

| | | |
|-----------|------|------------|
| PRINT.BAS | 0.23 | 10/21/2022 |
|-----------|------|------------|

DESCRIPTION :

- Program for Generating Historical Time Tables
- Program component for several output formats

REFERENCES :

9. Remember the former things of old: for I [am] God, and [there is] none else; [I am] God, and [there is] none like me,
10. Declaring the end from the beginning, and from ancient times [the things] that are not [yet] done, saying, My counsel shall stand, and I will do all my pleasure:
(Isaiah 46)
18. For verily I say unto you, Till heaven and earth pass, one jot or one tittle shall in no wise pass from the law, till all be fulfilled.
(St. Matthew 5)

The Bible

"The Bible, Authorized Version" by King James 1769, and Webster Update 1833, Oxford University Press, 1994

[Mar2002]

R. Maran: "Das neue Einsteigerbuch Internet - HTML-Webseiten" ("The New Starter Book Internet - HTML Web Pages") Serges Medien Köln, (2002)

[Kop2002]

H. Kopka: "LaTeX", vol. 1: "Einführung" ("Introduction"), Pearson Studium München, 3rd overworked edition, (2002)

HANDLING :

| | |
|-----------------------|-----------------|
| 2/ 8/2003 - 2/17/2004 | Norbert Südland |
| 4/ 5/2007 - 3/21/2017 | Norbert Südland |
| 10/21/2022 | Norbert Südland |

PREPARATION

Declarations

```

DECLARE FUNCTION SIZEOF% (StructureName$)
DECLARE FUNCTION STRLEN% (Text$, EndCharacter$)
DECLARE FUNCTION QuickPosition% (NameList$(), Name$)
DECLARE FUNCTION Part$ (Data$, Variable$)
DECLARE FUNCTION KeyInput$ ()
DECLARE FUNCTION ToUser$ (Symbol$)
DECLARE FUNCTION Load$ (File%, RecordLength%, FileLength%, Position%)

```

```

DECLARE SUB Present (Colors%, Quest$, Buf$, Of%, Kind$, Area%, Areas%, ly%)
DECLARE SUB Pause ()

```

| |
|-----------|
| Variables |
|-----------|

```

OPTION BASE 1                'Areas start by entry No.`1`!
'OPTION EXPLICIT             'Useful with VisualBasic

DIM InputFile$               'AS STRING
DIM FormatChoice$(8)
DIM Out$
DIM Program$
DIM Buffer$
DIM PositionFile$
DIM Text$
DIM Name$
DIM Data$
DIM Variable$
DIM Simultaneous$
DIM DateValue$
DIM CopyData$
DIM Align$

DIM Success%                 'AS INTEGER
DIM MenuChoice%
DIM ReadingTrial%
DIM Configuration%
DIM Position%
DIM l%                       'Length
DIM GlobVarNumber%
DIM DataLength%
DIM InputFile%
DIM y%
DIM PrintFile%
DIM Heading%
DIM x%
DIM ASCII%
DIM Lines%

DIM GIL&                     'AS LONG
DIM c&                       'counter
DIM Number&

```

| |
|----------------|
| First Commands |
|----------------|

```

COMMON HistoricChoice%, WorkingPlace$, WorkingTime$, CountingMode%

```

```

DIM HTML%(128)              'AS INTEGER

REDIM SHARED GVBEGIN%(1)    'AS INTEGER

```

```

REDIM SHARED GVLength%(1)      'AS INTEGER
REDIM SHARED GVName$(1)        'AS STRING
REDIM SHARED GVType$(1)        'AS STRING

```

Constants

```

CONST ConfigurationFile$ = "PRINT.CFG"
CONST PrintFlag$ = "PRINT.YES"

```

MAIN PART

Choice of the Format

'Preparation:

'-----'

```
CLEAR , , 4096
```

```
ON ERROR GOTO ErrorHandler
```

```
GOSUB ProgramStart
```

'Ask for Format to Present:

'-----'

```
Success% = 0
```

```
MenuChoice% = 1
```

```
WHILE Success% = 0 OR MenuChoice% > 0
```

```
CLS
```

```
Present 2, "(9)... Choose Output Format:", "", 0, "C", 1, 1, 1
```

```
Present 1, "Working Directory: ", WorkingPlace$, 0, "C", 1, 1, 5
```

```
Present 1, "Data File: ", InputFile$, 0, "C", 1, 1, 6
```

```
FOR y% = 1 TO 8
```

```
IF y% MOD 8 = MenuChoice% THEN
```

```
Present 1, "", FormatChoice$(y%), 0, "C", 1, 1, y% + 10
```

```
ELSE
```

```
Present 1, " " + FormatChoice$(y%) + " ", "", 0, "C", 1, 1, y% + 10
```

```
END IF
```

```
NEXT y%
```

```
Present 1, "Your Choice: ", LTRIM$(STR$(MenuChoice%)), 0, "C", 1, 1, 20
```

```
LOCATE CSRLIN, POS(0) - 2
```

```
Success% = 0
```

```
Out$ = KeyInput$
```

```
SELECT CASE ASC(Out$)
```

```
CASE ASC("0") TO ASC("7")
```

```
MenuChoice% = VAL(Out$) MOD 8
```

```
Success% = 1
```

```
CASE 10, 13
```

```
Success% = 1
```

```
CASE 27
```

```
Success% = 0
```

```
MenuChoice% = (MenuChoice% + 1) MOD 8
```

```
IF MenuChoice% = 1 THEN
```

```

        MenuChoice% = 0
        Success% = 1
    END IF
END SELECT
IF ASC(Out$) = 0 THEN
    PRINT ASC(RIGHT$(Out$, 1))
    SELECT CASE ASC(RIGHT$(Out$, 1))
    CASE 59 TO 63
        MenuChoice% = ASC(RIGHT$(Out$, 1)) - 58
        Success% = 1
    CASE 71, 73    'Home, Page Up
        MenuChoice% = 1
    CASE 72, 75    'Cursor Up, Cursor Left
        MenuChoice% = (MenuChoice% + 7) MOD 8
    CASE 77, 80    'Cursor Right, Cursor Down
        MenuChoice% = (MenuChoice% + 1) MOD 8
    CASE 79, 81    'End, Page Down
        MenuChoice% = 0
    END SELECT
END IF
IF Success% = 1 THEN
    Text$ = "Your Choice: "
    Present 1, Text$, LTRIM$(STR$(MenuChoice%)), 0, "C", 1, 1, 20
    FOR y% = 1 TO 8
        Text$ = FormatChoice$(y%)
        IF y% MOD 8 = MenuChoice% THEN
            Present 1, "", Text$, 0, "C", 1, 1, y% + 10
        ELSE
            Present 1, " " + Text$ + " ", "", 0, "C", 1, 1, y% + 10
        END IF
    NEXT y%
    ON MenuChoice% GOSUB Code437, LaTeX5pt, LaTeX7pt, LaTeX11pt, CSV, HTML,
Mathematica
    END IF
WEND

'=====
ProgramEnd:
'=====
'Present Eventually Error Number:
'-----'
IF ERR > 0 THEN
    PRINT "Error No."; ERR; " occurred in line"; ERL; "."
    Pause
END IF

'Close All Files:
'-----'
CLOSE

'Return to the Calling Program:
'-----'
IF Program$ <> "" THEN CHAIN Program$
SYSTEM
'_____ END OF THE MAIN PART _____'

'=====
```

ERROR HANDLING

```

=====
ErrorHandling:
'=====
  SELECT CASE ERR
CASE 53          'File not found:
  SELECT CASE ERL
CASE 10
  PRINT "Configuration file "; ConfigurationFile$; " is missing."
  GOTO ProgramEnd
CASE 30
  PRINT "Position file HISTORIC.POS is missing."
  ReadingTrial% = ReadingTrial% + 1
  RESUME 21
CASE ELSE
  Pause
  RESUME NEXT
END SELECT
CASE 75          'Path /File Access Error:
  SELECT CASE ERL
CASE 5          'Writing tests onto `WorkingPlace$`
  RESUME NEXT
CASE ELSE
  Pause
  RESUME
END SELECT
CASE 76          'Path not found:
  SELECT CASE ERL
CASE 5
  PRINT "The working directory " + CHR$(34) + WorkingPlace$ + CHR$(34)
  PRINT "has not been found. This program part is aborted."
  Pause
  GOTO ProgramEnd
CASE ELSE
  Pause
  RESUME
END SELECT
END SELECT

  PRINT "Error No."; ERR; " occurred in line"; ERL
ON ERROR GOTO 0
GOTO ProgramEnd
'
                                END OF THE ERROR HANDLING

```

SUBROUTINES VIA GOSUB

```
'====='
```

ProgramStart:

```
'====='
```

CLS

' As `WorkingPlace\$` the working directory is used, which is given by
' %HISTORICTEMP%, %TEMP%, or %TMP%.

```

' If (with old DOS versions) not any `WorkingPlace$` is mentioned,
' the trial is to write onto the medium, which also contains the program.
' Eventually the program terminates, if the `WorkingPlace$` is not
' writable:
' -----
WorkingPlace$ = ENVIRON$("HISTORICTEMP")
IF WorkingPlace$ = "" THEN
    WorkingPlace$ = ENVIRON$("QBASICTEMP")
    IF WorkingPlace$ <> "" THEN WorkingPlace$ = WorkingPlace$ + "\"
5    MKDIR WorkingPlace$ + "HISTORIC.TMP"
    WorkingPlace$ = WorkingPlace$ + "HISTORIC.TMP\"
ELSE
    IF WorkingPlace$ <> "" THEN WorkingPlace$ = WorkingPlace$ + "\"
END IF

'Check Writability of `WorkingPlace$`:
' -----
BSAVE WorkingPlace$ + "HISTORIC.CHK", 0, 0
KILL WorkingPlace$ + "HISTORIC.CHK"

Configuration% = FREEFILE
10 OPEN WorkingPlace$ + ConfigurationFile$ FOR INPUT AS #Configuration%
20 LINE INPUT #Configuration%, InputFile$
   LINE INPUT #Configuration%, Buffer$
   GIL% = VAL(Buffer$)
   LINE INPUT #Configuration%, Program$
CLOSE #Configuration%

'Read in the Structure Positions:
' -----
Position% = FREEFILE
PRINT "Read in the structure positions from HISTORIC.POS..."
ReadingTrial% = 1
21 SELECT CASE ReadingTrial%
CASE 1
    PositionFile$ = "HISTORIC.POS"
CASE 2
    PositionFile$ = WorkingPlace$ + PositionFile$
CASE 3
    Configuration% = FREEFILE
    OPEN "HISTORIC.STR" FOR INPUT AS #Configuration%
    CLOSE #Configuration%
22 OPEN WorkingPlace$ + "STRUKTUR.CFG" FOR OUTPUT AS #Configuration%
    PRINT #Configuration%, "HISTORIC.STR"      'Structure file
    PRINT #Configuration%, PositionFile$      'The very position file
    PRINT #Configuration%, "PRINT.BAS"        'Return program
    CLOSE #Configuration%
    CHAIN "STRUKTUR.BAS"
CASE ELSE
    Pause                                     'Programming or translation mistake!
END SELECT
30 OPEN PositionFile$ FOR INPUT AS #Position%
40 LINE INPUT #Position%, Text$
    l% = STRLEN$(Text$, " ")
    GlobVarNumber% = VAL(LEFT$(Text$, l%))
    PRINT LTRIM$(STR$(GlobVarNumber%)); " Structure Elements"
    REDIM SHARED GVBEGIN%(GlobVarNumber%)      'AS INTEGER
    REDIM SHARED GVLENGTH%(GlobVarNumber%)     'AS INTEGER

