```
100 LET reserved_start=RESPR(1000)
110 DIM answer$(5)
120 DIM Earth$(10)
130 DIM Radius$(10)
140 DIM SA_Lat$(5)
150 DIM SA_Long$(5)
160 DIM new_continent$(50)
170 DIM SA_Rot$(5)
180 DIM Af_Lat$(5)
190 DIM Af_Long$(5)
200 DIM Af_Rot$(5)
210 DIM new_age$(10)
220 DIM save_name$(20)
230 DIM seconds$(20)
240 DIM temp$(5)
250 DIM dcontin$(50)
260 DIM c_age$(10)
270 DIM dage$(10)
280 DIM match$(10)
290 DIM total_string$(100)
300 DIM low_byte$(5)
310 DIM longstring$(10)
320 DIM latstring$(10)
330 DIM c_contin$(50)
340 DIM high_byte$(5)
350 DIM key$(1)
360 DIM question$(50)
370 DIM dat$(5)
380 DIM default$(5)
390 DIM title$(50)
400 LET backed_up=0
410 LET active_red=0
420 REMark START UP
430 REMark EXPANDING EARTH PROGRAM
440 REMark by S.Hurrell
450 LET set_mode%=8
460 init
470 print_title "EXPANDING EARTH PROGRAM"
480 PRINT #2,"    □ S.Hurrell '95 (Version 1.09)"
490 PRINT #2\\"         Start   Info   Help"
500 LET key$=INKEY$(5000)
510 IF key$=="s": GO TO 580: END IF
520 IF key$=="i": Info: GO TO 470: END IF
530 IF key$=="h": help: GO TO 470: END IF
540 PRINT #2\\"Press the capital letter of a command"
550 PRINT #2\"to choose that operation."
560 PRINT #2\"ie  To Start press 'S' key.";
570 IF INKEY$(3000)<>"s": PRINT #2!!"Press 'S' key.";: GO TO 570: END IF
580 CLS #2
590 PRINT #2,"           Turbo Compiled"
600 PAUSE 25: CLS #2
610 REPeat main_loop
620  check_commands
630 END REPeat main_loop
640 :
650 DEFine PROCedure init
660 OPEN #0,con: OPEN #1,con
670 DIM age$(5):DIM earth_age$(5)
```

```
680 DIM continent$(50):DIM long$(5):DIM lat$(5):DIM strip$(50)
690 DIM name$(50):DIM strin$(50)
700 IF set_mode%=8 THEN
710  MODE 8
720  LET black=0 : blue=1 : red=2 : magenta=3 : green=4
730  cyan=5 : yellow=6 : white=7
740 ELSE
750  MODE 4
760  LET black=0 : blue=255 : red=2 : magenta=253 : green=4
770  cyan=125 : yellow=11 : white=7
780 END IF
790  scale_lat = -90 : scale_long = -180 : map_scale=180
800  start_on=0 : mercator=1
810  lat_count=1 : old_lat=1
820  globe=0 : age$="600"
830  lat=0: long=0: earth_age$=0: earth_age=0: earth_radius=6371
840  incr=1
850  OPEN #2,con: OPEN #3,con
860  WINDOW #0,130,34.2975,375,190
870  WINDOW #3,512,256,0,0 : PAPER #3,white :CLS #3 : INK #3,red
880  WINDOW #3,492,200,20,0 : STRIP #3,black
890  WINDOW #1,430,160,60,10 :PAPER #1,blue: CLS #1
900  WINDOW #2,460,54,30,200 :PAPER #2,red : INK #2,white: CLS #2
910  BORDER #2,2,black
920  SCALE #1,180,-180,-90
930  print_header
940  load_mercator_map
950 END DEFine
960 :
970  DEFine PROCedure view_window
980   CLS #2 : print_title ("DEFINE AN ENLARGED VIEW")
990   IF question ("Do you wish to view complete map?","Yes") THEN
1000    init
1010    RETurn
1020   ELSE
1030    PRINT #2;"Full Scale 1/154000000 "
1040    INPUT #2;"Input the map scale on the equator"\"1 :"!equ_scale
1050    LET map_scale =INT ((equ_scale/1.541314E8) * 180)
1060    INPUT #2;"Input the LATITUDE"!scale_lat
1070    INPUT #2;"Input the LONGTITUDE"!scale_long
1080     IF mercator THEN
1090      cal_mercator
1100      SCALE #1,map_scale,scale_long,merc_scale_lat
1110     ELSE
1120      SCALE #1,map_scale,scale_long,scale_lat
1130     END IF
1140    CLS #1 : CLS #2
1150    print_lat_long : part_lat_long
1160   END IF
1170 END DEFine
1180 :
1190  DEFine PROCedure print_lat_long
1200   OVER #1,1
1210   FOR lat_count=-90 TO 90 STEP 10
1220    LET ans=(lat_count MOD 30)
1230    IF ans THEN
1240     INK #1,black,blue
1250    ELSE
```

```
1260     INK #1,red,blue
1270    END IF
1280    IF mercator THEN
1290      cal_mercator
1300     LINE #1,-180,merc_lat_c TO 180,merc_lat_c
1310    ELSE
1320     LINE #1,-180,lat_count TO 180,lat_count
1330    END IF
1340   END FOR lat_count
1350   FOR long_count = -180 TO 180 STEP 10
1360    LET ans=long_count MOD 30
1370    IF ans THEN
1380     INK #1,black,blue
1390    ELSE
1400     INK #1,red,blue
1410    END IF
1420    LINE #1,long_count,-110 TO long_count,110
1430   END FOR long_count
1440 END DEFine
1450 :
1460 DEFine PROCedure check_commands
1470  LET key= CODE(INKEY$(50))
1480   SELect ON key
1490    ON key = 119,87 : REMark window
1500     warn: PAPER #3,white: AT #3,0,0: CLS #3,2:CLS #1
1510      view_window
1520    ON key = 79,111 : REMark outline
1530     warn
1540    ON key = 109,77 : REMark mercator
1550     warn
1560     IF NOT mercator THEN
1570      CLS #1
1580      load_mercator_map
1590      LET mercator = 1
1600     ELSE
1610      print_lat_long
1620      full_lat_long
1630     END IF
1640    ON key = 70,102 : REMark full screen
1650     warn
1660     IF mercator THEN
1670      CLS #1
1680      load_full_map
1690      LET mercator = 0
1700     ELSE
1710      print_lat_long
1720      full_lat_long
1730     END IF
1740    ON key = 82,114 : REMark redraw
1750     CLS #2
1760     warn
1770     redraw
1780     IF mercator THEN save_mercator_map
1790     IF NOT mercator THEN save_full_map
1800     CLS #2
1810    ON key = 67,99 : REMark cls
1820     PAPER #3,white:AT #3,0,0: CLS #3,2: CLS #1
1830    ON key=27 : REMark ESCape
```

```
1840     OVER #2,0
1850     IF globe THEN
1860      goto_map
1870     ELSE
1880      CLS #2 : PRINT #2,"Nothing to escape from"
1890     END IF
1900     PAUSE 100 : print_menu
1910    ON key= 68,100 : REMark delete outline
1920     CLEar_ram1
1930     warn
