DATA PROCESS IN LacZ ASSAY

AN COMPUTER PROGRAM

WRITTEN IN TURBO PASCAL

WUQIAO YUAN

Ohio University

April, 1991

```
{$R+}
PROGRAM LacZassay (INPUT, OUTPUT);

 USES PRINTER, CRT;

 CONST
  Title = 'ASSAY  FOR  LacZ  EXPRESSION';
  LineLength = 75;

 TYPE
  XYARRAY = ARRAY [1..10] of REAL;
  LINEARRAY = ARRAY [1..6] of INTEGER;
  DATAARRAY = ARRAY [1..6, 1..20] of REAL;
  MEDIUMARRAY = ARRAY [1..6] of STRING[14];
  SAMPLEARRAY = ARRAY [1..6, 1..20] OF STRING[25];
   STRING13 = STRING[13];

 VAR
  A, B : REAL;
  Group : INTEGER;
  Num : LINEARRAY;
  Medium : MEDIUMARRAY;
  SamppleName : SAMPLEARRAY;
  SpAct : DataArray;
  InData, OutData : TEXT;

 PROCEDURE PrintCenteredTitle;
  VAR
   CenteredTitleEnd : INTEGER;
  BEGIN
   CenteredTitleEnd := ROUND((LENGTH(Title) + LineLength) / 2);
   WRITELN(Lst, Title : CenteredTitleEnd);
   WRITELN(Lst);  WRITELN(Lst);
  END;

 PROCEDURE GetInformation;
  VAR
   Date, Host, Vector, DNA : STRING;
  BEGIN
   CLRSCR;
   WRITE('Enter the date of today :> ');
   READLN(Date);
   WRITE('Enter the name of the host strain used for the assay :> ');
   READLN(Host);
   WRITE('Enter the name of the vector used in the assay :> ');
   READLN(Vector);
   WRITE('Enter the name of the test DNA :> ');
   READLN(DNA);
   WRITELN(Lst, ' ':5, 'No.: _______', 'Date:':48, Date); writeln(Lst);
   WRITELN(Lst, ' ':5, 'Host: ', Host, ';'); writeln(Lst);
   WRITELN(Lst, ' ':5, 'Vector: ', Vector, ';', '   Test DNA: ', DNA, ';');
   WRITELN(Lst);
  END;
```

```
PROCEDURE DoRegression (VAR N : INTEGER; VAR XName, YName : STRING13;
                                VAR X, Y : XYARRAY; VAR A, B, R : REAL);
 VAR
  Signal : CHAR;
  Num1, Num2, SumX, SumX2, SumY, SumY2, SumXY : REAL;
 BEGIN
  REPEAT
   READLN(InData, Signal);
  UNTIL Signal = '@';
  READLN(InData, YName, XName);
  SumX := 0; SumX2 := 0; SumY := 0; SumY2 := 0; SumXY := 0; N := 0;
  READLN(InData, Num2, Num1);
  WHILE (Num1 <> -999) AND (Num2 <> -999) DO
   BEGIN
    N := N + 1;      X[N] := Num1;          Y[N] := Num2;
    SumX := SumX + X[N];
    SumX2 := SumX2 + X[N] * X[N];
    SumY := SumY + Y[N];
    SumY2 := SumY2 + Y[N] * Y[N];
    SumXY := SumXY + X[N] * Y[N];
    READLN(InData, Num2, Num1);
   END;
  A := (SumY*(SumX2-SumX*SumX/N)+SumX*(SumX*SumY/N-SumXY)) /
                                                        (N*SumX2-SumX*SumX);
  B := (N*SumXY-SumX*SumY)/(N*SumX2-SumX*SumX);
  R := (N*SumXY-SumX*SumY)/SQRT((N*SumX2-SumX*SumX)*(N*SumY2-SumY*SumY));
 END;

PROCEDURE WriteData (N : INTEGER; Data : XYARRAY);
 VAR
  I : INTEGER;
 BEGIN
  FOR I := 1 TO N DO
   WRITE(Lst, Data[I]:7:3);
  WRITELN(Lst)
 END;

PROCEDURE DoProteinStandard (VAR A, B : REAL);
 VAR
  N : INTEGER;
  X, Y : XYARRAY;
  XName, YName : STRING13;
  R : REAL;
 BEGIN
  DoRegression(N, XName, YName, X, Y, A, B, R);
  WRITE(Lst, 'Protein standard:':22, YName);
  WriteData(N, Y);
  WRITE(Lst, ' ':22, XName, ' ':4);
  WriteData(N, X);          WRITELN(Lst);
  WRITE(Lst, ' ':10, 'Total proteins (ug) =', A:8:4, ' + ', B:8:4, ' * OD595');
  WRITELN(Lst, 'R = ':12, R:5:3);
 END;
```

```
 PROCEDURE PrintLine;
  VAR
    I : INTEGER;
  BEGIN
    FOR I := 1 TO 5 DO
      WRITE(Lst, ' ');
    FOR I := 1 TO 70 DO
      WRITE(Lst, '_');
    WRITELN(Lst);
  END;

 PROCEDURE GetLacZActivity (A, B : REAL; VAR Group : INTEGER; VAR Num : LINEARRAY;