```

```

        REDIM SHARED GVName$(GlobVarNumber%)      'AS STRING
        REDIM SHARED GVType$(GlobVarNumber%)      'AS STRING
        FOR y% = 1 TO GlobVarNumber%
50          INPUT #Position%, GVBegin%(y%)
            INPUT #Position%, GVLength%(y%)
            INPUT #Position%, GVName$(y%)
            INPUT #Position%, GVType$(y%)
        NEXT y%
        CLOSE #Position%
        DataLength% = SIZEOF%("data")
        IF DataLength% = 0 THEN ERROR 100

        'Open Data File:
        '-----'
        InputFile% = FREEFILE
        Name$ = WorkingPlace$ + InputFile$
        OPEN Name$ FOR BINARY ACCESS READ WRITE AS #InputFile%

        'List of the Output Formats (Completable):
        '-----'
        FormatChoice$(1) = "(1).. ASCII (Codepage 437)      "
        FormatChoice$(2) = "(2).. LaTeX (Leslie Lamport TeX, 5pt) "
        FormatChoice$(3) = "(3).. LaTeX (Landscape, 7pt)    "
        FormatChoice$(4) = "(4).. LaTeX (Larger, 11pt)      "
        FormatChoice$(5) = "(5).. Comma Separated Values (CSV) "
        FormatChoice$(6) = "(6).. Hypertext Markup Language (HTML) "
        FormatChoice$(7) = "(7).. Mathematica List Format    "
        FormatChoice$(8) = "(0).. Return to the Main Program "

        FOR y% = 1 TO 128
            READ HTML%(y%)
        NEXT y%

DATA 8962,199,252,233,226,228,224,229,231,234,235,232,239,238,236,196
DATA 197,201,230,198,244,246,242,251,249,255,214,220,162,163,165,8359
DATA 131,225,237,243,250,241,209,170,186,191,8976,172,189,188,161,171
DATA 187,9617,9618,9619,9474,9508,9569,9570,9558,9557,9571,9553,9559,9565
DATA 9564,9563
DATA 9488,9492,9524,9516,9500,9472,9532,9566,9567,9562,9556,9577,9574,9568
DATA 9552,9580
DATA 9575,9576,9572,9573,9561,9560,9554,9555,9579,9578,9496,9484,9608,9604
DATA 9612,9616
DATA 9600,945,223,915,960,931,963,181,964,934,952,937,948,8734,248,949
DATA 8745,8801,177,8805,8804,8992,8993,247,8776,176,149,183,8730,8319,178
DATA 9642

RETURN 'ProgramStart _____'

'=====
Code437:
'=====

'Generate the Firt Output File:
'-----'
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".PR1 is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".PR1"

```

```

90 OPEN Buffer$ FOR OUTPUT AS #PrintFile%
   Heading% = FREEFILE
91 OPEN "HISTORIC.P_1" FOR INPUT AS #Heading%
   WHILE EOF(Heading%) = 0
       LINE INPUT #Heading%, Text$
       PRINT #PrintFile%, Text$
   WEND
CLOSE #Heading%
FOR c& = 1 TO GIL&
   Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
   FOR y% = 1 TO 6
       Variable$ = "data.date[" + LTRIM$(STR$(y%)) + "]"
       PRINT #PrintFile%, "||" + Part$(Data$, Variable$);
   NEXT y%
   PRINT #PrintFile%, "||"

   IF c& MOD 60 = 0 OR c& = GIL& THEN
       PRINT #PrintFile%, "===== ";
       FOR y% = 2 TO 6
           PRINT #PrintFile%, "||===== ";
       NEXT y%
       PRINT #PrintFile%, "||"
       IF c& < GIL& THEN
           PRINT #PrintFile%, CHR$(12)
           Heading% = FREEFILE
           OPEN "HISTORIC.P_1" FOR INPUT AS #Heading%
           WHILE EOF(Heading%) = 0
               LINE INPUT #Heading%, Text$
               PRINT #PrintFile%, Text$
           WEND
           CLOSE #Heading%
       END IF
   ELSEIF c& MOD 10 = 0 THEN
       PRINT #PrintFile%, "||----- ";
       FOR y% = 2 TO 6
           PRINT #PrintFile%, "||----- ";
       NEXT y%
       PRINT #PrintFile%, "||"
   ELSEIF c& MOD 5 = 0 THEN
       FOR y% = 1 TO 6
           PRINT #PrintFile%, "||----- ";
       NEXT y%
       PRINT #PrintFile%, "||"
   END IF
NEXT c&
CLOSE #PrintFile%

'Generate Second Output File:
'-----'
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".PR2 is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".PR2"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
   Heading% = FREEFILE
92 OPEN "HISTORIC.P_2" FOR INPUT AS #Heading%
   WHILE EOF(Heading%) = 0
       LINE INPUT #Heading%, Text$

```

```

    PRINT #PrintFile%, Text$
WEND
CLOSE #Heading%
FOR c& = 1 TO GIL&
    Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
    PRINT #PrintFile%, "|| " + Part$(Data$, "data.name") + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.p[1].moment1"))
    PRINT #PrintFile%, Buffer$ + " ";
    Buffer$ = ToUser$(Part$(Data$, "data.p[1].direct"))
    PRINT #PrintFile%, Buffer$ + " ";
    PRINT #PrintFile%, Part$(Data$, "data.p[1].name") + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.p[1].moment2"))
    PRINT #PrintFile%, Buffer$ + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.p[2].moment1"))
    PRINT #PrintFile%, Buffer$ + " ";
    Buffer$ = ToUser$(Part$(Data$, "data.p[2].direct"))
    PRINT #PrintFile%, Buffer$ + " ";
    PRINT #PrintFile%, Part$(Data$, "data.p[2].name") + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.p[2].moment2"))
    PRINT #PrintFile%, Buffer$ + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.source") + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.r[1].tol1") + " ";
    PRINT #PrintFile%, Part$(Data$, "data.r[1].date") + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.r[1].moment1"))
    PRINT #PrintFile%, Buffer$ + " ";
    PRINT #PrintFile%, Part$(Data$, "data.r[1].name") + " || "

    IF c& MOD 60 = 0 OR c& = GIL& THEN
        PRINT #PrintFile%, "||=====||";
        PRINT #PrintFile%, "||=====||";
        PRINT #PrintFile%, "||=====||";
        PRINT #PrintFile%, "||"
        IF c& < GIL& THEN
            PRINT #PrintFile%, CHR$(12)
            Heading% = FREEFILE
            OPEN "HISTORIC.P_2" FOR INPUT AS #Heading%
            WHILE EOF(Heading%) = 0
                LINE INPUT #Heading%, Text$
                PRINT #PrintFile%, Text$
            WEND
            CLOSE #Heading%
        END IF
    ELSEIF c& MOD 10 = 0 THEN
        PRINT #PrintFile%, "||=====||";
        PRINT #PrintFile%, "||=====||";
        PRINT #PrintFile%, "||=====||";
        PRINT #PrintFile%, "||"
    ELSEIF c& MOD 5 = 0 THEN
        PRINT #PrintFile%, "||-----||-----";
        PRINT #PrintFile%, "||-----||-----||-----";
        PRINT #PrintFile%, "||-----||-----||-----";
        PRINT #PrintFile%, "||-----||"
    END IF
NEXT c&
CLOSE #PrintFile%

'Generate Third Output File:
'-----'

```

```

Text$ = "Output file " + LEFT$(InputFile$, 8) + ".PR3 is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".PR3"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
    Heading% = FREEFILE
93  OPEN "HISTORIC.P_3" FOR INPUT AS #Heading%
    WHILE EOF(Heading%) = 0
        LINE INPUT #Heading%, Text$
        PRINT #PrintFile%, Text$
    WEND
CLOSE #Heading%
FOR c& = 1 TO GIL&
    Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
    Buffer$ = ToUser$(Part$(Data$, "data.r[1].moment2"))
    PRINT #PrintFile%, "|| " + Buffer$ + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.r[1].tol2") + " ";
    PRINT #PrintFile%, Part$(Data$, "data.r[1].duration") + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.r[1].source") + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.r[2].tol1") + " ";
    PRINT #PrintFile%, Part$(Data$, "data.r[2].date") + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.r[2].moment1"))
    PRINT #PrintFile%, Buffer$ + " ";
    PRINT #PrintFile%, Part$(Data$, "data.r[2].name") + " || ";
    Buffer$ = ToUser$(Part$(Data$, "data.r[2].moment2"))
    PRINT #PrintFile%, Buffer$ + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.r[2].tol2") + " ";
    PRINT #PrintFile%, Part$(Data$, "data.r[2].duration") + " || ";
    Simultaneous$ = ToUser$(Part$(Data$, "data.simultaneous"))
    PRINT #PrintFile%, MID$(Simultaneous$, 1, 1) + " ";
    PRINT #PrintFile%, MID$(Simultaneous$, 2, 1) + " ";
    PRINT #PrintFile%, MID$(Simultaneous$, 3, 1) + " || ";
    PRINT #PrintFile%, MID$(Simultaneous$, 4, 1) + " ";
    PRINT #PrintFile%, MID$(Simultaneous$, 5, 1) + " || ";
    PRINT #PrintFile%, Part$(Data$, "data.r[2].source") + " || "

    IF c& MOD 60 = 0 OR c& = GIL& THEN
        PRINT #PrintFile%, "||_____||";
        PRINT #PrintFile%, "||_____||";
        PRINT #PrintFile%, "||_____||";
        PRINT #PrintFile%, "||_____||"
        IF c& < GIL& THEN
            PRINT #PrintFile%, CHR$(12)
            Heading% = FREEFILE
            OPEN "HISTORIC.P_3" FOR INPUT AS #Heading%
                WHILE EOF(Heading%) = 0
                    LINE INPUT #Heading%, Text$
                    PRINT #PrintFile%, Text$
                WEND
            CLOSE #Heading%
        END IF
    ELSEIF c& MOD 10 = 0 THEN
        PRINT #PrintFile%, "||_____||";
        PRINT #PrintFile%, "||_____||";
        PRINT #PrintFile%, "||_____||";
        PRINT #PrintFile%, "||_____||"
    ELSEIF c& MOD 5 = 0 THEN
        PRINT #PrintFile%, "||---||-----||-----||";

```

```

        PRINT #PrintFile%, "----||-----||-----";
        PRINT #PrintFile%, "----||---||-----||-----||-----||---";
        PRINT #PrintFile%, "-----||"
    END IF
NEXT c&
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Present 2, "Please Exit the Program for Printing.", "", 0, "C", 1, 1, 24
Pause
PrintFile% = FREEFILE
OPEN WorkingPlace$ + PrintFlag$ FOR OUTPUT AS #PrintFile%
CLOSE #PrintFile%
'-----'
' Properly here should be programmed the output via QBASIC.
' However, the LPRINT of QBASIC can only manage 80 characters per line.
' Even the primitive BASIC of the Alpatronic PC was more successful...
'
' For the settingggg of the printer is small printing, subsrcript, and
' enlarged character set necessary, thus 160 characters can be printed
' in each line.
' For the EPSON LG500 the setting of the printer is the following:
' LPRINT CHR$(27) + "@" + CHR$(15) + CHR$(27) + CHR$(83) + CHR$(1);
' LPRINT CHR$(27) + "l" + CHR$(0) + CHR$(27) + "t" + CHR$(1);
' LPRINT CHR$(27) + "0" + CHR$(17);
' Afterwards you can use the DOS command "PRINT" to print out.
'
' Under Windows the command `PRINT` has been given up!
' It is better to switch over to the other data formats.
'-----'

RETURN 'Code437'

'=====
LaTeX5pt:
'=====
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".TeX is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".TeX"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
PRINT #PrintFile%, "\documentclass[10pt]{article}"
PRINT #PrintFile%, "\usepackage{geometry}"
PRINT #PrintFile%, "\usepackage[cp437de]{inputenc}"
PRINT #PrintFile%, "\usepackage{latexsym}"
PRINT #PrintFile%, "\setlength{\parindent}{0pt}"
PRINT #PrintFile%, "\setlength{\topmargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\headheight}{0mm}"
PRINT #PrintFile%, "\setlength{\headsep}{0mm}"
PRINT #PrintFile%, "\setlength{\topskip}{0mm}"
PRINT #PrintFile%, "\setlength{\textheight}{229.4mm}"
PRINT #PrintFile%, "\setlength{\oddsidemargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\evensidemargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\textwidth}{175.9mm}"
PRINT #PrintFile%, "\begin{document}\tt"

Lines% = 100

```

```

GOSUB GenerateLaTeXFile

PRINT #PrintFile%, "\end{document}"
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Text$ = "Please terminate the program for compiling."
Present 2, Text$, "", 0, "C", 1, 1, 24
Pause
RETURN 'LaTeX5pt _____'

'=====
LaTeX7pt:
'=====
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".TeX is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".TeX"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
PRINT #PrintFile%, "\documentclass[10pt,landscape]{article}"
PRINT #PrintFile%, "\usepackage{geometry}"
PRINT #PrintFile%, "\usepackage[cp437de]{inputenc}"
PRINT #PrintFile%, "\usepackage{latexsym}"
PRINT #PrintFile%, "\setlength{\parindent}{0pt}"
PRINT #PrintFile%, "\setlength{\topmargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\headheight}{0mm}"
PRINT #PrintFile%, "\setlength{\headsep}{0mm}"
PRINT #PrintFile%, "\setlength{\topskip}{0mm}"
PRINT #PrintFile%, "\setlength{\textheight}{160mm}"
PRINT #PrintFile%, "\setlength{\oddsidemargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\evensidemargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\textwidth}{239.4mm}"
PRINT #PrintFile%, "\begin{document}\tt"

Lines% = 50
GOSUB GenerateLaTeXFile

PRINT #PrintFile%, "\end{document}"
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Text$ = "Please terminate the program for compiling."
Present 2, Text$, "", 0, "C", 1, 1, 24
Pause
RETURN 'LaTeX7pt _____'

'=====
LaTeX11pt:
'=====
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".TeX is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".TeX"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
PRINT #PrintFile%, "\documentclass[11pt]{article}"
PRINT #PrintFile%, "\usepackage{geometry}"

