F3"));



QZS=QZS+SRUNOFF[i,j];
QZRSCA=QZRSCA+RSRUNOFF[i,j];
```

```
QZRSFA=QZRSFA+RRRUNOFF[i,j];
QZRM=QZRM+RMRUNOFF[i,j];



QZR=QZR+RSRUNOFF[i,j]+RRRUNOFF[i,j];



TQSCA=TQSCA+SRUNOFF[i,j]+RSRUNOFF[i,j]+RMRUNOFF[i,j];
TQSFA=TQSFA+RRRUNOFF[i,j];
TQ=TQSCA+TQSFA;

QZB=QZB+BRUNOFF[i,j];

//////page6/////


//QZBSCA=QZBSCA=SBRUNOFF[i,j];
//QZBSFA=QZBSFA+RBRUNOFF[i,j];


PETVOL[i,j]=PET[i,j]*ZAREA[j]*1000;

GEOFAC=1.0;

AETVOL[i,j]=AET[i,j]*ZAREA[j]*1000*GEOFAC;


AETZVOL=AETZVOL+AETVOL[i,j];
PETZVOL=PETZVOL+PETVOL[i,j];



WZSSCA=WZSSCA+SDEPSCA[i,j]*ZAREA[j]*CONV;
WZRSSCA=WZRSSCA+RSDEPSCA[i,j]*ZAREA[j]*CONV;
WZRSCA=WZRSCA*RDEPSCA[i,j]*ZAREA[j]*CONV;
WZRSFA=WZRSFA=RDEPSFA[i,j]*ZAREA[j]*CONV;


SCAPC[i,j]=SCAP[i,j];
if(SMELT[i,j]<0)
   SCAPC[i,j]=0.0;
ZSCAE=ZSCAE+SCAPC[i,j]*ZAREA[j];
ZSCA=ZSCA+SCAP[i,j]*ZAREA[j];
ZSFA=ZSFA+(1-SCAP[i,j])*ZAREA[j];

   //L102: continue;
```

```
        }

        AREAS[i]=ZSCAE;
        AREAR[i]=ZSFA;
        AREAST[i]=ZSCA;
        QSCA[i]=TQSCA;
        QSFA[i]=TQSFA;
        QTOT[i]=TQ;
        BASEIN[i]=QZB;


        EVP[i]=AETZVOL/(BAREA*1000);
        PEVP[i]=PETZVOL/(BAREA*1000);


        WBTRAIN[i]=WZRSFA;
        WBTRAINS[i]=WZRSCA;
        WBTMELT[i]=WZSSCA+WZRSSCA;


        RAINSCA[i]=QZRSCA;
        BASESCA[i]=QZBSCA;
        BASESFA[i]=QZBSFA;

        SUMS=SUMS+QSCA[i]*86400;
        SUMR=SUMR+QSFA[i]*86400;
        SUMB=SUMB+BASEIN[i]*86400;
        SUMQOB=SUMQOB+QO[i]*86400;
        SUMEV=SUMEV+EVP[i]*BAREA*1000;
        SUMPEV=SUMPEV+PEVP[i]*BAREA*1000;


        SUMWBR=SUMWBR+WBTRAIN[i]*86400;
        SUMWBRS=SUMWBRS+WBTRAINS[i]*86400;
        SUMWBM=SUMWBM+WBTMELT[i]*86400;


       // System.Print(15,"  ",i.ToString("F3"),"  ",SUMQOB.ToString("F3"),"
",SUMR.ToString("F3"),
         //          "  ",SUMS.ToString("F3"),"  ",SUMB.ToString("F3"),"
",SUMEV.ToString("F3"),"  ",
          //          SUMPEV.ToString("F3"),"  ",SUMWBR.ToString("F3"),"
",SUNWBRS.ToString("F3"));


        //L101: continue;



      }
```

```
            for(int i=1; i<=NDAYS; i++)
            {
            //   FileSystem.Print(15," ",i.ToString("F3")," ",QSCA[i].ToString("F3")," 
",QSFA[i].ToString("F3")," ",AREAR[i].ToString("F3")," ",AREAS[i].ToString("F3")," 
",AREAST[i].ToString("F3")," ",QO[i].ToString("F3"));
            }



           ////////page7/////////////


        int MODE=0;
        FileSystem.Input(3, ref MODE);
        switch(MODE)
        {
           case 1: OPTIMIZE(); break;
           case 2: ROUTE(); break;

        }

        for(int I=1; I<=NDAYS; I++)
        {
//            FileSystem.Print(55," ",I.ToString("F3")," ",QO[I].ToString("F3")," 
",QTOT[I].ToString("F3")," ",QSCA[I].ToString("F3")," ",QSFA[I].ToString("F3")," 
",AREAS[I].ToString("F3")," ",AREAR[I].ToString("F3")," ",AREAST[I].ToString("F3")," 
",BASEIN[I].ToString("F3")," ");
        }


        for(int i=1; i<=NDAYS; i++)
        {
              QO[i]=QO[i]*86.4/BAREA;
            QEST[i]=QEST[i]*86.4/BAREA;
            QESTR[i]=QESTR[i]*86.4/BAREA;
            QESTS[i]=QESTS[i]*86.4/BAREA;
            QESTB[i]=QESTB[i]*86.4/BAREA;
            EVP[i]=EVP[i];
            PEVP[i]=PEVP[i];


