

```
«
  IFERR STD -55 CF
  { } 'FDC' STO ΣDAT
  SIZE LASTARG SWAP
  SIZE
    IF 2 ≠
    THEN OBJ→ 1 +
→ARRY STOΣ
    ELSE DROP
    END ΣDAT SIZE
  OBJ→ DROP2 DUP {
  NDTOS RNDTOS } STO
  DUP LOG 3.322 * 1 +
  SWAP √ 1 + CLLCD
  "SELECCIONE LA EQUACION:"
  { { "√n+1" 1 } {
  "1+3.322*LOG(n)" 2
  } } 1 CHOOSE
    IF 0 ==
    THEN DROP2 {
  RNDTOS NDTOS FDC }
  PURGE KILL
  END
  IF 1 ≠
  THEN DROP
  ELSE SWAP DROP
  END DUP "n="
  SWAP + 1 DISP
  "CRITERIO DE AJUSTE,RND."
  { { "Rn→ [0 RND]."
  1 } { "←Rn [IP]." 2
  } } 1 CHOOSE
    IF 0 ==
    THEN DROP {
  RNDTOS NDTOS FDC }
  PURGE KILL
  END
  IF 1 ≠
  THEN IP
  ELSE 0 RND
  END CLLCD 'NDI'
  STO MAXΣ MINΣ -
  'RANGO' STO RANGO
  NDI / 'TDI' STO { }
  'LINT' STO MINΣ 1
  NDI 1 -
  FOR STEP DUP
```

POR: J. COLBERT. E-MAIL: javiercolbert@yahoo.com.

```

TDI + DUP 3 ROLL {
} + + 'LINT' STO+
  NEXT LINT +
MAXΣ + SORT OBJ→ 2
/ 2 DUP →LIST →ARRY
→COL DROP SWAP OBJ→
OBJ→ - →LIST DUP FP
{ } 'PRC' STO SIZE
LASTARG SWAP 'c'
STO EVAL 1 c
  FOR STEP DUP
    IF 0 ==
      THEN .1 'PRC'
STO+ DROP
    ELSE →STR
SIZE 1 - NEG "1E"
SWAP + OBJ→ 'PRC'
STO+
  END
  NEXT PRC { c
PRC } PURGE ADD +
OBJ→ COL→ SWAP 2
COL→ OBJ→ EVAL *
→LIST DUP 'LINT'
STO OBJ→ 2 / 'd'
STO ΣDAT OBJ→ EVAL
DROP →LIST 3 ROLL
1 d
  FOR STEP 'B'
STO 'A' STO 0
'FCLAS' STO EVAL 1
NDTOS
  FOR STEP DUP
A ≥ SWAP DUP B ≤
ROT
    IF ==
      THEN 1
'FCLAS' STO+ DROP
'NDTOS' DECR DROP
    ELSE NDTOS
ROLLD
  END
  NEXT FCLAS
'FDC' STO+ NDTOS
→LIST
  IFERR 3 ROLL
  THEN DROP2

```

```
END
NEXT FDC DUP
'FDC' STO { FCLAS A
B d } PURGE OBJ→ 1
2 →LIST →ARRY
RNDTOS 'NDTOS' STO
LINT EVAL NDI 2 2
→LIST →ARRY 1 COL+
{ LINT RNDTOS FDC }
PURGE DUP 'DATA'
STO # A2010h
LIBEVAL
THEN DROP 'FDC'
PURGE
END
» 'CINT.LT' STO
      # 6202h
      1511 BYTES.
```