```

```

PRINT #PrintFile%, "\usepackage[cp437de]{inputenc}"
PRINT #PrintFile%, "\usepackage{latexsym}"
PRINT #PrintFile%, "\setlength{\parindent}{0pt}"
PRINT #PrintFile%, "\setlength{\topmargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\headheight}{0mm}"
PRINT #PrintFile%, "\setlength{\headsep}{0mm}"
PRINT #PrintFile%, "\setlength{\topskip}{0mm}"
PRINT #PrintFile%, "\setlength{\textheight}{229.4mm}"
PRINT #PrintFile%, "\setlength{\oddsidemargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\evensidemargin}{-5.4mm}"
PRINT #PrintFile%, "\setlength{\textwidth}{180mm}"
PRINT #PrintFile%, "\begin{document}\tt"
Lines% = 40
FOR c& = 1 TO GIL&
  Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
  IF (c& - 1) MOD Lines% = 0 THEN
    IF c& > 1 THEN
      PRINT #PrintFile%, "\newpage"
    END IF
    Text$ = "Datings (1/6) of " + RTRIM$(LEFT$(InputFile$, 8))
    Text$ = Text$ + ".HQL ("
    Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
    Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
    Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
    Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
    GOSUB LaTeXOutput
    PRINT #PrintFile%, "\\"
    PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
    Text$ = "\begin{tabular}{*{3}{|c@{}r@{}l@{}r@{}l@{}r@{}l@{}c@{}}}"
    Text$ = Text$ + "c@{}r@{}l@{}r@{}l@{}r@{}l|| \hline"
    PRINT #PrintFile%, Text$
    PRINT #PrintFile%, "\multicolumn{15}{|c|}{Begin of Life [$*]$ } &"
    Text$ = "\multicolumn{15}{|c|}{Begin \ensuremath{1^{\mathrm{st}}}"
    Text$ = Text$ + " Periode [B] } &"
    PRINT #PrintFile%, Text$
    Text$ = "\multicolumn{15}{|c|}{Begin \ensuremath{2^{\mathrm{nd}}}"
    Text$ = Text$ + " Periode ["
    PRINT #PrintFile%, Text$;
    Text$ = "α"
    GOSUB LaTeXOutput
    PRINT #PrintFile%, "]" }\\
    PRINT #PrintFile%, "\hline"
  ELSEIF (c& - 1) MOD 5 = 0 THEN
    PRINT #PrintFile%, "\hline"
  END IF
  FOR x% = 1 TO 3
    Variable$ = "data.date[" + LTRIM$(STR$(x%)) + "]"
    DateValue$ = Part$(Data$, Variable$)
    Text$ = Part$(DateValue$, "date.minimum.sign")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.year")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.ys")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.month")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.ms")
    GOSUB LaTeXTable
  
```

```

Text$ = Part$(DateValue$, "date.minimum.day")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.minimum.ds")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.status")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(DateValue$, "date.maximum.sign")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.maximum.year")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.maximum.ys")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.maximum.month")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.maximum.ms")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.maximum.day")
GOSUB LaTeXTable
Text$ = Part$(DateValue$, "date.maximum.ds")
GOSUB LaTeXOutput
IF x% < 3 THEN
    PRINT #PrintFile%, " & "
ELSE
    PRINT #PrintFile%, " \\"
END IF
NEXT x%
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
    PRINT #PrintFile%, "\hline"
    PRINT #PrintFile%, "\end{tabular} \hspace*{\fill}"
END IF
NEXT c&

FOR c& = 1 TO GIL&
    Data$ = Load$(InputFile$, DataLength%, GIL&, c&)
    IF (c& - 1) MOD Lines% = 0 THEN
        PRINT #PrintFile%, "\newpage"
        Text$ = "Datings (2/6) of " + RTRIM$(LEFT$(InputFile$, 8))
        Text$ = Text$ + ".HQL ("
        Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
        Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
        GOSUB LaTeXOutput
        PRINT #PrintFile%, "\\"
        PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
        Text$ = "\begin{tabular}{*{3}{|c@{}r@{}l@{}r@{}l@{}r@{}l@{}c@{}}}"
        Text$ = Text$ + "c@{}r@{}l@{}r@{}l@{}r@{}l|| \hline"
        PRINT #PrintFile%, Text$
        Text$ = "\multicolumn{15}{|c|}{End \ensuremath{2^{\mathrm{nd}}}}}"
        Text$ = Text$ + " Periode ["
        PRINT #PrintFile%, Text$;
        Text$ = "Ω"
        GOSUB LaTeXOutput
        PRINT #PrintFile%, "]" & "
        Text$ = "\multicolumn{15}{|c|}{End \ensuremath{1^{\mathrm{st}}}}}"
        Text$ = Text$ + " Periode [E] } &"
        PRINT #PrintFile%, Text$
        PRINT #PrintFile%, "\multicolumn{15}{|c|}{End of Life [+]} \\"
    
```

```

    PRINT #PrintFile%, "\hline"
ELSEIF (c& - 1) MOD 5 = 0 THEN
    PRINT #PrintFile%, "\hline"
END IF
FOR x% = 4 TO 6
    Variable$ = "data.date[" + LTRIM$(STR$(x%)) + "]"
    DateValue$ = Part$(Data$, Variable$)
    Text$ = Part$(DateValue$, "date.minimum.sign")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.year")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.ys")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.month")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.ms")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.day")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.minimum.ds")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.status")
    GOSUB LaTeXTableSpecialCharacter
    Text$ = Part$(DateValue$, "date.maximum.sign")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.maximum.year")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.maximum.ys")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.maximum.month")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.maximum.ms")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.maximum.day")
    GOSUB LaTeXTable
    Text$ = Part$(DateValue$, "date.maximum.ds")
    GOSUB LaTeXOutput
    IF x% < 6 THEN
        PRINT #PrintFile%, " & "
    ELSE
        PRINT #PrintFile%, " \\"
    END IF
NEXT x%
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
    PRINT #PrintFile%, "\hline"
    PRINT #PrintFile%, "\end{tabular} \hspace*{\fill}"
END IF
NEXT c&

FOR c& = 1 TO GIL&
    Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
    IF (c& - 1) MOD Lines% = 0 THEN
        PRINT #PrintFile%, "\newpage"
        Text$ = "Content (3/6) of " + RTRIM$(LEFT$(InputFile$, 8))
        Text$ = Text$ + ".HQL ("
        Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
        Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1

```

```

Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
GOSUB LaTeXOutput
PRINT #PrintFile%, "\\\"
PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
Text$ = "\begin{tabular}{|l|}"
Text$ = Text$ + "c@{\hspace{0.5em}}c@{\hspace{0.5em}}}"
Text$ = Text$ + "l@{\hspace{0.5em}}c|"
Text$ = Text$ + "c@{\hspace{0.5em}}c@{\hspace{0.5em}}l"
Text$ = Text$ + "@{\hspace{0.5em}}c|"
Text$ = Text$ + "} \hline"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\multicolumn{1}{|c}{Name /Event} &"
Text$ = "\multicolumn{4}{|c}{\ensuremath{1^{\mathrm{st}}}"
Text$ = Text$ + " Predecessor} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{4}{|c}{\ensuremath{2^{\mathrm{nd}}}"
Text$ = Text$ + " Predecessor} \\\"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\hline"
ELSEIF (c& - 1) MOD 5 = 0 THEN
PRINT #PrintFile%, "\hline"
END IF
Text$ = Part$(Data$, "data.name")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.p[1].moment1"))
GOSUB LaTeXTableSpecialCharacter
Text$ = ToUser$(Part$(Data$, "data.p[1].direct"))
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.p[1].name")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.p[1].moment2"))
GOSUB LaTeXTableSpecialCharacter
Text$ = ToUser$(Part$(Data$, "data.p[2].moment1"))
GOSUB LaTeXTableSpecialCharacter
Text$ = ToUser$(Part$(Data$, "data.p[2].direct"))
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.p[2].name")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.p[2].moment2"))
GOSUB LaTeXTableSpecialCharacter
PRINT #PrintFile%, " \\\"
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
PRINT #PrintFile%, "\hline"
PRINT #PrintFile%, "\end{tabular} \hspace*{\fill}"
END IF
NEXT c&

FOR c& = 1 TO GIL&
Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
IF (c& - 1) MOD Lines% = 0 THEN
PRINT #PrintFile%, "\newpage"
Text$ = "Content (4/6) of " + RTRIM$(LEFT$(InputFile$, 8))
Text$ = Text$ + ".HQL ("
Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
GOSUB LaTeXOutput

```

```

PRINT #PrintFile%, "\\\"
PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
Text$ = "\begin{tabular}{|l|}"
Text$ = Text$ + "c@{\hspace{0.5em}}r@{}l@{\hspace{0.5em}}r"
Text$ = Text$ + "@{}l@{}r@{}l|"
Text$ = Text$ + "c@{\hspace{0.5em}}l|} \hline"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{1}{|c|}{\ensuremath{1^{\mathrm{st}}}}"
Text$ = Text$ + " Source} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{7}{|c|}{\ensuremath{1^{\mathrm{st}}}}"
Text$ = Text$ + " Relation Date} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{2}{|c|}{\ensuremath{1^{\mathrm{st}}}}"
Text$ = Text$ + " Relation Name} \\"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\hline"
ELSEIF (c& - 1) MOD 5 = 0 THEN
  PRINT #PrintFile%, "\hline"
END IF
Text$ = Part$(Data$, "data.source")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].tol1")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].date.year")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.ds")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.r[1].moment1"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].name")
GOSUB LaTeXOutput
PRINT #PrintFile%, " \\"
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
  PRINT #PrintFile%, "\hline"
  PRINT #PrintFile%, "\end{tabular} \hspace*{\fill}"
END IF
NEXT c&

FOR c& = 1 TO GIL&
  Data$ = Load$(InputFile$, DataLength%, GIL&, c&)
  IF (c& - 1) MOD Lines% = 0 THEN
    PRINT #PrintFile%, "\newpage"
    Text$ = "Content (5/6) of " + RTRIM$(LEFT$(InputFile$, 8))
    Text$ = Text$ + ".HQL ("
    Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
    Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
    Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
    Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
    GOSUB LaTeXOutput
  
```

```

PRINT #PrintFile%, "\\"
PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
Text$ = "\begin{tabular}{|c@{\hspace{0.5em}}c@{\hspace{0.5em}}r@{}l}"
Text$ = Text$ + "@{\hspace{0.5em}}r@{}l@{\hspace{0.5em}}r@{}l|l|"
Text$ = Text$ + "c@{\hspace{0.5em}}r@{}l@{\hspace{0.5em}}r@{}l"
Text$ = Text$ + "@{\hspace{0.5em}}r@{}l|)"
Text$ = Text$ + " \hline"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{8}{|c|}{\ensuremath{1^{\mathrm{st}}}"
Text$ = Text$ + " Duration} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{1}{|c|}{\ensuremath{2^{\mathrm{nd}}}"
Text$ = Text$ + " Source} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{7}{|c|}{\ensuremath{2^{\mathrm{nd}}}"
Text$ = Text$ + " Relation Date} \\"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\hline"
ELSEIF (c& - 1) MOD 5 = 0 THEN
  PRINT #PrintFile%, "\hline"
END IF
Text$ = ToUser$(Part$(Data$, "data.r[1].moment2"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].tol2")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].duration.year")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.ds")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].source")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].tol1")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[2].date.year")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.ds")
GOSUB LaTeXOutput
PRINT #PrintFile%, "\\"
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
  PRINT #PrintFile%, "\hline"
  PRINT #PrintFile%, "\end{tabular}"
END IF

```

NEXT c&

FOR c& = 1 TO GIL&

Data\$ = Load\$(InputFile%, DataLength%, GIL&, c&)

IF (c& - 1) MOD Lines% = 0 THEN

PRINT #PrintFile%, "\newpage"