1940     delete_outline
1950    ON key=103,71 : REMark globe
1960     IF globe THEN init
1970     LET globe=1
1980     draw_globe:  CLS #2
1990    ON key=112,80 : REMark position plates
2000     position_plates
2010    ON key=88,120: REMark export
2020     print_title ("EXPORT DATA")
2030     EXPort
2040     PAUSE:CLS #2
2050     OVER #2,0
2060    ON key=73,105: REMark print
2070     dump
2080    ON key=83,115: REMark easel
2090     CLEAr_ram2
2100     EXEC_W flp1_easel
2110     init
2120    ON key=97,65: REMark Active
2130     active_globe: CLS #2
2140    ON key=72,104 :REMark Help
2150     help
2160    ON key=98,66: REMark Backup
2170     backup: CLS #2
2180    ON key=110,78: REMark paiNt
2190     CLEar_ram1
2200     EXEC_W flp1_painter
2210     init
2220    ON key=81,113
2230     print_title("QUIT")
2240     IF question("Do you wish to quit?","No"):  FORMAT ram1_0: FORMAT
ram2_0: STOP: END IF
2250     CLS #2
2260   END SELect
2270   print_menu
2280 END DEFine
2290 :
2300 DEFine PROCedure print_menu
2310  AT #2,0,0 : PRINT
#2;"Window","Outline","Redraw","Full"!"screen","Cls","Globe","Mercator","
Delete","ESCape","Position","prInt","eXport","paiNt","eaSel","Active","He
lp","Backup","Quit"
2320 END DEFine
2330 :
2340 DEFine PROCedure full_lat_long
2350  STRIP #3,white : INK #3,black
2360  AT #3,1,0
2370    IF mercator THEN
```

```
2380     PRINT #3,\" 60"\\\" 30"\\\"  0"\\\"-30"\\\\"-60"
2390    ELSE
2400     PRINT #3," 90"\\" 60"\\\" 30"\\"  0"\\\"-30"\\\"-60"\\"-90"
2410    END IF
2420   AT #3,17,1 : PRINT #3,"-180  -120  -60     0    60   120   180"
2430 END DEFine
2440 :
2450 DEFine PROCedure part_lat_long
2460  INK #3,black : STRIP #3,white
2470  AT #3,16,0
2480  IF scale_lat>0 THEN PRINT #3," ";
2490  IF scale_lat=0 THEN PRINT #3,"  ";
2500  PRINT #3,scale_lat
2510  AT #3,17,3 : PRINT #3,scale_long
2520 END DEFine
2530 :
2540 DEFine PROCedure line_con
2550   create_ram2
2560   CLS #2: start_on=0
2570   print_title "DEFINE PLATE OUTLINE"
2580   INPUT #2,"Which plate ?"!continent$
2590   INPUT #2,"Input the age of the crust ?"!age$
2600   LET new_continent$=continent$: new_age$=age$
2610   insert_cont
2620   CLS #2
2630   PRINT #2,"Outline"!continent$!"at"!age$!"million"!"years. ";
2640   PRINT #2," "!"Position the cursor and press 'S'"!"key"!"to start.
";
2650   PRINT #2," "!"Input clockwise."
2660    RECOL 0,7,2,0,0,0,0,0
2670    INK #1,cyan : OVER #1,-1
2680    LET lat=scale_lat+9 : long=scale_long+9
2690     IF mercator THEN
2700      cal_mercator
2710      LINE #1,-250,merc_lat TO 250,merc_lat
2720      LET merc_cursor_lat=merc_lat
2730     ELSE
2740      LINE #1,-250,lat TO 250,lat
2750     END IF
2760    LINE #1,long,-90 TO long,90
2770    OPEN_NEW #4,"ram2_"&continent$&"_"&age$
2780    PRINT #4,continent$\age$
2790  REPeat out_loop
2800   LET key = CODE(INKEY$(-1))
2810    SELect ON key
2820      CLS #2 : print_menu
2830     ON key = 115,83: REMark start
2840      out (long): out (lat)
2850      LET old_long = long : old_lat = lat : start_on = 1
2860      start_long=long : start_lat=lat
2870      AT #3,19,3: INK #3,blue: PRINT #3,"ESCape on LAT
="!start_lat!"LONG ="!start_long
2880      incr=.5
2890      CLS #2 : print_menu
2900     ON key = 87,119: REMark window
2910      IF NOT start_on THEN PRINT #4,9\9
2920      CLOSE #4
2930      view_window
```

```
2940      redraw
2950      RECOL 0,7,2,0,0,0,0,0
2960      OVER #1,-1: INK #1,red
2970      LET lat=old_lat : long=old_long
2980      IF mercator THEN
2990        cal_mercator
3000        LINE #1,0,merc_lat TO 360,merc_lat
3010      ELSE
3020        LINE #1,0,lat TO 360,lat
3030       END IF
3040        LINE #1,long,0 TO long,180
3050      IF start_on THEN
3060       OPEN #4,"ram2_"&new_continent$&"_"&new_age$
3070        REPeat reop_loop
3080         IF EOF(#4) THEN EXIT reop_loop
3090         INPUT #4,strip$
3100        END REPeat reop_loop
3110      ELSE
3120       DELETE "ram2_"&new_continent$&"_"&new_age$
3130       OPEN_NEW #4,"ram2_"&new_continent$&"_"&new_age$
3140       PRINT #4,new_continent$\new_age$
3150      END IF
3160     ON key = 208: REMark up
3170       IF start_on THEN
3180        out (long): out (lat)
3190        IF mercator THEN
3200         cal_mercator
3210         LINE #1, old_long,merc_old_lat TO long,merc_lat
3220        ELSE
3230         LINE #1, old_long,old_lat TO long,lat
3240        END IF
3250         LET old_long = long : old_lat = lat
3260      END IF
3270      LET lat = lat +incr
3280       print_position
3290       IF mercator THEN
3300        cal_mercator
3310        LINE #1,-250,merc_cursor_lat TO 250,merc_cursor_lat : LINE
#1,-250,merc_lat TO 250,merc_lat
3320        LET merc_cursor_lat=merc_lat
3330       ELSE
3340        LINE #1,-250,lat-1 TO 250,lat-1 : LINE #1,-250,lat TO 250,lat
3350       END IF
3360     ON key = 216: REMark down
3370      IF start_on THEN
3380         out (long): out (lat)
3390          IF mercator THEN
3400           cal_mercator
3410           LINE #1, old_long,merc_old_lat TO long,merc_lat
3420          ELSE
3430           LINE #1, old_long,old_lat TO long,lat
3440          END IF
3450         LET old_long = long : old_lat=lat
3460        END IF
3470      LET lat = lat -incr
3480      print_position
3490       IF mercator THEN
3500        cal_mercator
```