            WBTRAIN[i]=WBTRAIN[i]*86.4/BAREA;
            WBTRAINS[i]=WBTRAINS[i]*86.4/BAREA;
            WBTMELT[i]=WBTMELT[i]*86.4/BAREA;


            //RAINSCA[i]=RAINSCA[i]*86.4/BAREA;
```

```
            //BASESCA[i]=BASESCA[i]*86.4/BAREA;
            //BASESFA[i]=BASESFA[i]*86.4/BAREA;

            //QESTS[i]=QESTS[i]-RAINSCA[i];
            //QESTR[i]=QESTR[i]+RAINSCA[i];


        }

          for(int i=1; i<=NDAYS;i++)
          {
            /*
            QO[i]=QO[i]*86400/1000000;
            QEST[i]=QEST[i]*86400/1000000;
            QESTR[i]=QESTR[i]*86400/1000000;
            QESTS[i]=QESTS[i]*86400/1000000;
            QESTB[i]=QESTB[i]*86400/1000000;
            EVP[i]=EVP[i]*BAREA*1000/1000000;
            PEVP[i]=PEVP[i]*BAREA*1000/1000000;



            WBTRAIN[i]=WBTRAIN[i]*86400/1000000;
            WBTRAINS[i]=WBTRAINS[i]*86400/1000000;
            WBTMELT[i]=WBTMELT[i]*86400/1000000;



            RAINSCA[i]=RAINSCA[i]*86400/1000000;
            BASESCA[i]=BASESCA[i]*86400/1000000;
            BASESFA[i]=BASESFA[i]*86400/1000000;



            QESTS[i]=QESTS[i]-RAINSCA[i];
            QESTR[i]=QESTR[i]+RAINSCA[i];

             */
          }

        //FileSystem.Print(15);

        /////PAGE 8/////

          for(int i=1; i<NDAYS;i++)
          {
            //FileSystem.Print(15,"  ",i.ToString("F3"), "    ",QO[i].ToString("F3"), "
",QEST[i].ToString("F3"));
```

```
//                FileSystem.Print(15,"  ",i.ToString("F3"), "    ",QO[i].ToString("F3"), "
",QEST[i].ToString("F3"),"  ",QESTR[i].ToString("F3"), "    ",QESTB[i].ToString("F3"), "
",EVP[i].ToString("F3"), "    ",1PEVP[i].ToString("F3"));
//                FileSystem.Print(*,"  ",i.ToString("F3"), "    ",QO[i].ToString("F3"), "
",QEST[i].ToString("F3"),"  ",QESTR[i].ToString("F3"), "    ",QESTS[i].ToString("F3"), "
",QESTB[i].ToString("F3"), "    ",EVP[i].ToString("F3"), "    ",1PEVP[i].ToString("F3"));
//                FileSystem.Print(15,"  ",i.ToString("F3"), "    ",QO[i].ToString("F3"), "
",QEST[i].ToString("F3"),"  ",WBTRAIN.ToString("F3"),"  ",WBTRAINS.ToString("F3"),"
",WBTMELT.ToString("F3"));
//                FileSystem.Print(15,"  ",i.ToString("F3"), "    ",QO[i].ToString("F3"), "
",QEST[i].ToString("F3"),"  ",WBTRAIN.ToString("F3"),"  ",WBTRAINS.ToString("F3"),"
",WBTMELT.ToString("F3"),"  ",1EVP.ToString("F3"),"  ",PEVP.ToString("F3"));

            }

            for(int I=1; I<=NDAYS; I++)
            {
//                FileSystem.Print(15,"  ",I.ToString("F3"), "    ",QESTB.ToString("F3"), "
",BASESCA.ToString("F3"), "    ",BASESFA.ToString("F3"), "    ",FAKE1.ToString("F3"), "
",FAKE2.ToString("F3"), "    ",FAKE3.ToString("F3"));

            }

            if(MODE==1)
               goto L777;

          //WRITE(*,666) AR,BR,AS,BS,AKB

            //L777: continue;
          L777:
            EFF(QEST,QO,NDAYS);
          //   MONQRTANL(NHYEAR,IYR,QO,QEST,QESTR,QESTS,QESTB,EVP);
            MONQRTANL(NHYEAR, IYR,QO,QEST,WBTRAIN,WBTRAINS,PEVP,EVP);

//MONQRTANL(NDAYS,IYR,QO,QEST,WBTRAIN,WBTRAINS,WBTMELT,EVP);

//MONQRTANL(NHYEAR,IYR,QO,QEST,WBTRAIN,WBTRAINS,BASESCA,BASESFA)
;
            //QRTANL(QO,QEST,QESTR,QESTS,QESTB,EVP);
            ZONEMELT(TMELT,SCAP,NZ);


    }

    }
    }
```

CONCLUSIONS

1. Many modules have been developed in Fortran for water management which are not so user friendly.

2. In the present study, part of the Fortran code are modified and developed successfully using IDE Visual Studio technology and C# to characterize the river flow.

3. Due to paucity of time, the programme code has been developed, thoroughly checked, and run using test data.

4. A full-fledged GUI can be developed at later stage.

REFERENCES

- www.tutorialpoint.com
- www.codeproject.com
- http://msdn.microsoft.com
- **http://www.tandfonline.com/doi/pdf/10.1080/02626660209492910**