Text\$ = "Content (6/6) of " + RTRIM\$(LEFT\$(InputFile\$, 8))

Text\$ = Text\$ + ".HQL ("

Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1

Text\$ = Text\$ + LTRIM\$(STR\$(Number&)) + "/"

Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1

Text\$ = Text\$ + LTRIM\$(STR\$(Number&)) + "):"

GOSUB LaTeXOutput

PRINT #PrintFile%, "\\"

PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"

Text\$ = "\begin{tabular}{|c@{\hspace{0.5em}}l|}"

Text\$ = Text\$ + "c@{\hspace{0.5em}}c@{\hspace{0.5em}}r@{1}"

Text\$ = Text\$ + "@{\hspace{0.5em}}"

Text\$ = Text\$ + "r@{1}l@{\hspace{0.5em}}r@{1}l|"

Text\$ = Text\$ + "l@{\hspace{0.5em}}l|l|}"

Text\$ = Text\$ + " \hline"

PRINT #PrintFile%, Text\$

Text\$ = "\multicolumn{2}{|c|}{\ensuremath{2^{\mathrm{nd}}}}"

Text\$ = Text\$ + " Relation Name} &"

PRINT #PrintFile%, Text\$

Text\$ = "\multicolumn{8}{|c|}{\ensuremath{2^{\mathrm{nd}}}}"

Text\$ = Text\$ + " Duration} &"

PRINT #PrintFile%, Text\$

PRINT #PrintFile%, "\multicolumn{2}{|c|}{Same Time} &"

Text\$ = "\multicolumn{1}{|c|}{\ensuremath{3^{\mathrm{rd}}}}"

Text\$ = Text\$ + " Source} \\"

PRINT #PrintFile%, Text\$

PRINT #PrintFile%, "\hline"

ELSEIF (c& - 1) MOD 5 = 0 THEN

PRINT #PrintFile%, "\hline"

END IF

Text\$ = ToUser\$(Part\$(Data\$, "data.r[2].moment1"))

GOSUB LaTeXTableSpecialCharacter

Text\$ = Part\$(Data\$, "data.r[2].name")

GOSUB LaTeXTable

Text\$ = ToUser\$(Part\$(Data\$, "data.r[2].moment2"))

GOSUB LaTeXTableSpecialCharacter

Text\$ = Part\$(Data\$, "data.r[2].tol2")

GOSUB LaTeXTableSpecialCharacter

Text\$ = Part\$(Data\$, "data.r[2].duration.year")

GOSUB LaTeXTable

Text\$ = Part\$(Data\$, "data.r[2].duration.ys")

GOSUB LaTeXTable

Text\$ = Part\$(Data\$, "data.r[2].duration.month")

GOSUB LaTeXTable

Text\$ = Part\$(Data\$, "data.r[2].duration.ms")

GOSUB LaTeXTable

Text\$ = Part\$(Data\$, "data.r[2].duration.day")

GOSUB LaTeXTable

Text\$ = Part\$(Data\$, "data.r[2].duration.ds")

GOSUB LaTeXTable

Simultaneous\$ = ToUser\$(Part\$(Data\$, "data.simultaneous"))

Text\$ = LEFT\$(Simultaneous\$, 3)

```

GOSUB LaTeXTable
Text$ = RIGHT$(Simultaneous$, 2)
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].source")
GOSUB LaTeXOutput
PRINT #PrintFile%, " \\"
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
    PRINT #PrintFile%, "\hline"
    PRINT #PrintFile%, "\end{tabular}"
END IF
NEXT c&

PRINT #PrintFile%, "\end{document}"
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Text$ = "Please terminate the program for compiling."
Present 2, Text$, "", 0, "C", 1, 1, 24
Pause
RETURN 'LaTeX11pt _____'

'=====
GenerateLaTeXFile:
'=====
FOR c& = 1 TO GIL&
    Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
    IF (c& - 1) MOD Lines% = 0 THEN
        IF c& > 1 THEN
            PRINT #PrintFile%, "\newpage"
        END IF
        Text$ = "Datings (1/3) of " + RTRIM$(LEFT$(InputFile$, 8))
        Text$ = Text$ + ".HQL ("
        Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
        Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
        GOSUB LaTeXOutput
        PRINT #PrintFile%, "\\"
        IF Lines% = 100 THEN
            PRINT #PrintFile%, "\begin{tiny}"
        ELSE
            PRINT #PrintFile%, "\begin{scriptsize}"
        END IF
        PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
        Text$ = "\begin{tabular}{*{6}{|c|}{r@{}l@{}r@{}l@{}r@{}l@{}c@{}}"
        Text$ = Text$ + "c@{}r@{}l@{}r@{}l@{}r@{}l@{} \hline"
        PRINT #PrintFile%, Text$
        PRINT #PrintFile%, "\multicolumn{15}{|c|}{Begin of Life [$*$$]} &"
        Text$ = "\multicolumn{15}{|c|}{Begin \ensuremath{1^{\mathrm{st}}}"
        Text$ = Text$ + " Periode [B] } &"
        PRINT #PrintFile%, Text$
        Text$ = "\multicolumn{15}{|c|}{Begin \ensuremath{2^{\mathrm{nd}}}"
        Text$ = Text$ + " Periode ["
        PRINT #PrintFile%, Text$;
        Text$ = "\alpha"
        GOSUB LaTeXOutput
        PRINT #PrintFile%, "]" } &"
    
```

```

Text$ = "\multicolumn{15}{|c|}{End of \ensuremath{2^{\mathrm{nd}}}}}"
Text$ = Text$ + " Periode ["
PRINT #PrintFile%, Text$;
Text$ = "Ω"
GOSUB LaTeXOutput
PRINT #PrintFile%, "]" } &"
Text$ = "\multicolumn{15}{|c|}{End of \ensuremath{1^{\mathrm{st}}}}}"
Text$ = Text$ + " Periode [E] } &"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\multicolumn{15}{|c|}{End of Life [+] }\\\"
PRINT #PrintFile%, "\hline"
ELSEIF (c& - 1) MOD (Lines% / 10) = 0 THEN
  PRINT #PrintFile%, "\hline"
END IF
FOR x% = 1 TO 6
  Variable$ = "data.date[" + LTRIM$(STR$(x%)) + "]"
  DateValue$ = Part$(Data$, Variable$)
  Text$ = Part$(DateValue$, "date.minimum.sign")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.minimum.year")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.minimum.ys")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.minimum.month")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.minimum.ms")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.minimum.day")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.minimum.ds")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.status")
  GOSUB LaTeXTableSpecialCharacter
  Text$ = Part$(DateValue$, "date.maximum.sign")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.maximum.year")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.maximum.ys")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.maximum.month")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.maximum.ms")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.maximum.day")
  GOSUB LaTeXTable
  Text$ = Part$(DateValue$, "date.maximum.ds")
  GOSUB LaTeXOutput
  IF x% < 6 THEN
    PRINT #PrintFile%, " & "
  ELSE
    PRINT #PrintFile%, " \\"
  END IF
NEXT x%
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
  PRINT #PrintFile%, "\hline"
  PRINT #PrintFile%, "\end{tabular} \hspace*{\fill}"
  IF Lines% = 100 THEN
    PRINT #PrintFile%, "\end{tiny}"
  
```

```

ELSE
    PRINT #PrintFile%, "\end{scriptsize}"
END IF
END IF
NEXT c&

FOR c& = 1 TO GIL&
    Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
    IF (c& - 1) MOD Lines% = 0 THEN
        PRINT #PrintFile%, "\newpage"
        Text$ = "Content (2/3) of " + RTRIM$(LEFT$(InputFile$, 8))
        Text$ = Text$ + ".HQL ("
        Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
        Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
        Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
        GOSUB LaTeXOutput
        PRINT #PrintFile%, "\\\"
        IF Lines% = 100 THEN
            PRINT #PrintFile%, "\begin{tiny}"
        ELSE
            PRINT #PrintFile%, "\begin{scriptsize}"
        END IF
        PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
        Text$ = "\begin{tabular}{|l|}"
        Text$ = Text$ + "c@{\hspace{0.5em}}c@{\hspace{0.5em}}l"
        Text$ = Text$ + "@{\hspace{0.5em}}c|"
        Text$ = Text$ + "c@{\hspace{0.5em}}c@{\hspace{0.5em}}l"
        Text$ = Text$ + "@{\hspace{0.5em}}c|"
        Text$ = Text$ + "l|c@{\hspace{0.5em}}r@{}l@{\hspace{0.5em}}|"
        Text$ = Text$ + "r@{}l@{} }r@{}l|c@{\hspace{0.5em}}l|} \hline"
        PRINT #PrintFile%, Text$
        PRINT #PrintFile%, "\multicolumn{1}{|c}{Name /Event} &"
        Text$ = "\multicolumn{4}{|c}{\ensuremath{1^{\mathrm{st}}}}"
        Text$ = Text$ + " Predecessor} &"
        PRINT #PrintFile%, Text$
        Text$ = "\multicolumn{4}{|c}{\ensuremath{2^{\mathrm{nd}}}}"
        Text$ = Text$ + " Predecessor} &"
        PRINT #PrintFile%, Text$
        Text$ = "\multicolumn{1}{|c}{\ensuremath{1^{\mathrm{st}}}}"
        Text$ = Text$ + " Source} &"
        PRINT #PrintFile%, Text$
        Text$ = "\multicolumn{7}{|c}{\ensuremath{1^{\mathrm{st}}}}"
        Text$ = Text$ + " Relation Date} &"
        PRINT #PrintFile%, Text$
        Text$ = "\multicolumn{2}{|c|}{\ensuremath{1^{\mathrm{st}}}}"
        Text$ = Text$ + " Relation Name} \\"
        PRINT #PrintFile%, Text$
        PRINT #PrintFile%, "\hline"
    ELSEIF (c& - 1) MOD (Lines% / 10) = 0 THEN
        PRINT #PrintFile%, "\hline"
    END IF
    Text$ = Part$(Data$, "data.name")
    GOSUB LaTeXTable
    Text$ = ToUser$(Part$(Data$, "data.p[1].moment1"))
    GOSUB LaTeXTableSpecialCharacter
    Text$ = ToUser$(Part$(Data$, "data.p[1].direct"))
    GOSUB LaTeXTable

```

```

Text$ = Part$(Data$, "data.p[1].name")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.p[1].moment2"))
GOSUB LaTeXTableSpecialCharacter
Text$ = ToUser$(Part$(Data$, "data.p[2].moment1"))
GOSUB LaTeXTableSpecialCharacter
Text$ = ToUser$(Part$(Data$, "data.p[2].direct"))
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.p[2].name")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.p[2].moment2"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.source")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].tol1")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].date.year")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].date.ds")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.r[1].moment1"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].name")
GOSUB LaTeXOutput
PRINT #PrintFile%, " \\"
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
  PRINT #PrintFile%, "\hline"
  PRINT #PrintFile%, "\end{tabular} \hspace*{\fill}"
  IF Lines% = 100 THEN
    PRINT #PrintFile%, "\end{tiny}"
  ELSE
    PRINT #PrintFile%, "\end{scriptsize}"
  END IF
END IF
NEXT c&

FOR c& = 1 TO GIL&
  Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
  IF (c& - 1) MOD Lines% = 0 THEN
    PRINT #PrintFile%, "\newpage"
    Text$ = "Content (3/3) of " + RTRIM$(LEFT$(InputFile$, 8))
    Text$ = Text$ + ".HQL ("
    Number& = ((c& - 1) - ((c& - 1) MOD Lines%)) / Lines% + 1
    Text$ = Text$ + LTRIM$(STR$(Number&)) + "/"
    Number& = ((GIL& - 1) - ((GIL& - 1) MOD Lines%)) / Lines% + 1
    Text$ = Text$ + LTRIM$(STR$(Number&)) + "):"
    GOSUB LaTeXOutput
    PRINT #PrintFile%, "\\"
    IF Lines% = 100 THEN
      PRINT #PrintFile%, "\begin{tiny}"
    END IF
  END IF
NEXT c&