```
3510        LINE #1,-250,merc_cursor_lat TO 250,merc_cursor_lat : LINE
#1,-250,merc_lat TO 250,merc_lat
3520        LET merc_cursor_lat=merc_lat
3530       ELSE
3540        LINE #1,-250,lat+1 TO 250,lat+1 : LINE #1,-250,lat TO 250,lat
3550       END IF
3560     ON key = 192: REMark left
3570      LET long = long -incr
3580      print_position
3590      LINE #1,long+incr,-200 TO long+incr,200 : LINE #1,long,-200 TO
long,200
3600     ON key = 200: REMark right
3610      LET long = long +incr
3620      print_position
3630      LINE #1,long-incr,-200 TO long-incr,200 : LINE #1,long,-200 TO
long,200
3640     ON key = 27: REMark ESCape
3650      IF start_long=long AND start_lat=lat
3660       IF question ("Do you wish to Quit outline","Yes")
3670        IF NOT start_on THEN PRINT #4,0\0
3680        LET start_on=0
3690        LET incr=1
3700        OVER #1,1
3710        CLOSE #4
3720        COPY
"ram2_"&new_continent$&"_"&new_age$,"flp1_"&new_continent$&"_"&new_age$
3730        CLS #1 : CLS #2
3740        AT #3,17,0 : CLS #3,2
3750        redraw
3760        RETurn
3770       END IF
3780      ELSE
3790       IF question ("Finish without matching start position?","No")
3800        LET long=start_long: lat=start_lat
3810        PRINT #2,"Press ESCape again to finish!"
3820       ELSE
3830        PRINT #2,"ERROR : Match LAT ="!start_lat!"LONG
="!start_long!"to"!"finish. ": PAUSE 50
3840       END IF
3850      END IF
3860    ON key=32: REMark space
3870     PRINT #4,"!"
3880     LET start_on=0: incr=1
3890     PRINT #2,"Press 'S' key to re-start at new position"
3900     END SELect
3910    END REPeat out_loop
3920  END DEFine
3930 :
3940  DEFine PROCedure redraw
3950   CLS #2 : LET quit=0: start_on=1: fill_on_redraw=0
3960   print_title ("REDRAW ALL PLATES")
3970   backup_to_ram2
3980   reset_earth_age
3990   print_header
4000   LET quick_on=0
4010   IF question("Do quick redraw?","Yes"): quick_on=1
4020   OVER #1,1
4030   OPEN #5,ram2_continents
```

```
4040    REPeat re_loop
4050     LET key=CODE(INKEY$(#2))
4060     IF key=27: CLOSE #5: CLS #2: RETurn
4070     IF EOF(#5) THEN
4080      CLOSE #5
4090      CLS #2
4100      RETurn
4110     ELSE
4120      REPeat check_age
4130       INPUT #5,continent$,age$
4140       IF earth_age<=age$ THEN EXIT check_age
4150      END REPeat check_age
4160      IF earth_age THEN
4170       pick_up_positions
4180      END IF
4190     IF NOT quit THEN
4200      PRINT #2,"Outlining "; : strip_header
4210      LET long=in: lat=in: checked_long%=long: checked_lat%=lat
4220      IF earth_age THEN reposition
4230       IF mercator THEN
4240        cal_mercator
4250        LINE #1,long,merc_lat
4260       ELSE
4270        LINE #1,long,lat
4280       END IF
4290         REPeat re_sec_loop
4300         LET key=CODE(INKEY$(#2))
4310         IF key=27: CLOSE #5: CLOSE #4: CLS #2: RETurn
4320          IF quit THEN EXIT re_sec_loop
4330          IF EOF(#4) THEN
4340           CLOSE #4
4350           EXIT re_sec_loop
4360          ELSE
4370           IF quick_on THEN quick_draw: IF quit: EXIT re_sec_loop
4380           long=in: IF quit THEN EXIT re_sec_loop
4390           lat=in: IF quit THEN EXIT re_sec_loop
4400           check_input: IF data_error: EXIT re_sec_loop
4410           IF earth_age THEN reposition
4420           IF mercator THEN
4430             cal_mercator
4440             IF start_on THEN
4450              LINE #1 TO long,merc_lat
4460             ELSE
4470              LINE #1,long,merc_lat
4480             END IF
4490            ELSE
4500             IF start_on THEN
4510              LINE #1 TO long,lat
4520             ELSE
4530              LINE #1,long,lat
4540             END IF
4550            END IF
4560           END IF
4570           LET start_on=1
4580         END REPeat re_sec_loop
4590         LET quit=0
4600        END IF
4610       END IF
```

```
4620     END REPeat re_loop
4630  END DEFine
4640 :
4650  DEFine PROCedure cal_mercator
4660   IF lat<85 AND lat>-85 THEN
4670    merc_lat=(INT(((1/COS (RAD(lat)))^(1/3))*lat*10))/10
4680    merc_lat_c=INT(((1/COS (RAD(lat_count)))^(1/3))*lat_count)
4690    merc_old_lat=(INT(((1/COS (RAD(old_lat)))^(1/3))*old_lat*10))/10
4700    merc_scale_lat=INT(((1/COS (RAD(scale_lat)))^(1/3))*scale_lat)
4710   ELSE
4720    IF lat>85 THEN
4730     merc_lat=85 : merc_old_lat=85 : merc_scale_lat=85
4740    END IF
4750    IF lat<-85 THEN
4760     merc_lat=-85 : merc_old_lat=-85 : merc_scale_lat=-85
4770    END IF
4780   END IF
4790  END DEFine
4800 :
4810 DEFine PROCedure delete_outline
4820   create_ram2
4830   CLS #1: INK #1,white: printout_cont
4840   CLS #2
4850   print_title ("DELETE PLATE OUTLINE")
4860   INPUT #2,"Which continental plate ?"!dcontin$
4870   INPUT #2,"Input the age of this crust ?"!dage$
4880   OPEN #5,flp1_continents
4890   OPEN_NEW #4,ram2_cont_temp : LET de_cont=0
4900    REPeat input_con
4910     IF EOF(#5) THEN EXIT input_con
4920      INPUT #5,continent$,age$
4930      IF continent$&age$=dcontin$&dage$ THEN
4940       DELETE "flp1_"&dcontin$&"_"&dage$
4950       CLS #2 : LET de_cont=1
4960       PRINT #2,dcontin$;" Plate - ";dage$;" million
years"\"DELETED."