                            VAR Medium : MEDIUMARRAY; VAR SampleName : SAMPLEARRAY;
                            VAR SpAct : DataArray);
  VAR
    Line, Time : INTEGER;
    Signal : CHAR;
    OD595, OD420, Proteinmg : REAL;
  BEGIN
    Group := 0;
    WHILE NOT EOF(InData) DO
      BEGIN
        REPEAT
          READLN(InData, Signal);
        UNTIL (Signal = '@') OR EOF(InData);
        IF NOT EOF(InData) THEN
          BEGIN
            Group := Group + 1;
            READLN(InData, Medium[Group], Time);
            WRITELN(Lst, '     ', Medium[Group], '-', Time, ' min:');
            REPEAT
              READLN(InData, Signal);
            UNTIL Signal = '@';
            Line := 1;
            READLN(InData, SampleName[Group, Line], OD420, OD595);
            WHILE (OD420 <> -999) AND (OD595 <> -999) DO
              BEGIN
                Proteinmg := (A + B * OD595) / 1000;
                SpAct[Group, Line] := OD420/Time/0.0045/Proteinmg/4;
                WRITELN(Lst, SampleName[Group, Line]:30, OD420:10:3, OD595:10:3, Proteinmg:10:3,
SpAct[Group, Line] :15:3);
                Line := Line + 1;
                READLN(InData, SampleName[Group, Line], OD420, OD595)
              END;
            Num[Group] := Line - 1
          END
      END
  END;
```

```
  PROCEDURE PrintTableHead;
    BEGIN
      WRITELN(Lst);
      PrintLine;
      WRITELN(Lst, 'Specific':75);
      WRITELN(Lst, 'Sample':30, 'OD420':10, 'OD595':10, 'Protein':10, 'Activity':15);
      WRITELN(Lst, '(mg)':59, '(U/mg)':14);
      PrintLine;
    END;

  PROCEDURE SaveResult (Group : INTEGER; Num : LINEARRAY; Medium : MEDIUMARRAY;
                                  SampleName : SAMPLEARRAY; SpAct : DataArray);
    VAR
      I, J, N : INTEGER;
    BEGIN
      ASSIGN(OutData, 'RESULT.PAS');
      REWRITE(OutData);
      FOR I := 1 TO GROUP DO
        BEGIN
          WRITELN(OutData, Medium[I]);
          WRITELN(OutData, 'SampleName':25, 'Specific Activity':20);
          N := Num[I];
          FOR J := 1 TO N DO
             WRITELN(OutData, SampleName[I, J], SpAct[I, J]:20:3);
          WRITELN(OutData, '@':25, '-999':20)
        END;
      Close(InData);
      Close(OutData)
    END;

(*****************************    MAIN    LOGIC    *****************************)

BEGIN
  PrintCenteredTitle;
  GetInformation;
  ASSIGN(InData, 'NONAME.PAS');
  RESET(InData);
  DoProteinStandard(A, B);
  PrintTableHead;
  GetLacZActivity(A, B, Group, Num, Medium, SamppleName, SpAct);
  SaveResult(Group, Num, Medium, SamppleName, SpAct);
  PrintLine;
END.
```

```
PROGRAM Derepression (INPUT, OUTPUT);

  USES PRINTER;

 TYPE
   ARRAYTYPE = ARRAY [1..30] OF REAL;

 VAR
   SpeAct1 : ARRAYTYPE;
   SpeAct2, Ratio : REAL;
   N, I : INTEGER;
   Name : STRING[25];
   Infile :TEXT;

BEGIN
  ASSIGN(Infile, 'Result.pas');
  RESET(Infile);
  READLN(Infile);  READLN(Infile);
  N := 1;
  READLN(Infile, Name, SpeAct1[N]);
  WHILE SpeAct1[N] <> -999 DO
    BEGIN
      N := N + 1;
     READLN(Infile, Name, SpeAct1[N]);
    END;
  READLN(Infile);  READLN(Infile);
  WRITELN(Lst, 'SampleName' :30, 'Derepression Ratio':20);
  WRITELN(Lst);
  FOR I := 1 TO (N - 1) DO
    BEGIN
     READLN(Infile, Name, SpeAct2);
     Ratio := SpeAct2 / SpeAct1[I];
     WRITELN(Lst, Name :30, Ratio :20:2);
    END;
  Close(Infile);
END.
```