```

```

ELSE
    PRINT #PrintFile%, "\begin{scriptsize}"
END IF
PRINT #PrintFile%, "\setlength{\rightmargin}{\leftmargin}"
Text$ = "\begin{tabular}{|c@{\hspace{0.5em}}c@{\hspace{0.5em}}r@{}l}"
Text$ = Text$ + "@{\hspace{0.5em}}r@{}l@{\hspace{0.5em}}r@{}l|l|"
Text$ = Text$ + "c@{\hspace{0.5em}}r@{}l@{\hspace{0.5em}}r@{}l|"
Text$ = Text$ + "@{\hspace{0.5em}}r@{}l|c@{\hspace{0.5em}}l|"
Text$ = Text$ + "c@{\hspace{0.5em}}c@{\hspace{0.5em}}r@{}l|"
Text$ = Text$ + "@{\hspace{0.5em}}|"
Text$ = Text$ + "r@{}l@{\hspace{1em}}r@{}l|l@{\hspace{1em}}l|l|l|"
Text$ = Text$ + " \hline"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{8}{|c|}{\ensuremath{1^{\mathrm{st}}}}"
Text$ = Text$ + " Duration} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{1}{|c|}{\ensuremath{2^{\mathrm{nd}}}}"
Text$ = Text$ + " Source} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{7}{|c|}{\ensuremath{2^{\mathrm{nd}}}}"
Text$ = Text$ + " Relation Date} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{2}{|c|}{\ensuremath{2^{\mathrm{nd}}}}"
Text$ = Text$ + " Relation Name} &"
PRINT #PrintFile%, Text$
Text$ = "\multicolumn{8}{|c|}{\ensuremath{2^{\mathrm{nd}}}}"
Text$ = Text$ + " Duration} &"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\multicolumn{2}{|c|}{Same Time} &"
Text$ = "\multicolumn{1}{|c|}{\ensuremath{3^{\mathrm{rd}}}}"
Text$ = Text$ + " Source} \\"
PRINT #PrintFile%, Text$
PRINT #PrintFile%, "\hline"
ELSEIF (c& - 1) MOD (Lines% / 10) = 0 THEN
    PRINT #PrintFile%, "\hline"
END IF
Text$ = ToUser$(Part$(Data$, "data.r[1].moment2"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].tol2")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[1].duration.year")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].duration.ds")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[1].source")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].tol1")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[2].date.year")
GOSUB LaTeXTable

```

```

Text$ = Part$(Data$, "data.r[2].date.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].date.ds")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.r[2].moment1"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[2].name")
GOSUB LaTeXTable
Text$ = ToUser$(Part$(Data$, "data.r[2].moment2"))
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[2].tol2")
GOSUB LaTeXTableSpecialCharacter
Text$ = Part$(Data$, "data.r[2].duration.year")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].duration.ys")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].duration.month")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].duration.ms")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].duration.day")
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].duration.ds")
GOSUB LaTeXTable
Simultaneous$ = ToUser$(Part$(Data$, "data.simultaneous"))
Text$ = LEFT$(Simultaneous$, 3)
GOSUB LaTeXTable
Text$ = RIGHT$(Simultaneous$, 2)
GOSUB LaTeXTable
Text$ = Part$(Data$, "data.r[2].source")
GOSUB LaTeXOutput
PRINT #PrintFile%, " \\"
IF (c& - 1) MOD Lines% = Lines% - 1 OR c& = GIL& THEN
    PRINT #PrintFile%, "\hline"
    PRINT #PrintFile%, "\end{tabular}"
    IF Lines% = 100 THEN
        PRINT #PrintFile%, "\end{tiny}"
    ELSE
        PRINT #PrintFile%, "\end{scriptsize}"
    END IF
END IF
NEXT c&

'Checks of Singular Letters:
'-----'
'PRINT #PrintFile%, "\begin{tiny}"
'FOR x% = 0 TO 255
'    Text$ = CHR$(x%)
'    GOSUB LaTeXOutput
'    IF x% MOD 16 = 15 THEN
'        PRINT #PrintFile%, " \\"
'    END IF

```

```
'NEXT x%
'PRINT #PrintFile%, "\end{tiny}"
RETURN 'GenerateLaTeXFile _____'
```

```
'=====
LaTeXOutput:
'=====
```

```
CopyData$ = ""
```

```
FOR y% = 1 TO LEN(Text$)
```

```
  ASCII% = ASC(MID$(Text$, y%, 1))
```

```
  SELECT CASE ASCII%
```

```
  CASE 0 TO 2, 7 TO 15, 28
```

```
    A$ = "\" + LTRIM$(STR$(ASCII%)) + "\"
```

```
  CASE 3
```

```
    A$ = "\ensuremath{\mathrm{\heartsuit}}"
```

```
  CASE 4
```

```
    A$ = "\ensuremath{\mathrm{\diamondsuit}}"
```

```
  CASE 5
```

```
    A$ = "\ensuremath{\mathrm{\clubsuit}}"
```

```
  CASE 6
```

```
    A$ = "\ensuremath{\mathrm{\spadesuit}}"
```

```
  CASE 16
```

```
    A$ = "\ensuremath{\mathrm{\rhd}}"
```

```
  CASE 17
```

```
    A$ = "\ensuremath{\mathrm{\lhd}}"
```

```
  CASE 18
```

```
    A$ = "\ensuremath{\mathrm{\updownarrow}}"
```

```
  CASE 19
```

```
    A$ = "! \rule{-0.15em}{0ex} !" "
```

```
  CASE 20
```

```
    A$ = "\P"
```

```
  CASE 21
```

```
    A$ = "\S"
```

```
  CASE 22
```

```
    A$ = "\rule{0.25em}{0ex}\rule{0.5em}{1ex}\rule{0.25em}{0ex}"
```

```
  CASE 23
```

```
    A$ = "\underline{\ensuremath{\mathrm{\updownarrow}}}"
```

```
  CASE 24
```

```
    A$ = "\ensuremath{\mathrm{\uparrow}}"
```

```
  CASE 25
```

```
    A$ = "\ensuremath{\mathrm{\downarrow}}"
```

```
  CASE 26
```

```
    A$ = "\ensuremath{\mathrm{\rightarrow}}"
```

```
  CASE 27
```

```
    A$ = "\ensuremath{\mathrm{\leftarrow}}"
```

```
  CASE 29
```

```
    A$ = "\ensuremath{\mathrm{\leftrightarrow}}"
```

```
  CASE 30
```

```
    A$ = "\ensuremath{\mathrm{\bigtriangleup}}"
```

```
  CASE 31
```

```
    A$ = "\ensuremath{\mathrm{\bigtriangledown}}"
```

```
  CASE 32
```

```
    A$ = "\rule{0.5em}{0ex}" ' "\hspace{0.5em}" would be less precise!
```

```
  CASE 34, 42, 94, 126
```

```
    A$ = "\symbol{" + LTRIM$(STR$(ASCII%)) + "}"
```

```
  CASE 35 TO 38, 95, 123, 125
```

```
    A$ = "\" + CHR$(ASCII%)
```

```
  CASE 60, 62
```

```

    A$ = "\ensuremath{\mathrm{" + CHR$(ASCII%) + "}}"
CASE 92
    A$ = "\ensuremath{\mathrm{\backslash}}"
CASE 124
    A$ = "\ensuremath{\mathrm{\mid}}"
CASE 127
    A$ = "\ensuremath{\mathrm{\Delta}}"
CASE 155, 157, 169 TO 172, 176 TO 218, 244 TO 245
    A$ = "\{" + LTRIM$(STR$(ASCII%)) + "\"
CASE 168
    A$ = "?`"
CASE 173
    A$ = "!\`"
CASE 219
    A$ = "\rule[-0.9ex]{1em}{2.9ex}"
CASE 220
    A$ = "\rule[-0.9ex]{1em}{1.45ex}"
CASE 221
    A$ = "\rule[-0.9ex]{0.5em}{2.9ex}\rule{0.5em}{0ex}"
CASE 222
    A$ = "\rule{0.5em}{0ex}\rule[-0.9ex]{0.5em}{2.9ex}"
CASE 223
    A$ = "\rule[0.55ex]{1em}{1.45ex}"
CASE 226
    A$ = "\symbol{0}"
CASE 224, 226 TO 243, 246 TO 247, 251 TO 253
    A$ = "\ensuremath{\mathrm{" + CHR$(ASCII%) + "}}"
CASE 249
    A$ = "\ensuremath{\mathrm{\bullet}}"
CASE 250
    A$ = "\ensuremath{\mathrm{\cdot}}"
CASE 255
    A$ = "\rule{1em}{0ex}"      '"\hspace{1em}" would be less precise!
CASE ELSE
    A$ = CHR$(ASCII%)          'as much clear text as possible
END SELECT
CopyData$ = CopyData$ + A$
NEXT y%
PRINT #PrintFile%, CopyData$;
RETURN 'LaTeXOutput _____'

'=====
LaTeXOutputSpecialCharacter:
'=====
    IF Text$ = SPACE$(1) THEN
        Text$ = CHR$(255)      'necessary because of \usepackage{latexsym}
    END IF
    GOSUB LaTeXOutput
RETURN 'LaTeXOutputSpecialCharacter _____'

'=====
LaTeXTable:
'=====
    GOSUB LaTeXOutput
    PRINT #PrintFile%, " & ";
RETURN 'LaTeXTable _____'

'=====

```

```

LaTeXTableSpecialCharacter:
'=====
  GOSUB LaTeXOutputSpecialCharacter
  PRINT #PrintFile%, " & ";
RETURN 'LaTeXTableSpecialCharacter _____'

'==
CSV:
'==
  Text$ = "Output file " + LEFT$(InputFile$, 8) + ".CSV is generated."
  Present 2, Text$, "", 0, "C", 1, 1, 24
  PrintFile% = FREEFILE
  Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".CSV"
  OPEN Buffer$ FOR OUTPUT AS #PrintFile%

  'Headlines:
  '-----'
  PRINT #PrintFile%, ", ";
  Text$ = "Begin of Life [ * ]"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "Begin 1st Periode [ B ]"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "Begin 2nd Periode [ α ]"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "End 2nd Periode [ Ω ]"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "End 1st Periode [ E ]"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "End of Life [ + ]"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", ";
  Text$ = "Name /Event"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "1st Predecessor"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , , ";
  Text$ = "2nd Predecessor"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", ";
  Text$ = "1st Source"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", ";
  Text$ = "1st Relation Date"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", ";
  Text$ = "1st Relation Name"
  GOSUB CSVOutput
  PRINT #PrintFile%, ", , ";
  Text$ = "1st Duration"
  GOSUB CSVOutput
  Text$ = "2nd Source"
  GOSUB CSVOutput

```

```

PRINT #PrintFile%, ", ";
Text$ = "2nd Relation Date"
GOSUB CSVOutput
PRINT #PrintFile%, ", ";
Text$ = "2nd Relation Name"
GOSUB CSVOutput
PRINT #PrintFile%, ", , ";
Text$ = "2nd Duration"
GOSUB CSVOutput
Text$ = "Same Time"
GOSUB CSVOutput
Text$ = "3rd Source"
GOSUB CSVOutput
PRINT #PrintFile%, ""

FOR c& = 1 TO GIL&
  Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
  FOR x% = 1 TO 6
    Variable$ = "data.date[" + LTRIM$(STR$(x%)) + "]"
    Text$ = Part$(Data$, Variable$ + ".minimum")
    GOSUB CSVOutput
    Text$ = Part$(Data$, Variable$ + ".status")
    SELECT CASE Text$
      CASE "X"
        Text$ = "X"
      CASE ">"
        Text$ = ">"
      CASE "<"
        Text$ = "<"
      CASE "≥", "≤"
        Text$ = " "
    END SELECT
    GOSUB CSVOutput
    Text$ = Part$(Data$, Variable$ + ".maximum")
    GOSUB CSVOutput
  NEXT x%
  Text$ = Part$(Data$, "data.name")
  GOSUB CSVOutput
  FOR x% = 1 TO 2
    Variable$ = "data.p[" + LTRIM$(STR$(x%)) + "]"
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment1"))
    GOSUB CSVOutput
    Text$ = ToUser$(Part$(Data$, Variable$ + ".direct"))
    GOSUB CSVOutput
    Text$ = Part$(Data$, Variable$ + ".name")
    GOSUB CSVOutput
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment2"))
    GOSUB CSVOutput
  NEXT x%
  Text$ = Part$(Data$, "data.source")
  GOSUB CSVOutput
  FOR x% = 1 TO 2
    Variable$ = "data.r[" + LTRIM$(STR$(x%)) + "]"
    Text$ = Part$(Data$, Variable$ + ".tol1")
    GOSUB CSVOutput
    Text$ = Part$(Data$, Variable$ + ".date")
    GOSUB CSVOutput
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment1"))