4970      ELSE
4980       PRINT #4,continent$\age$
4990      END IF
5000     END REPeat input_con
5010    CLOSE #5 : CLOSE #4 : DELETE flp1_continents
5020    COPY ram2_cont_temp TO flp1_continents
5030    DELETE ram2_continents
5040    COPY ram2_cont_temp TO ram2_continents : DELETE ram2_cont_temp
5050    IF NOT de_cont THEN PRINT #2,dcontin$!"Plate NOT FOUND"
5060    PAUSE 300 : CLS #2
5070 END DEFine
5080 :
5090 DEFine PROCedure print_position
5100  INK #3,red: STRIP #3,white
5110  AT #3,18,2 : PRINT #3,"Cursor @ LAT=";lat;"   "
5120  AT #3,18,23 : PRINT #3,"& LONG=";long;"   "
5130 END DEFine
5140 :
5150 DEFine PROCedure RESAVE
5160  SAVE flp1_earth_bas
5170 END DEFine
5180 :
```

```
5190 DEFine PROCedure globe_prompt_window
5200  WINDOW #2,120,236,380,10
5210  PAPER #2,black : INK #2,red : CLS #2
5220 END DEFine
5230 :
5240  DEFine PROCedure draw_globe
5250   LET draw_on_screen=0: quit=0: fill_on_redraw=0
5260   LET start_on=1: quick_on=0
5270   WINDOW #1,512,256,0,0 : PAPER #1,black
5280   IF active_red THEN GO TO 5390
5290   print_title ("DRAW GLOBE")
5300   IF question ("View last globe drawn?","No") THEN
5310    globe_prompt_window : CLS #1
5320    load_globe
5330    RETurn
5340   END IF
5350   backup_to_ram2
5355   LET backed_up=1
5360   reset_earth_age
5370   IF question ("Do quick redraw?","Yes"): quick_on=1
5380   INPUT #2,"View above which longitude?"!view_long
5390   LET scale_change=6371/earth_radius
5400   SCALE #1,100*scale_change,-60*scale_change,-50*scale_change
5410   PAPER #1,black: CLS #1
5420   INK #1,blue : FILL #1,1 : CIRCLE #1,0,0,50 : FILL #1,0
5430   globe_prompt_window
5440   OPEN #5,ram2_continents
5450    LET key=CODE(INKEY$(#2))
5460    IF key=27 THEN
5470     CLOSE #5: CLS #2
5480     IF NOT active_red THEN save_globe
5490     RETurn
5500    END IF
5510    REPeat globe_loop
5520     IF EOF(#5) THEN
5530      CLOSE #5
5540      PAPER #1,white : CLS #2
5550      IF NOT active_red THEN
5560       save_globe
5570      END IF
5580      RETurn
5590     ELSE
5600      REPeat check_age
5610      INPUT #5,continent$,age$
5620       IF earth_age<=age$ THEN EXIT check_age
5630      END REPeat check_age
5640      IF earth_age THEN
5650       pick_up_positions
5660      END IF
5670     IF NOT quit THEN
5680      PRINT #2,"Outlining "; : strip_header
5690      long=in
5700      lat=in
5710      checked_long%=long: checked_lat%=lat
5720      IF earth_age THEN reposition
5730      IF fill_on_redraw: IF age$>=500: INK #1,black
5740      LET long=rotate_globe(long,view_long)
5750      LET globe_long=cal_globe(long,lat) : globe_lat=cal_globe(lat,0)
```

```
5760      LINE #1,globe_long,globe_lat
5770      REPeat globe_sec_loop
5780       LET key=CODE(INKEY$(#2))
5790       IF key=27: CLOSE #4: CLOSE #5: CLS #2: RETurn
5800          IF EOF(#4) THEN
5810           CLOSE #4
5820           EXIT globe_sec_loop
5830          ELSE
5840           long=in: IF quit THEN EXIT globe_sec_loop
5850           lat=in: IF quit THEN EXIT globe_sec_loop
5860           IF earth_age THEN reposition
5870           LET long=rotate_globe(long,view_long)
5880           LET globe_long=cal_globe(long,lat) :
globe_lat=cal_globe(lat,0)
5890            IF long>-90 AND long<90
5900             IF start_on THEN
5910              LINE #1 TO globe_long,globe_lat
5920             ELSE
5930              LINE #1,globe_long,globe_lat
5940             END IF
5950             LET draw_on_screen=1
5960            ELSE
5970             LINE #1, globe_long,globe_lat
5980            END IF
5990            LET start_on=1
6000          END IF
6010        END REPeat globe_sec_loop
6020        LET quit=0
6030      END IF
6040     END IF
6050     END REPeat globe_loop
6060  END DEFine
6070 :
6080 DEFine FuNction cal_globe(theta,delta)
6090  o_len=SIN(RAD(theta))*50
6100  b_len=80-(COS(RAD(theta))*50)
6110  alpha=ATAN(RAD(o_len/b_len))
6120  a_len=o_len/COS(RAD(alpha))
6130  c_len=COS(RAD(delta))*a_len
6140  RETurn c_len
6150 END DEFine
6160 :
6170 DEFine FuNction rotate_globe(theta,delta)
6180  IF delta=0 THEN RETurn theta
6190  LET theta=theta - delta
6200  FOR again=1 TO 2
6210  IF theta > 180 THEN LET theta=theta - 360
6220  IF theta < -180 THEN LET theta=theta + 360
6230  END FOR again
6240  RETurn theta
6250 END DEFine
6260 :
6270 DEFine PROCedure save_mercator_map
6280  print_title ("SAVE MERCATOR MAP")
6290  IF NOT question ("Overwrite existing mercator map?","No"): RETurn
6300  CLS #2
6310  DELETE flp1_mercator_map
6320  COMSAVE "flp1_mercator_map"
```

```
6330 END DEFine
6340 :
6350 DEFine PROCedure save_full_map
6360  print_title ("SAVE FULL MAP")
6370  IF NOT question ("Overwrite existing full map?","No"): RETurn
6380  CLS #2
6390  DELETE flp1_full_map
6400  COMSAVE "flp1_full_map"
6410 END DEFine
6420 :
6430 DEFine PROCedure save_globe
6440  PRINT #2,"SAVE"\"GLOBE"
6450  IF NOT question ("Overwrite existing globe?","No"): RETurn
6460  CLS #2
6470  DELETE "flp1_globe_pic"
6480  COMSAVE "flp1_globe_pic"
6490 END DEFine
6500 :
6510 DEFine PROCedure load_mercator_map
6520  COMLOAD "flp1_mercator_map"
6530 END DEFine
6540 :
6550 DEFine PROCedure load_full_map
6560  COMLOAD "flp1_full_map"
6570 END DEFine
6580 :
6590 DEFine PROCedure load_globe
6600  COMLOAD "flp1_globe_pic"
6610 END DEFine
6620 :
6630 DEFine PROCedure backup
6640  print_title ("BACKUP EARTH DISC")
6650  IF NOT question("Is original disc in FLP1_?","No"): RETurn : END IF
6660  PRINT #2,"Copying to memory.  WAIT...."
6670   create_ram1
6680   COPY flp1_readme_doc TO ram1_readme_doc
6690   COPY flp1_updates_doc TO ram1_updates_doc
6700   COPY flp1_manual_doc TO ram1_manual_doc
6710   COPY flp1_mercator_map TO ram1_mercator_map
6720   COPY flp1_full_map TO ram1_full_map
6730   COPY flp1_globe_pic TO ram1_globe_pic
6740   COPY flp1_painter_back TO ram1_painter
6750   COPY flp1_easel_back TO ram1_easel
6760   COPY flp1_earth_task TO ram1_earth_task
6770   COPY flp1_boot TO ram1_boot
6780   COPY flp1_continents TO ram1_continents
6790   COPY flp1_tk TO ram1_tk
6800   COPY flp1_ramdisc_cde TO ram1_ramdisc_cde
6810   FOR globe=1 TO 21
6820    COPY "flp1_"&globe&"_globe","ram1_"&globe&"_globe"
6830   END FOR globe
6840   OPEN_IN #4,ram1_continents
6850   REPeat move_con
6860    IF EOF(#4) THEN EXIT move_con
6870    INPUT #4,continent$,age$
6880    DELETE "ram1_"&continent$&"_"&age$
6890    COPY "flp1_"&continent$&"_"&age$ TO "ram1_"&continent$&"_"&age$
6900   END REPeat move_con
```