```

```

        GOSUB CSVOutput
        Text$ = Part$(Data$, Variable$ + ".name")
        GOSUB CSVOutput
        Text$ = ToUser$(Part$(Data$, Variable$ + ".moment2"))
        GOSUB CSVOutput
        Text$ = Part$(Data$, Variable$ + ".tol2")
        GOSUB CSVOutput
        Text$ = Part$(Data$, Variable$ + ".duration")
        GOSUB CSVOutput
        IF x% = 2 THEN
            Text$ = ToUser$(Part$(Data$, "data.simultaneous"))
            Text$ = LEFT$(Text$, 3) + " " + RIGHT$(Text$, 2)
            GOSUB CSVOutput
        END IF
        Text$ = Part$(Data$, Variable$ + ".source")
        GOSUB CSVOutput
    NEXT x%
    PRINT #PrintFile%, ""
NEXT c&
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Present 2, "Please terminate the program to view.", "", 0, "C", 1, 1, 24
Pause
RETURN 'CSV _____'

'=====
CSVOutput:
'=====
CopyData$ = ""
FOR y% = 1 TO LEN(Text$)
    ASCII% = ASC(MID$(Text$, y%, 1))
    SELECT CASE ASCII%
    CASE 20
        Out$ = CHR$(182)
    CASE 21
        Out$ = CHR$(167)
    CASE 127 TO 223, 225 TO 233, 235 TO 254
        IF HTML%(ASCII% - 126) < 256 THEN
            Out$ = CHR$(HTML%(ASCII% - 126))
        ELSE
            Out$ = "<" + LTRIM$(STR$(ASCII%)) + ">"
        END IF
    CASE 224
        Out$ = "A"
    CASE 234
        Out$ = "O"
    CASE ELSE
        Out$ = CHR$(ASCII%)
    END SELECT
    CopyData$ = CopyData$ + Out$
NEXT y%
PRINT #PrintFile%, CHR$(34) + CopyData$ + CHR$(34) + ",";
RETURN 'CSVOutput _____'

'===
HTML:

```

```

'===
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".HTM is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".HTM"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
  PRINT #PrintFile%, "<HTML>"
  PRINT #PrintFile%, "<HEAD>"
  Align$ = "Title"
  Text$ = InputFile$
  GOSUB HTMLOutput
  PRINT #PrintFile%, "</HEAD>"
  PRINT #PrintFile%, "<BODY><CENTER>"
  PRINT #PrintFile%, "<TABLE BORDER=5 CELSPACING=0 CELLPADDING=2><TR>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=7> Begin of Life [ * ] </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=7>Begin 1st Periode [ A ]</TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=7>Begin 2nd Periode [ W ]</TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=7> End 2nd Periode [ X ] </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=7> End 1st Periode [ E ] </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=7> End of Life [ + ] </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >Name /Event </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=3>1st Predecessor </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >Moment </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=3>2nd Predecessor </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >Moment </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >1st Source </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=4>1st Relation Date </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=2>1st Relation Name </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >Moment </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=4>1st Duration </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >2nd Source </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=4>2nd Relation Date </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=2>2nd Relation Name </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >Moment </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=4>2nd Duration </TH>"
  PRINT #PrintFile%, "<TH NOWRAP COLSPAN=2>Same Time </TH>"
  PRINT #PrintFile%, "<TH NOWRAP >3rd Source </TH>"
  PRINT #PrintFile%, "</TR>"
  FOR c% = 1 TO GIL&
    Data$ = Load$(InputFile%, DataLength%, GIL&, c&)
    PRINT #PrintFile%, "<TR>"
    FOR x% = 1 TO 6
      Align$ = "right"
      Variable$ = "data.date[" + LTRIM$(STR$(x%)) + "]"
      Text$ = Part$(Data$, Variable$ + ".minimum.day")
      Text$ = Text$ + Part$(Data$, Variable$ + ".minimum.ds")
      GOSUB HTMLOutput
      Text$ = Part$(Data$, Variable$ + ".minimum.month")
      Text$ = Text$ + Part$(Data$, Variable$ + ".minimum.ms")
      GOSUB HTMLOutput
      Text$ = Part$(Data$, Variable$ + ".minimum.sign")
      Text$ = Text$ + Part$(Data$, Variable$ + ".minimum.year")
      Text$ = Text$ + Part$(Data$, Variable$ + ".minimum.ys")
      GOSUB HTMLOutput
      Align$ = "center"
      Text$ = Part$(Data$, Variable$ + ".status")
      GOSUB HTMLOutput
      Align$ = "right"
    
```

```
Text$ = Part$(Data$, Variable$ + ".maximum.day")
Text$ = Text$ + Part$(Data$, Variable$ + ".maximum.ds")
GOSUB HTMLOutput
Text$ = Part$(Data$, Variable$ + ".maximum.month")
Text$ = Text$ + Part$(Data$, Variable$ + ".maximum.ms")
GOSUB HTMLOutput
Text$ = Part$(Data$, Variable$ + ".maximum.sign")
Text$ = Text$ + Part$(Data$, Variable$ + ".maximum.year")
Text$ = Text$ + Part$(Data$, Variable$ + ".maximum.ys")
GOSUB HTMLOutput
NEXT x%
Align$ = "left"
Text$ = Part$(Data$, "data.name")
GOSUB HTMLOutput
FOR x% = 1 TO 2
  Variable$ = "data.p[" + LTRIM$(STR$(x%)) + "]"
  Align$ = "center"
  Text$ = ToUser$(Part$(Data$, Variable$ + ".moment1"))
  GOSUB HTMLOutput
  Text$ = ToUser$(Part$(Data$, Variable$ + ".direct"))
  GOSUB HTMLOutput
  Align$ = "left"
  Text$ = Part$(Data$, Variable$ + ".name")
  GOSUB HTMLOutput
  Align$ = "center"
  Text$ = ToUser$(Part$(Data$, Variable$ + ".moment2"))
  GOSUB HTMLOutput
NEXT x%
Align$ = "left"
Text$ = Part$(Data$, "data.source")
GOSUB HTMLOutput
FOR x% = 1 TO 2
  Variable$ = "data.r[" + LTRIM$(STR$(x%)) + "]"
  Align$ = "center"
  Text$ = Part$(Data$, Variable$ + ".tol1")
  GOSUB HTMLOutput
  Align$ = "right"
  Text$ = Part$(Data$, Variable$ + ".date.year")
  Text$ = Text$ + Part$(Data$, Variable$ + ".date.ys")
  GOSUB HTMLOutput
  Text$ = Part$(Data$, Variable$ + ".date.month")
  Text$ = Text$ + Part$(Data$, Variable$ + ".date.ms")
  GOSUB HTMLOutput
  Text$ = Part$(Data$, Variable$ + ".date.day")
  Text$ = Text$ + Part$(Data$, Variable$ + ".date.ds")
  GOSUB HTMLOutput
  Align$ = "center"
  Text$ = ToUser$(Part$(Data$, Variable$ + ".moment1"))
  GOSUB HTMLOutput
  Align$ = "left"
  Text$ = Part$(Data$, Variable$ + ".name")
  GOSUB HTMLOutput
  Align$ = "center"
  Text$ = ToUser$(Part$(Data$, Variable$ + ".moment2"))
  GOSUB HTMLOutput
  Align$ = "right"
  Text$ = Part$(Data$, Variable$ + ".tol2")
  GOSUB HTMLOutput
```

```

Text$ = Part$(Data$, Variable$ + ".duration.year")
Text$ = Text$ + Part$(Data$, Variable$ + ".duration.ys")
GOSUB HTMLOutput
Text$ = Part$(Data$, Variable$ + ".duration.month")
Text$ = Text$ + Part$(Data$, Variable$ + ".duration.ms")
GOSUB HTMLOutput
Text$ = Part$(Data$, Variable$ + ".duration.day")
Text$ = Text$ + Part$(Data$, Variable$ + ".duration.ds")
GOSUB HTMLOutput
Align$ = "left"
IF x% = 2 THEN
    Text$ = LEFT$(ToUser$(Part$(Data$, "data.simultaneous")), 3)
    CopyData$ = LEFT$(Text$, 1) + " " + MID$(Text$, 2, 1)
    CopyData$ = CopyData$ + " " + RIGHT$(Text$, 1)
    Text$ = CopyData$
    GOSUB HTMLOutput
    Text$ = RIGHT$(ToUser$(Part$(Data$, "data.simultaneous")), 2)
    CopyData$ = LEFT$(Text$, 1) + " " + RIGHT$(Text$, 1)
    GOSUB HTMLOutput
END IF
Text$ = Part$(Data$, Variable$ + ".source")
GOSUB HTMLOutput
NEXT x%

PRINT #PrintFile%, "</TR>"
NEXT c&
PRINT #PrintFile%, "</TABLE></CENTER>"

' Check of HTML Letters:
' -----
' PRINT #PrintFile%, "<TABLE BORDER=2><TR>"
' FOR x% = 0 TO 255
'   IF x% > 0 AND x% MOD 16 = 0 THEN
'     PRINT #PrintFile%, "</TR><TR>"
'   END IF
'   Text$ = STR$(x%) + ":" + CHR$(x%)
'   GOSUB HTMLOutput
' NEXT x%
' PRINT #PrintFile%, "</TR></TABLE>"
' FOR l& = 0 TO 31'65536
'   PRINT #PrintFile%, STR$(l&); "&#"; LTRIM$(STR$(l&)); ";"
' NEXT l&

PRINT #PrintFile%, "</BODY></HTML>"
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Present 2, "Please terminate the program to view.", "", 0, "C", 1, 1, 24
Pause
RETURN 'HTML _____'

' =====
HTMLOutput:
' =====
CopyData$ = ""
FOR y% = 1 TO LEN(Text$)

```

```
ASCII% = ASC(MID$(Text$, y%, 1))
SELECT CASE ASCII%
CASE 1, 2
    Out$ = "&#" + LTRIM$(STR$(9785 + ASCII%)) + ";"
CASE 3, 4
    Out$ = "&#" + LTRIM$(STR$(9826 + ASCII%)) + ";"
CASE 5
    Out$ = "&#9827;"
CASE 6
    Out$ = "&#9824;"
CASE 7
    Out$ = "&#9679;"
CASE 8
    Out$ = "&#9688;"
CASE 9
    Out$ = "&#9675;"
CASE 10
    Out$ = "&#9689;"
CASE 11
    Out$ = "&#9794;"
CASE 12
    Out$ = "&#9792;"
CASE 13, 14
    Out$ = "&#" + LTRIM$(STR$(9821 + ASCII%)) + ";"
CASE 15
    Out$ = "&#9788;"
CASE 16
    Out$ = "&#9658;"
CASE 17
    Out$ = "&#9668;"
CASE 18
    Out$ = "&#8597;"
CASE 19
    Out$ = "&#8252;"
CASE 20
    Out$ = "&#182;"
CASE 21
    Out$ = "&#167;"
CASE 22
    Out$ = "&#9644;"
CASE 23
    Out$ = "&#8616;"
CASE 24
    Out$ = "&#8593;"
CASE 25
    Out$ = "&#8595;"
CASE 26
    Out$ = "&#8594;"
CASE 27
    Out$ = "&#8592;"
CASE 28
    Out$ = "&#8735;"
CASE 29
    Out$ = "&#8596;"
CASE 30
    Out$ = "&#9650;"
CASE 31
    Out$ = "&#9660;"
```

```

CASE 34, 35, 38, 60, 62      'HTML command letters
  Out$ = "&#" + LTRIM$(STR$(ASCII%)) + ";"
CASE 127 TO 254
  Out$ = "&#" + LTRIM$(STR$(HTML%(ASCII% - 126))) + ";"
CASE 0, 255
  Out$ = " "                'Convert to real space letter!
CASE ELSE
  Out$ = CHR$(ASCII%)        'As much clear text as possible
END SELECT
CopyData$ = CopyData$ + Out$
NEXT y%
IF UCASE$(Align$) = "TITLE" THEN
  PRINT #PrintFile%, "<TITLE>" + CopyData$ + "</TITLE>"
ELSE
  PRINT #PrintFile%, "<TD NOWRAP ALIGN=" + Align$ + ">" + CopyData$ + "</TD>"
END IF
RETURN 'HTMLOutput _____'

'=====
Mathematica:
'=====
Text$ = "Output file " + LEFT$(InputFile$, 8) + ".m is generated."
Present 2, Text$, "", 0, "C", 1, 1, 24
PrintFile% = FREEFILE
Buffer$ = WorkingPlace$ + LEFT$(InputFile$, 8) + ".m"
OPEN Buffer$ FOR OUTPUT AS #PrintFile%
  PRINT #PrintFile%, "{"
  PRINT #PrintFile%, "data["
  FOR x% = 1 TO 6
    PRINT #PrintFile%, "Date[" + LTRIM$(STR$(x%)) + "]"
    PRINT #PrintFile%, "T1[Sign[],Year[],YS[],Month[],MS[],Day[],DS[]],"
    PRINT #PrintFile%, "Status[],";
    PRINT #PrintFile%, "T2[Sign[],Year[],YS[],Month[],MS[],Day[],DS[]]"
    PRINT #PrintFile%, "], "
  NEXT x%
  PRINT #PrintFile%, "Name[],"
  FOR x% = 1 TO 2
    PRINT #PrintFile%, "Predecessor[" + LTRIM$(STR$(x%)) + "]"
    PRINT #PrintFile%, "Moment1[],Direct[],Name[],Moment2[]"
    PRINT #PrintFile%, "], "
  NEXT x%
  PRINT #PrintFile%, "Source[],"
  FOR x% = 1 TO 2
    PRINT #PrintFile%, "Relation[" + LTRIM$(STR$(x%)) + "]"
    PRINT #PrintFile%, "Tol1[],Date[Year[],YS[],Month[],MS[],Day[],DS[]],"
    PRINT #PrintFile%, "Moment1[],Name[],Moment2[],Tol2[],"
    PRINT #PrintFile%, "Duration[Year[],YS[],Month[],MS[],Day[],DS[]],"
    IF x% = 2 THEN
      PRINT #PrintFile%, "Simultaneous[Case1[],Case2[]],"
    END IF
    PRINT #PrintFile%, "Source[]"
    PRINT #PrintFile%, "];"
    IF x% = 1 THEN
      PRINT #PrintFile%, ", ";
    END IF
  NEXT x%
  PRINT #PrintFile%, "], "