```
6910   FOR age=10 TO 200 STEP 10
6920    COPY "flp1_positions_"&age TO "ram1_positions_"&age
6930   END FOR age
6940   CLOSE #4
6950  IF NOT question("Is backup disc in FLP1_?","No"): RETurn : END IF
6960  IF question("Format backup disc?","No"): FORMAT flp1_Earth_Back:
END IF
6970   PRINT #2,"Copying to disc. Wait...."
6980   DELETE flp1_readme_doc
6990   COPY ram1_readme_doc TO 'flp1_README_doc'
7000   DELETE flp1_updates_doc
7010   COPY ram1_updates_doc TO 'flp1_UPDATES_doc'
7020   DELETE flp1_manual_doc
7030   COPY ram1_manual_doc TO 'flp1_MANUAL_doc'
7040   DELETE flp1_mercator_map
7050   COPY ram1_mercator_map TO flp1_mercator_map
7060   DELETE flp1_full_map
7070   COPY ram1_full_map TO flp1_full_map
7080   DELETE flp1_globe_pic
7090   COPY ram1_globe_pic TO flp1_globe_pic
7100   DELETE flp1_earth_task
7110   COPY ram1_earth_task TO flp1_earth_task
7120   DELETE flp1_painter_back
7130   COPY ram1_painter TO flp1_painter_back
7140   DELETE flp1_painter
7150   COPY ram1_painter TO flp1_painter
7160   DELETE flp1_easel_back
7170   COPY ram1_easel TO flp1_easel_back
7180   DELETE flp1_easel
7190   COPY ram1_easel TO flp1_easel
7200   DELETE flp1_boot
7210   COPY ram1_boot TO flp1_boot
7220   FOR globe=1 TO 21
7230    DELETE "flp1_"&globe&"_globe"
7240    COPY "ram1_"&globe&"_globe","flp1_"&globe&"_globe"
7250   END FOR globe
7260   DELETE flp1_continents
7270   COPY ram1_continents TO flp1_continents
7280   OPEN_IN #4,ram1_continents
7290   DELETE flp1_tk
7300   COPY ram1_tk,flp1_tk
7310   DELETE flp1_ramdisc_cde
7320   COPY ram1_ramdisc_cde TO flp_ramdisc_cde
7330   REPeat move_con
7340    IF EOF(#4) THEN EXIT move_con
7350    INPUT #4,continent$,age$
7360    DELETE "flp1_"&continent$&"_"&age$
7370    COPY "ram1_"&continent$&"_"&age$ TO "flp1_"&continent$&"_"&age$
7380   END REPeat move_con
7390   FOR age=10 TO 200 STEP 10
7400    DELETE "flp1_positions_"&age
7410    COPY "ram1_positions_"&age TO "flp1_positions_"&age
7420   END FOR age
7430   CLOSE #4
7440  PRINT #2,"Copy complete."
7450  IF question ('Do you wish another copy?','No'):GO TO 6950: END IF
7460  PAUSE 50
7470 END DEFine
```

```
7480 :
7490 DEFine PROCedure goto_map
7500  LET globe=0
7510  init
7520 END DEFine
7530 :
7540 DEFine PROCedure warn
7550   IF NOT globe THEN RETurn
7560   CLS #2
7570   goto_map
7580 END DEFine
7590 :
7600 DEFine PROCedure strip_header
7610  OPEN_IN #4,"ram2_"&continent$&"_"&age$
7620  PRINT #2,continent$!"plate"!age$!"million"!"years"!"old."
7630  INPUT #4,continent$,age$
7640  set_plate_colour
7650 END DEFine
7660 :
7670 DEFine PROCedure set_plate_colour
7680  LET age=age$
7690  IF age>=0 AND age<5: INK #1,green
7700  IF age>=5 AND age<10: INK #1,black
7710  IF age>=10 AND age<20: INK #1,red
7720  IF age>=20 AND age<30: INK #1,magenta
7730  IF age>=30 AND age<40: INK #1,cyan
7740  IF age>=40 AND age<50: INK #1,yellow
7750  IF age>=50 AND age<60: INK #1,white
7760  IF age>=60 AND age<70: INK #1,green
7770  IF age>=70 AND age<80: INK #1,red
7780  IF age>=80 AND age<90: INK #1,magenta
7790  IF age>=90 AND age<95: INK #1,cyan
7800  IF age>=95 AND age<100: INK #1,yellow
7810  IF age>=100 AND age<105: INK #1,white
7820  IF age>=105 AND age<110: INK #1,green
7830  IF age>=110 AND age<115: INK #1,black
7840  IF age>=115 AND age<125: INK #1,red
7850  IF age>=125 AND age<500: INK #1,white
7860  IF age=14: INK #1,green
7870  IF age>=600 AND age<700: INK #1,green
7880  IF age>=500 AND age<600: INK #1,magenta
7890 END DEFine
7900 :
7910 DEFine PROCedure insert_cont
7920  create_ram2
7930  LET new_cont_saved=0
7940  OPEN #5,flp1_continents
7950  OPEN_NEW #4,ram2_cont_temp
7960  REPeat input_con
7970   IF EOF(#5) THEN EXIT input_con
7980   INPUT #5,c_contin$,c_age$
7990   IF new_cont_saved THEN
8000    PRINT #4,c_contin$\c_age$
8010   ELSE
8020    IF age$<c_age$ THEN
8030     PRINT #4,continent$\age$\c_contin$\c_age$
8040     LET new_cont_saved=1
8050    ELSE
```

```
8060     PRINT #4,c_contin$\c_age$
8070    END IF
8080   END IF
8090  END REPeat input_con
8100 IF NOT new_cont_saved THEN PRINT #4,continent$\age$
8110 CLOSE #4 : CLOSE #5
8120 DELETE flp1_continents
8130 COPY ram2_cont_temp TO flp1_continents
8140 DELETE ram2_continents
8150 COPY ram2_cont_temp TO ram2_continents
8160 DELETE ram2_cont_temp
8170 END DEFine
8180 :
8190 DEFine PROCedure printout_cont
8200  backup_to_ram2
8210  OPEN #5,ram2_continents
8220  REPeat printout
8230   IF EOF(#5) THEN EXIT printout
8240   INPUT #5,continent$,age$
8250   PRINT #1," "!"*"&continent$!"@"!age$&"* ";
8260  END REPeat printout
8270  CLOSE #5
8280 END DEFine
8290 :
8300 DEFine PROCedure position_plates
8310  create_ram2
8320  CLS #2 : LET quit=0: OVER #2,1
8330  print_title ("DEFINE ANCIENT POSITIONS")
8340  INPUT #2,"Input earth age before present in"!"millions of
years."!earth_age$: earth_age=earth_age$
8350  IF question ("EDIT existing data?","Yes") THEN
8360   edit_positions
8370   IF quit THEN RETurn
8380  ELSE
8390   IF question ("DELETE existing data?","No") THEN
8400    LET over_write = 1
8410   ELSE
8420    LET over_write = 0
8430   END IF
8440  END IF
8450   PRINT #2,"Input earth radius at"!earth_age$!"million years ago":
INPUT #2,earth_radius
8460   IF over_write THEN DELETE "ram2_positions_"&earth_age$
8470  LET current=1
8480  INK #1,green : OVER #1,-1
8490  LET lat=scale_lat+90 : long=scale_long+90
8500   IF mercator THEN
8510      cal_mercator
8520      LINE #1,-180,merc_lat TO 180,merc_lat
8530      LET merc_cursor_lat=merc_lat
8540     ELSE
8550      LINE #1,-180,lat TO 180,lat
8560   END IF
8570    LINE #1,long,-90 TO long,90
8580    OPEN_NEW #4,"ram2_positions_"&earth_age$
8590    PRINT #4,earth_radius
8600  REPeat out_loop
8610   IF NOT current THEN
```

```
8620    quit_proc("POSITION"):IF quit THEN RETurn
8630    LET current=current+1
8640   END IF
8650   IF current=1
8660    INPUT #2,"Continent plate to position"!continent$
8670    PRINT #4,continent$
8680    PRINT #2,"Indicate centre of plate. Choose with space bar."
8690    LET current=current+1
8700   END IF
8710   LET key = CODE(INKEY$(-1))
8720   SELect ON key
8730    ON key = 32 :REMark space bar
8740    SELect ON current
8750    ON current=2
8760      PRINT #4,lat\long
8770      PRINT #2,"Indicate position centre moved to. Choose with space
bar."