```

```

FOR c% = 1 TO GIL%
  Data$ = Load$(InputFile%, DataLength%, GIL%, c%)
  PRINT #PrintFile%, "{";
  FOR x% = 1 TO 6
    Variable$ = "data.date[" + LTRIM$(STR$(x%)) + "]"
    PRINT #PrintFile%, "{";
    Text$ = Part$(Data$, Variable$ + ".minimum")
    GOSUB MathematicaDate
    PRINT #PrintFile%, ","
    Text$ = Part$(Data$, Variable$ + ".status")
    SELECT CASE Text$
      CASE "X"
        Text$ = "X"
      CASE ">"
        Text$ = ">"
      CASE "<"
        Text$ = "<"
      CASE ELSE
        Text$ = SPACE$(1)
    END SELECT
    GOSUB MathematicaList
    Text$ = Part$(Data$, Variable$ + ".maximum")
    GOSUB MathematicaDate
    PRINT #PrintFile%, "},"
  NEXT x%
  Text$ = Part$(Data$, "data.name")
  GOSUB MathematicaList
  FOR x% = 1 TO 2
    Variable$ = "data.p[" + LTRIM$(STR$(x%)) + "]"
    PRINT #PrintFile%, "{";
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment1"))
    GOSUB MathematicaList
    Text$ = ToUser$(Part$(Data$, Variable$ + ".direct"))
    GOSUB MathematicaList
    Text$ = Part$(Data$, Variable$ + ".name")
    GOSUB MathematicaList
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment2"))
    GOSUB MathematicaOutput
    PRINT #PrintFile%, "},"
  NEXT x%
  Text$ = Part$(Data$, "data.source")
  GOSUB MathematicaList
  FOR x% = 1 TO 2
    Variable$ = "data.r[" + LTRIM$(STR$(x%)) + "]"
    PRINT #PrintFile%, "{";
    Text$ = Part$(Data$, Variable$ + ".tol1")
    GOSUB MathematicaList
    Text$ = Part$(Data$, Variable$ + ".date")
    GOSUB MathematicaDuration
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment1"))
    GOSUB MathematicaList
    Text$ = Part$(Data$, Variable$ + ".name")
    GOSUB MathematicaList
    Text$ = ToUser$(Part$(Data$, Variable$ + ".moment2"))
    GOSUB MathematicaList
    Text$ = Part$(Data$, Variable$ + ".tol2")
    GOSUB MathematicaList
    Text$ = Part$(Data$, Variable$ + ".duration")
  
```

```

GOSUB MathematicaDuration
IF x% = 2 THEN
    PRINT #PrintFile%, "{";
    Text$ = LEFT$(ToUser$(Part$(Data$, "data.simultaneous")), 3)
    GOSUB MathematicaList
    Text$ = RIGHT$(ToUser$(Part$(Data$, "data.simultaneous")), 2)
    GOSUB MathematicaOutput
    PRINT #PrintFile%, "},"
END IF
Text$ = Part$(Data$, Variable$ + ".source")
GOSUB MathematicaOutput
IF x% = 1 THEN
    PRINT #PrintFile%, "},"
ELSE
    PRINT #PrintFile%, "}"
END IF
NEXT x%
IF c% = GIL% THEN
    PRINT #PrintFile%, "}"
ELSE
    PRINT #PrintFile%, "},"
END IF
NEXT c%

PRINT #PrintFile%, "}"
CLOSE #PrintFile%

LOCATE 24, 1
PRINT SPACE$(80);
Present 2, "Please terminate the program to view.", "", 0, "C", 1, 1, 24
Pause

```

```
RETURN 'Mathematica _____'
```

```
'====='
```

```
MathematicaDate:
```

```
'====='
```

```

PRINT #PrintFile%, "{" + LTRIM$(Part$(Text$, "term.sign"));
PRINT #PrintFile%, LTRIM$(Part$(Text$, "term.year")) + ",";
PRINT #PrintFile%, CHR$(34) + RTRIM$(LTRIM$(Part$(Text$, "term.ys")));
PRINT #PrintFile%, CHR$(34) + ",";
PRINT #PrintFile%, LTRIM$(Part$(Text$, "term.month")) + ",";
PRINT #PrintFile%, CHR$(34) + RTRIM$(LTRIM$(Part$(Text$, "term.ms")));
PRINT #PrintFile%, CHR$(34) + ",";
PRINT #PrintFile%, LTRIM$(Part$(Text$, "term.day")) + ",";
PRINT #PrintFile%, CHR$(34) + RTRIM$(LTRIM$(Part$(Text$, "term.ds")));
PRINT #PrintFile%, CHR$(34) + "}";

```

```
RETURN 'MathematicaDate _____'
```

```
'====='
```

```
MathematicaDuration:
```

```
'====='
```

```

PRINT #PrintFile%, "{" + LTRIM$(Part$(Text$, "time.year")) + ",";
PRINT #PrintFile%, CHR$(34) + RTRIM$(LTRIM$(Part$(Text$, "time.ys")));
PRINT #PrintFile%, CHR$(34) + ",";
PRINT #PrintFile%, LTRIM$(Part$(Text$, "time.month")) + ",";
PRINT #PrintFile%, CHR$(34) + RTRIM$(LTRIM$(Part$(Text$, "time.ms")));
PRINT #PrintFile%, CHR$(34) + ",";

```

```
PRINT #PrintFile%, LTRIM$(Part$(Text$, "time.day")) + ",";
PRINT #PrintFile%, CHR$(34) + RTRIM$(LTRIM$(Part$(Text$, "time.ds")));
PRINT #PrintFile%, CHR$(34) + "},";
RETURN 'MathematicaDuration _____'

'=====
MathematicaList:
'=====
  GOSUB MathematicaOutput
  PRINT #PrintFile%, ",";
RETURN 'MathematicaList _____'

'=====
MathematicaOutput:
'=====
  Text$ = RTRIM$(LTRIM$(Text$))
  CopyData$ = ""
  FOR y% = 1 TO LEN(Text$)
    ASCII% = ASC(MID$(Text$, y%, 1))
    SELECT CASE ASCII%
      CASE 1
        Out$ = "[HappySmiley]"
      CASE 3
        Out$ = "[HeartSuit]"
      CASE 4
        Out$ = "[DiamondSuit]"
      CASE 5
        Out$ = "[ClubSuit]"
      CASE 6
        Out$ = "[SpadeSuit]"
      CASE 7
        Out$ = "[FilledSmallCircle]"
      CASE 9
        Out$ = "[EmptyCircle]"
      CASE 15
        Out$ = "[LightBulb]"
      CASE 16
        Out$ = "[RightTriangle]"
      CASE 17
        Out$ = "[LeftTriangle]"
      CASE 18
        Out$ = "[UpDownArrow]"
      CASE 19
        Out$ = "[NegativeMediumSpace]!"
      CASE 20
        Out$ = "[Paragraph]"
      CASE 21
        Out$ = "[Section]"
      CASE 24
        Out$ = "[UpArrow]"
      CASE 25
        Out$ = "[DownArrow]"
      CASE 26
        Out$ = "[RightArrow]"
      CASE 27
        Out$ = "[LeftArrow]"
      CASE 29
        Out$ = "[LeftRightArrow]"
```

```
CASE 30
  Out$ = "\[FilledUpTriangle]"
CASE 31
  Out$ = "\[FilledDownTriangle]"
CASE 32, 255
  Out$ = SPACE$(1)
CASE 34
  Out$ = "\" + CHR$(34)
CASE 92
  Out$ = "\[Backslash]"
CASE 127
  Out$ = "\[CapitalDelta]"
CASE 128
  Out$ = "\[CapitalCCedilla]"
CASE 129
  Out$ = "\[UDoubleDot]"
CASE 130
  Out$ = "\[EAcute]"
CASE 131
  Out$ = "\[AHat]"
CASE 132
  Out$ = "\[ADoubleDot]"
CASE 133
  Out$ = "\[AGrave]"
CASE 134
  Out$ = "\[ARing]"
CASE 135
  Out$ = "\[CCedilla]"
CASE 136
  Out$ = "\[EHat]"
CASE 137
  Out$ = "\[EDoubleDot]"
CASE 138
  Out$ = "\[EGrave]"
CASE 139
  Out$ = "\[IDoubleDot]"
CASE 140
  Out$ = "\[IHat]"
CASE 141
  Out$ = "\[IGrave]"
CASE 142
  Out$ = "\[CapitalADoubleDot]"
CASE 143
  Out$ = "\[CapitalARing]"
CASE 144
  Out$ = "\[CapitalEAcute]"
CASE 145
  Out$ = "\[AE]"
CASE 146
  Out$ = "\[CapitalAE]"
CASE 147
  Out$ = "\[OHat]"
CASE 148
  Out$ = "\[ODoubleDot]"
CASE 149
  Out$ = "\[OGrave]"
CASE 150
  Out$ = "\[UHat]"
```

```
CASE 151
  Out$ = "\[UGrave] "
CASE 152
  Out$ = "\[YDoubleDot] "
CASE 153
  Out$ = "\[CapitalODoubleDot] "
CASE 154
  Out$ = "\[CapitalUDoubleDot] "
CASE 155
  Out$ = "\[Cent] "
CASE 156
  Out$ = "\[Sterling] "
CASE 157
  Out$ = "\[Yen] "
CASE 159
  Out$ = "\[Florin] "
CASE 160
  Out$ = "\[AAcute] "
CASE 161
  Out$ = "\[IAcute] "
CASE 162
  Out$ = "\[OAcute] "
CASE 163
  Out$ = "\[UAcute] "
CASE 164
  Out$ = "\[NTilde] "
CASE 165
  Out$ = "\[CapitalNTilde] "
CASE 168
  Out$ = "\[DownQuestion] "
CASE 170
  Out$ = "\[Not] "
CASE 173
  Out$ = "\[DownExclamation] "
CASE 174
  Out$ = "\[LeftGuillemet] "
CASE 175
  Out$ = "\[RightGuillemet] "
CASE 224
  Out$ = "\[Alpha] "
CASE 225
  Out$ = "\[SZ] "
CASE 226
  Out$ = "\[CapitalGamma] "
CASE 227
  Out$ = "\[Pi] "
CASE 228
  Out$ = "\[CapitalSigma] "
CASE 229
  Out$ = "\[Sigma] "
CASE 230
  Out$ = "\[Micro] "
CASE 231
  Out$ = "\[Gamma] "
CASE 232
  Out$ = "\[CapitalPhi] "
CASE 233
  Out$ = "\[Theta] "
```

```

CASE 234
    Out$ = "\[CapitalOmega]"
CASE 235
    Out$ = "\[Delta]"
CASE 236
    Out$ = "\[Infinity]"
CASE 237
    Out$ = "\[OSlash]"
CASE 238
    Out$ = "\[Epsilon]"
CASE 239
    Out$ = "\[Intersection]"
CASE 240
    Out$ = "\[Congruent]"
CASE 241
    Out$ = "\[PlusMinus]"
CASE 242
    Out$ = "\[GreaterEqual]"
CASE 243
    Out$ = "\[LessEqual]"
CASE 246
    Out$ = "\[Divide]"
CASE 247
    Out$ = "\[TildeTilde]"
CASE 248
    Out$ = "\[Degree]"
CASE 249
    Out$ = "\[Bullet]"
CASE 250
    Out$ = "\[CenterDot]"
CASE 251
    Out$ = "\[Sqrt]"
CASE 254
    Out$ = "\[FilledRectangle]"
CASE 127 TO 254
    Out$ = LCASE$(HEX$(HTML%(ASCII% - 126)))
    IF LEN(Out$) = 2 THEN
        Out$ = "\" + Out$
    ELSE
        Out$ = "\:" + STRING$(4 - LEN(Out$), ASC("0")) + Out$
    END IF
CASE ELSE
    Out$ = CHR$(ASCII%)
END SELECT
CopyData$ = CopyData$ + Out$
NEXT y%
PRINT #PrintFile%, CHR$(34) + CopyData$ + CHR$(34);
RETURN 'MathematicaOutput _____'

```

SUBROUTINES AND FUNCTIONS

```

'=====
FUNCTION KeyInput$
'=====
' Waits for a keyboard input and gives back the corresponding ASCII letter.

```

```

'
' Handling:
' 8/18/2001: Norbert Suedland and Eckhard Walter, Adelshofen
' Check:
' 8/18/2001: Norbert Suedland and Eckhard Walter, Adelshofen
' 9/ 6/2002: Norbert Suedland, Aalen
'-----'
DIM answer$ 'AS STRING

'Clear the Keyboard Buffer:
'-----'
WHILE INKEY$ <> ""
WEND

'Question Keyboard again, until a Key has been Pressed:
'-----'
answer$ = ""
WHILE LEN(answer$) = 0
  answer$ = INKEY$
WEND

'Result:
'-----'
KeyInput$ = answer$
END FUNCTION 'KeyInput$ _____

'=====
FUNCTION Load$ (File%, RecordLength%, FileLength%, Position%)
'=====
' Will load the `Position`th record of a binary `File%`, of which the
' `RecordLength%` is constant.
' The 0th record is used for documentation.
'
' Handling:
' 8/ 4/2001 - 1/20/2003: Norbert Suedland
' Translation:
' 1/14/2009 Norbert Suedland
'-----'
DIM Array$ 'AS STRING
DIM l% 'AS INTEGER

Array$ = SPACE$(RecordLength%)
IF Position% = 0 THEN
  Position% = FileLength%
END IF
GET #File%, Position% * RecordLength% + 1, Array$

Load$ = Array$
END FUNCTION 'Load$ _____

'=====
FUNCTION Part$ (Data$, Variable$)
'=====
' Will extract the part of `Data$` that is given by `Variable$`.
'
' Handling:

```

```

' 8/ 4/2001 - 9/ 4/2001:    Norbert Südland, Munich
' Translation:
' 1/14/2009:                Norbert Südland, Aalen
' -----
DIM Seek$      'AS STRING
DIM Result$
DIM FurtherSeek$
DIM Length%    'AS INTEGER
DIM p%
DIM Where%

Length% = LEN(Variable$)
p% = STRLEN%(Variable$, ".") + 1
IF p% = Length% THEN
    Seek$ = Variable$
ELSE
    p% = p% + STRLEN%(MID$(Variable$, p% + 1, Length% - p%), ".") + 1
    IF p% < Length% THEN
        Seek$ = LEFT$(Variable$, p% - 1)
    ELSE
        Seek$ = Variable$
    END IF
END IF
Where% = QuickPosition%(GVName$(), Seek$)
IF Where% = 0 THEN
    Result$ = ""
ELSE
    Result$ = MID$(Data$, GVBegin%(Where%), GVLength%(Where%))
    IF p% < Length% THEN
        FurtherSeek$ = GVType$(Where%) + MID$(Variable$, p%, Length% - p% + 1)
        Result$ = Part$(Result$, FurtherSeek$)
    END IF
END IF
Part$ = Result$
END FUNCTION 'Part$

' =====
' SUB Pause
' =====
' Will present a statement in line 25 and wait for a pressed key.
' This pause gives the opportunity to interrupt the program and look at the
' program code.
' Under Windows XP, the command STOP not always works reliably.
'
' Handling:
' 9/ 4/2001: Norbert Südland, Munich
' 8/18/2001: Norbert Südland and Eckhard Walter, Adelshofen
' 10/10/2016: Norbert Südland, Aalen
' 10/21/2022: Norbert Südland, Aalen
' Check:
' 8/18/2001: Norbert Südland and Eckhard Walter, Adelshofen
' Translation:
' 1/14/2009: Norbert Südland, Aalen
' 10/13/2016: Norbert Südland, Aalen
' -----
DIM answer$    'AS STRING
DIM x%         'AS INTEGER

```

```
DIM y%
```

```
'Get Current Cursor Position:
'-----'
```

```
x% = CSRLIN
y% = POS(0)
```

```
'Clear Line 25:
'-----'
```

```
COLOR 7, 0
LOCATE 25, 1
PRINT SPACE$(80);
```

```
'Print Text Messages:
'-----'
```

```
LOCATE 25, 6
COLOR 0, 7
PRINT " Press any key to go on ";
COLOR 15, 0
PRINT " View Program Code: ";
COLOR 0, 7
IF ENVIRON$("COMSPEC") = "Z:\COMMAND.COM" THEN
    PRINT " [ Ctrl ] - [ Scroll ] ";
ELSEIF ENVIRON$("DOSDIR") <> "" THEN
    PRINT " [ Ctrl ] - [ Scroll ] ";
ELSE
    PRINT " [ Ctrl ] - [ Pause ] ";
END IF
COLOR 7, 0
```

```
'Wait for Pressed Key:
'-----'
```

```
answer$ = KeyInput$
```

```
'Clear Line 25:
'-----'
```

```
LOCATE 25, 1
PRINT SPACE$(80);
```

```
END SUB 'Pause _____'
```

```
'=====
SUB Present (Colors%, Quest$, Buffer$, Offset%, Kind$, Area%, Areas%, ly%)
'=====
' Presents `Quest$`, followed by `Buffer$` in line `ly%`.
'
' Meaning of the further parameters:
' `Colors` Choice mode 1 (normal) or 2 (emphasized)
' `Offset` Shifting possibility within a column
' `Area` Wanted column
' `Areas` Total number of columns
' `Kind` "L" (left hand sided), "C" (centered),
' "R" (right hand sided)
'
' Handling:
' 8/ 7/2001 - 9/ 4/2001: Norbert Südland and Eckhard Walter, Adelshofen
'-----'
```

```

DIM Length%      'AS INTEGER
DIM x%

SELECT CASE Colors%
CASE 1
    COLOR 7, 0
CASE 2
    COLOR 15, 0
END SELECT
Length% = LEN(Quest$) + LEN(Buffer$) + 2
IF LEN(Buffer$) = 0 THEN
    Length% = Length% - 2
END IF
IF Length% > 80 / Areas% THEN
    Length% = 80 / Areas%
    IF LEN(Quest$) >= Length% THEN      'Programming discrepancy!
        Quest$ = LEFT$(Quest$, Length%)
        Buffer$ = ""
    ELSE
        Pause
        Buffer$ = LEFT$(Buffer$, LEN(Quest$) - Length%)
    END IF
END IF

SELECT CASE Kind$
CASE "L"
    x% = INT(Offset% + (Area% - 1) * 80 / Areas%)
CASE "C"
    x% = INT(Offset% + (Area% - .5) * 80 / Areas% - Length% / 2) + 1
CASE "R"
    x% = INT(Offset% + Area% * 80 / Areas% - Length%)
END SELECT
LOCATE ly%, x%
PRINT Quest$;

IF Buffer$ <> "" THEN
    COLOR 0, 7
    SELECT CASE Colors%
    CASE 1
        PRINT " "; Buffer$; " ";
    CASE 2
        IF Buffer$ <> "" THEN
            x% = POS(0)
            LOCATE ly%, x% + 1
            PRINT Buffer$;
        END IF
    END SELECT
END IF

COLOR 7, 0
END SUB 'Present _____'

'=====
FUNCTION QuickPosition% (NameList$(), Name$)
'=====
' Will seek in `NameList`() for `Name$`.
' - There is a presupposition, that `NameList`() is sorted alphabetically,

```

```

'   where all special letters are dealed due to their ASCII numbers.
' - The result is clear, if `Name$` occurs maximal once in `NameList$()`.
'   The internal indexes of `NameList$()` are not considered.
'
' Handling:
'   8/ 4/2001 - 12/27/2002:    Norbert Südland
' Check:
' -----
DIM SeekStart%      'AS INTEGER
DIM SeekEnd%
DIM Found%
DIM Check%

SeekStart% = LBOUND(NameList$)
SeekEnd% = UBOUND(NameList$)
Found% = 0
WHILE Found% = 0 AND SeekStart% <= SeekEnd%
    Check% = INT(SeekStart% / 2 + SeekEnd% / 2)      'Avoids overflow!
    SELECT CASE NameList$(Check%)
    CASE IS < Name$
        SeekStart% = Check% + 1
    CASE IS = Name$
        Found% = Check%
    CASE IS > Name$
        SeekEnd% = Check% - 1
    END SELECT
WEND

QuickPosition% = Found%
END FUNCTION 'QuickPosition% _____

'=====
FUNCTION SIZEOF% (StructureName$)
'=====
' Will give the length of the data structure being named `StructureName$`
'
' Handling:
'   8/ 4/2001 - 2/12/2003:    Norbert Südland
' Check:
'   2/12/2003:                Norbert Südland
' Translation:
'   1/14/2009:                Norbert Südland
' -----
DIM StructurePosition%      'AS INTEGER
DIM StructureLength%        'AS INTEGER

StructurePosition% = QuickPosition%(GVName$(), StructureName$)
IF StructurePosition% = 0 THEN
    Pause
    StructureLength% = 0      'Name not found: writing mistake?
ELSE
    'Read saved structure length:
    '-----
    StructureLength% = GVLength%(StructurePosition%)

```

```

    IF StructureLength% <= 0 THEN
        Pause
        ERROR 190          'Non-positive structure length has been saved!
    END IF
END IF

SIZEOF% = StructureLength%
END FUNCTION 'SIZEOF% _____

'=====
FUNCTION STRLEN% (Text$, EndCharacter$)
'=====
' Will find the string length of `Text$` until to the first occurrence of
' all `EndCharacter$`, like being usual within the C programming language.
' If `EndCharacter$` does not occur, the string length `LEN(Text$)` is
' returned.
'
' Handling:
' 8/ 4/2001: Norbert Suedland, Munich
' Check:
' 8/ 4/2001: Norbert Suedland, Munich
' Translation:
' 1/14/2009: Norbert Suedland, Aalen
'-----
DIM IntermediateResult%      'AS INTEGER

IntermediateResult% = INSTR(Text$, EndCharacter$)
IF IntermediateResult% = 0 THEN
    IntermediateResult% = LEN(Text$)
ELSE
    IntermediateResult% = IntermediateResult% - 1
END IF

STRLEN% = IntermediateResult%
END FUNCTION 'STRLEN% _____

'=====
FUNCTION ToUser$ (Symbol$)
'=====
' Will transform the international 'Symbol$' code to an English user code.
' This function will work correctly with an old data file, too.
'
' Handling:
' 11/20/2007 - 11/22/2007:    Norbert Suedland
'-----
DIM Result$
DIM c%

Result$ = SPACE$(LEN(Symbol$))
FOR c% = 1 TO LEN(Symbol$)
    SELECT CASE MID$(Symbol$, c%, 1)
        CASE "1", "J", "Y"
            MID$(Result$, c%, 1) = "Y"          'Yes
        CASE "0", "N"
            MID$(Result$, c%, 1) = "N"          'No
        CASE "{", "*"
            MID$(Result$, c%, 1) = "*"          'Birth
    
```

```
CASE "[", "A", "B"
    MID$(Result$, c%, 1) = "B"           'Begin 1st Periode
CASE "(", "W", "α"
    MID$(Result$, c%, 1) = "α"         'Begin 2nd Periode
CASE ")", "X", "Ω"
    MID$(Result$, c%, 1) = "Ω"         'End 2nd Periode
CASE "]", "E"
    MID$(Result$, c%, 1) = "E"         'End 1st Periode
CASE "}", "+"
    MID$(Result$, c%, 1) = "+"         'Death
CASE ELSE
    MID$(Result$, c%, 1) = SPACE$(1)
END SELECT
NEXT c%

ToUser$ = Result$
END FUNCTION 'ToUser$ _____ '
```