8780     ON current=3
8790      INPUT #2,"Enter rotation."!rotation
8800      PRINT #4,lat\long\rotation
8810     LET current=-1
8820   END SELect
8830    LET old_long = long : old_lat = lat : current = current +1
8840   ON key = 87,119 : REMark window
8850      CLOSE #4
8860      view_window
8870      redraw
8880      OVER #1,-1
8890       LET lat=old_lat : long=old_long
8900       IF mercator THEN
8910        cal_mercator
8920        LINE #1,0,merc_lat TO 360,merc_lat
8930       ELSE
8940        LINE #1,0,lat TO 360,lat
8950       END IF
8960        LINE #1,long,0 TO long,180
8970       OPEN #4,"mdv2_position_"&earth_age$
8980        REPeat reop_loop
8990         IF EOF(#4) THEN EXIT reop_loop
9000         INPUT #4,strip$
9010        END REPeat reop_loop
9020     ON key = 208: REMark up
9030      LET lat = lat +incr
9040       print_position
9050       IF mercator THEN
9060        cal_mercator
9070        LINE #1,-180,merc_cursor_lat TO 180,merc_cursor_lat : LINE
#1,-180,merc_lat TO 180,merc_lat
9080        LET merc_cursor_lat=merc_lat
9090       ELSE
9100        LINE #1,-180,lat-1 TO 180,lat-1 : LINE #1,-180,lat TO 180,lat
9110       END IF
9120     ON key = 216: REMark down
9130      LET lat = lat -incr
9140      print_position
9150       IF mercator THEN
9160        cal_mercator
```

```
9170        LINE #1,-180,merc_cursor_lat TO 180,merc_cursor_lat : LINE
#1,-180,merc_lat TO 180,merc_lat
9180        LET merc_cursor_lat=merc_lat
9190       ELSE
9200        LINE #1,-180,lat+1 TO 180,lat+1 : LINE #1,-180,lat TO 180,lat
9210       END IF
9220     ON key = 192: REMark left
9230      LET long = long - incr
9240      print_position
9250      LINE #1,long+incr,-100 TO long+incr,100 : LINE #1,long,-100 TO
long,100
9260     ON key = 200: REMark right
9270      LET long = long +incr
9280      print_position
9290      LINE #1,long-incr,-100 TO long-incr,100 : LINE #1,long,-100 TO
long,100
9300     ON key = 27: REMark ESCape
9310  quit_proc("POSITION"): IF quit THEN RETurn
9320     END SELect
9330    END REPeat out_loop
9340  END DEFine
9350 :
9360 DEFine PROCedure quit_proc(name$)
9370  AT #3,19,0 : CLS #3,2
9380  PRINT #2,"Do you wish to QUIT"!name$;
9390  INPUT #2," "!answer$
9400  IF answer$(1)=="y" THEN
9410   LET start_on=0
9420   LET incr=1
9430   OVER #1,1
9440   CLOSE #4
9450   CLS #1 : CLS #2
9460   redraw
9470   LET quit=1 : RETurn
9480  ELSE
9490   LET quit=0 : RETurn
9500  END IF
9510 END DEFine
9520 :
9530 DEFine PROCedure reposition
9540 LET long=long+.1:lat=lat+.1
9550 IF centre_long=0:centre_long=centre_long+.1: END IF
9560 IF centre_lat=0:centre_lat=centre_lat+.1: END IF
9570 IF new_centre_long=0:new_centre_long=new_centre_long+.1: END IF
9580 IF new_centre_lat=0:new_centre_lat=new_centre_lat+.1: END IF
9590 lat=RAD(lat):long=RAD(long)
9600 IF centre_long<long THEN
9610   LET switch%=1
9620 ELSE
9630   LET switch%=0
9640 END IF
9650  LET sin_centre_lat=SIN(centre_lat): sin_lat=SIN(lat)
9660  cos_centre_lat=COS(centre_lat): cos_lat=COS(lat)
9670  cos_new_centre_lat=COS(new_centre_lat)
9680  sin_new_centre_lat=SIN(new_centre_lat)
9690  LET theta_x_y =
ACOS((sin_centre_lat*sin_lat)+(cos_centre_lat*cos_lat*COS(centre_long-
long)))
```

```
9700  LET length_x_y = 6371*theta_x_y
9710  LET phi_x_y= ACOS((sin_lat-
(COS(theta_x_y)*sin_centre_lat))/((SIN(theta_x_y)*cos_centre_lat)))
9720  new_theta_x_y=length_x_y/earth_radius
9730  IF rotation=0: LET new_phi_x_y=phi_x_y + rotation: END IF
9740  IF rotation<0 THEN
9750   IF switch% THEN
9760    new_phi_x_y=phi_x_y + rotation
9770    IF new_phi_x_y<0 THEN LET switch%=0: END IF
9780   ELSE
9790    new_phi_x_y=phi_x_y - rotation
9800    IF new_phi_x_y>PI: LET switch%=1: END IF
9810   END IF
9820  END IF
9830  IF rotation>0 THEN
9840   IF switch% THEN
9850    new_phi_x_y=phi_x_y + rotation
9860    IF new_phi_x_y>PI THEN LET switch%=0: END IF
9870   ELSE
9880    new_phi_x_y=phi_x_y - rotation
9890    IF new_phi_x_y<0 THEN LET switch%=1: END IF
9900   END IF
9910  END IF
9920  lat =
ASIN(((COS(new_phi_x_y)*SIN(new_theta_x_y)*cos_new_centre_lat))+((COS(new
_theta_x_y)*sin_new_centre_lat)))
9930  LET sin_lat=SIN(lat): cos_lat=COS(lat)
9940  long=(COS(new_theta_x_y)-
(sin_new_centre_lat*sin_lat))/(cos_new_centre_lat*cos_lat)
9950 IF long>1: LET long=1: END IF
9960 IF long<-1: LET long=-1: END IF
9970  long=ACOS(long)
9980  IF switch% THEN
9990   long=new_centre_long + long
10000  ELSE
10010   long=new_centre_long - long
10020  END IF
10030  lat=DEG(lat): long=DEG(long)
10040 END DEFine
10050 :
10060 DEFine PROCedure pick_up_positions
10070  OPEN #6,"ram2_positions_"&earth_age : quit=0
10080  INPUT #6,strin$: earth_radius=strin$
10090  REPeat find
10100   IF EOF(#6) THEN
10110    PRINT #2,"WARNING : No reposition data found
for"!continent$!"plate."!"It will"!"not"!"be"!"drawn."
10120    LET quit=1
10130    CLOSE #6: RETurn
10140   ELSE
10150    INPUT #6,strin$
10160    IF strin$=continent$ THEN
10170     INPUT
#6,centre_lat,centre_long,new_centre_lat,new_centre_long,rotation
10180 LET centre_lat=RAD(centre_lat): centre_long=RAD(centre_long):
new_centre_lat=RAD(new_centre_lat): new_centre_long=RAD(new_centre_long):
rotation=RAD(rotation)
10190     CLOSE #6: RETurn
```

```
10200    END IF
10210   END IF
10220  END REPeat find
10230 END DEFine
10240 :
10250 DEFine PROCedure edit_positions
10260  OPEN_IN #4,"flp1_positions_"&earth_age$
10270  DELETE ram1_positions_temp
10280  OPEN_NEW #5,ram1_positions_temp
10290  INPUT #4,strin$: earth_radius=strin$
10300  PRINT #2,"Earth radius set to"!earth_radius!"kilometre
at"!earth_age$!"million years before present."
10310  IF NOT question (" OK?","Yes") THEN
10320   INPUT #2,"New earth radius?"!earth_radius
10330  END IF
10340  PRINT #5,earth_radius
10350  REPeat loop
10360   IF EOF (#4) THEN EXIT loop
10370   INPUT
#4,continent$,centre_lat,centre_long,new_centre_lat,new_centre_long,rotat
ion
10380   PRINT #2\\"Existing"!continent$!"centre set
at"!"Lat=";centre_lat!"Long=";centre_long;"."!"Repositioned"!"centre"!"se
t"!"at"!"Lat=";new_centre_lat!"Long=";new_centre_long!"with"!rotation!"de
gree"!"rotation"!"about"!"centre.";
10390   IF NOT question (" OK?","Yes") THEN
10400    PRINT #2\\"Re-input existing"!continent$!"centre
LAT(";centre_lat;"),LONG(";centre_long;"), new centre
LAT(";new_centre_lat;"),LONG(";new_centre_long;"),ROTATION(";rotation;").
": INPUT
#2,centre_lat!centre_long!new_centre_lat!new_centre_long!rotation
10410   END IF
10420   PRINT
#5,continent$\centre_lat\centre_long\new_centre_lat\new_centre_long\rotat
ion
10430  END REPeat loop
10440  IF question ("Add a new plate?","No")
10450    PRINT #2\\"Input new PLATE": INPUT #2!continent$
10460    PRINT #2\\"Input new"!continent$!"centre LAT, LONG, new centre
LAT, LONG, ROTATION.": INPUT
#2,centre_lat!centre_long!new_centre_lat!new_centre_long!rotation
10470    PRINT
#5,continent$\centre_lat\centre_long\new_centre_lat\new_centre_long\rotat
ion
10480  END IF
10490  CLOSE #4: CLOSE #5
10500  DELETE "flp1_positions_"&earth_age$
10510  COPY ram1_positions_temp TO "flp1_positions_"&earth_age$
10520  DELETE "ram2_positions_"&earth_age$
10530  COPY ram1_positions_temp TO "ram2_positions_"&earth_age$
10540  DELETE ram1_positions_temp
10550  LET quit=1
10560  CLS #2
10570 END DEFine
10580 :
10590 DEFine PROCedure print_header
10600 CSIZE #3,2,0: AT #3,0,4: PAPER #3,magenta: CLS #3,3: INK #3,black
10610 IF mercator THEN
```

```
10620  AT #3,0,4: PRINT #3,earth_age$!"MBP"
10630  AT #3,0,12: PRINT #3,"Mercator"
10640 ELSE
10650  AT #3,0,4: PRINT #3,earth_age$!"MBP"
10660  AT #3,0,12: PRINT #3,"Full Screen"
10670 END IF
10680 AT #3,0,24: PRINT #3,earth_radius*2;" km Dia."
10690 END DEFine
10700 :
10710 DEFine PROCedure quick_draw
10720  REMark +
10730  FOR quick=1,2,3,4,5,6
10740   IF EOF(#4):CLOSE #4:quit=1:RETurn
10750   INPUT #4,dat$
10760   IF dat$="!": start_on=0: EXIT quick
10770   IF LEN(dat$)<>2
10780    INPUT #4,dat$,dat$,dat$: PRINT #2,"Data error detected"
10790    check_input
10800    IF data_error: quit=1: RETurn : ELSE : NEXT quick
10810   END IF
10820  END FOR quick
10830  REMark -
10840 END DEFine
10850 :
10860 DEFine PROCedure print_title (title$)
10870  CSIZE #2,3,1: OVER #2,0
10880  CLS #2: UNDER #2,1: INK #2,black: PAPER #2,green
10890  AT #2,0,13-(LEN(title$)/2): PRINT #2,title$
10900  UNDER #2,0: OVER #2,1: INK #2,white: PAPER #2,red
10910  PAUSE 50
10920  CSIZE #2,0,0
10930 END DEFine
10940 :
10950 DEFine PROCedure reset_earth_age
10960  REPeat reset
10970   PRINT #2,"Earth age set
to"!earth_age!"million"!"years"!"before"!"present.  Radius set
to"!earth_radius!"kilometres.";
10980   IF NOT question (" Reset Earth Age?","no") THEN EXIT reset
10990   INPUT #2,"Age of earth before present
in"!"million"!"years?"!earth_age$
11000   LET earth_age=earth_age$
11010   LET continent$="dummy"
11020   IF earth_age THEN pick_up_positions
11030   CLS #2
11040  END REPeat reset
11050 END DEFine
11060 :
11070 DEFine PROCedure check_input
11080  LET data_error=0
11090  RETurn
11100  IF start_on: LET checked_long%=long: checked_lat%=lat
11110  REPeat check_range
11120   IF (checked_lat%+5>lat AND checked_lat%-5<lat) AND
(checked_long%+5>long AND checked_long%-5<long) THEN
11130    EXIT check_range
11140   ELSE
11150    PRINT #2,"Error reading data :- remaining data ignored"
```

```
11160    LET data_error=1: RETurn
11170   END IF
11180  END REPeat check_range
11190  checked_lat%=lat: checked_long%=long
11200 END DEFine
11210 :
11220 DEFine FuNction question (question$,default$)
11230  OVER #2,0
11240  INK #2,blue: PRINT #2;!question$!"(y/n)";
11250  INK #2,white
11260  LET key$=INKEY$(-1)
11270  RECOL #2,0,7,2,3,4,5,6,7
11280  OVER #2,1
11290  IF key$=="n": PRINT #2;!"No": RETurn 0
11300  IF key$=="y": PRINT #2;!"Yes": RETurn 1
11310  PRINT #2;!default$!"(Default)"
11320  IF  default$(1)=="y": RETurn 1: ELSE RETurn 0
11330 END DEFine
11340 :
11350 DEFine FuNction in
11360  REMark +
11370  IF EOF(#4): quit=1: RETurn dat
11380  INPUT #4,dat$
11390  IF dat$="!": LET start_on=0: GO TO 11370
11400  IF LEN(dat$)<>2: INPUT #4,dat$,dat$: PRINT #2,"Data error
detected": GO TO 11370
11410  high_byte=CODE(dat$(1))
11420  low_byte=CODE(dat$(2))
11430  dat=(((high_byte * 256) + low_byte) - 32768)/10
11440  RETurn dat
11450  REMark -
11460 END DEFine
11470 :
11480 DEFine PROCedure out(dat)
11490  LET dat=32768+(dat*10)
11500  high_byte$=CHR$(INT(dat/256))
11510  low_byte$=CHR$(dat-(INT(dat/256)*256))
11520  IF high_byte$=CHR$(13): high_byte$=CHR$(12)
11530  IF low_byte$=CHR$(13): low_byte$=CHR$(12)
11540  PRINT #4,high_byte$&low_byte$
11550 END DEFine
11560 :
11570 :
11580 DEFine PROCedure backup_to_ram2
11590   IF backed_up: RETurn : END IF
11600   PRINT #2,"Accessing data."\"WAIT...."
11610   create_ram2
11620   COPY flp1_continents TO ram2_continents
11630   OPEN_IN #4,flp1_continents
11640   REMark +
11650   REPeat move_con
11660    IF EOF(#4) THEN EXIT move_con
11670    INPUT #4,continent$,age$
11680    DELETE "ram2_"&continent$&"_"&age$
11690    COPY "flp1_"&continent$&"_"&age$ TO "ram2_"&continent$&"_"&age$
11700   END REPeat move_con
11710   FOR age=10 TO 200 STEP 10
11720    COPY "flp1_positions_"&age TO "ram2_positions_"&age
```

```
11730   END FOR age
11740   REMark -
11750   CLOSE #4
11760   LET backed_up = 1
11770 END DEFine
11780 :
11790 DEFine PROCedure EXPort
11800 IF NOT question ("Overwrite existing export data?","No"): RETurn
11810 PRINT #2,"WAIT...."
11820 DELETE flp1_Earth_exp
11830 OPEN_NEW #5,flp1_Earth_exp
11840 PRINT
#5,'"Earth$","Radius","SA_Lat","SA_Long","SA_Rot","Af_Lat","Af_Long","Af_
Rot","NA_Lat","NA_Long","NA_Rot","Gr_Lat","Gr_Long","Gr_Rot","An_Lat","An
_Long","An_Rot","Ma_Lat","Ma_Long","Ma_Rot","Au_Lat","Au_Long","Au_Rot"'
11850 FOR age=10 TO 200 STEP 10
11860  OPEN_IN #4,"flp1_positions_"&age
11870  INPUT #4,strin$: earth_radius=strin$
11880  REPeat get_data
11890   IF EOF (#4) THEN EXIT get_data
11900   INPUT
#4,continent$,centre_lat,centre_long,new_centre_lat,new_centre_long,rotat
ion
11910   IF "south america" INSTR continent$ THEN
11920
SA_lat=new_centre_lat:SA_long=new_centre_long:SA_rot=rotation:SA_centre_l
at=centre_lat:SA_centre_long=centre_long
11930   END IF
11940   IF "africa" INSTR continent$ THEN
11950    LET Af_lat=new_centre_lat
11960    Af_long=new_centre_long:Af_rot=rotation:
Af_centre_lat=centre_lat
11970    Af_centre_long=centre_long
11980   END IF
11990   IF "north america" INSTR continent$ THEN
12000    NA_lat=new_centre_lat:NA_long=new_centre_long: NA_rot=rotation:
NA_centre_lat=centre_lat: NA_centre_long=centre_long
12010   END IF
12020   IF "greenland" INSTR continent$ THEN
12030    Gr_lat=new_centre_lat: Gr_long=new_centre_long: Gr_rot=rotation:
Gr_centre_lat=centre_lat: Gr_centre_long=centre_long
12040   END IF
12050   IF "Antarctica" INSTR continent$ THEN
12060    An_lat=new_centre_lat: An_long=new_centre_long: An_rot=rotation:
An_centre_lat=centre_lat: An_centre_long=centre_long
12070   END IF
12080   IF "Madagascar" INSTR continent$ THEN
12090    Ma_lat=new_centre_lat: Ma_long=new_centre_long: Ma_rot=rotation:
Ma_centre_lat=centre_lat: Ma_centre_long=centre_long
12100   END IF
12110   IF "Australia" INSTR continent$ THEN
12120    Au_lat=new_centre_lat: Au_long=new_centre_long: Au_rot=rotation:
Au_centre_lat=centre_lat: Au_centre_long=centre_long
12130   END IF
12140  END REPeat get_data
12150  IF age=10
12160  PRINT
#5,'"0",6378,'&SA_centre_lat&","&SA_centre_long&",0,"&Af_centre_lat&","&A
```

```
f_centre_long&",0,"&NA_centre_lat&","&NA_centre_long&",0,"&Gr_centre_lat&
","&Gr_centre_long&",0"&An_centre_lat&","&An_centre_long&",0"&Ma_centre_l
at&","&Ma_centre_long&",0"&Au_centre_lat&","&Au_centre_long&",0"
12170  END IF
12180  PRINT
#5,'"'&(age/10)&'",'&earth_radius&","&SA_lat&","&SA_long&","&SA_rot&","&A
f_lat&","&Af_long&","&Af_rot&","&NA_lat&","&NA_long&","&NA_rot&","&Gr_lat
&","&Gr_long&","&Gr_rot&","&An_lat&","&An_long&","&An_rot&","&Ma_lat&","&
Ma_long&","&Ma_rot&","&Au_lat&","&Au_long&","&Au_rot
12190  CLOSE #4
12200 END FOR age
12210 PRINT #5,CHR$(26)
12220 CLOSE #5
12230 PRINT #2,'Now import "earth" data into Easel'
12240 END DEFine
12250 :
12260 DEFine PROCedure dump
12270 LBYTES flp1_dump,reserved_start:CALL reserved_start:OPEN #7,ser1
12280 END DEFine
12290 :
12300 DEFine PROCedure active_globe
12310  REMark rot_globe
12320  print_title ("ACTIVE GLOBE")
12330  LET active_red=1
12340  IF question ("Show active globe?","No"): INK #1,blue: Rot_globe:
RETurn :END IF
12350  print_title ("DEFINE ACTIVE GLOBE")
12360  IF question("Define expanding globe?","No")
12370   IF backed_up=0
12380    backup_to_ram2
12390   END IF
12400   expanding_globe
12410  ELSE
12420  print_title ("DEFINE ROTATING GLOBE")
12430  IF NOT question ("Define rotating globe?","No"): RETurn :END IF
12440  maxlong=180
12450  minlong=-180
12460   IF backed_up=0
12470    backup_to_ram2
12480   END IF
12490  LET offset=(maxlong-minlong)/21
12500  reset_earth_age
12510  WINDOW #1,512,256,0,0:PAPER #1,black:CLS #1
12520 FOR rotate=1 TO 21
12530  LET view_long=maxlong
12540  draw_globe
12550  print_age
12560 save_name$="flp1_"&rotate&"_globe"
12570  DELETE save_name$: COMSAVE save_name$
12580  REMark Ver 1.07
save_name$="flp1_"&rotate&"_globe":Save_compressed_image
12590  LET maxlong=maxlong - offset
12600 END FOR rotate
12610 END IF
12620  Rot_globe
12630 END DEFine
12640 :
12650 DEFine PROCedure Rot_globe
```

```
12660  create_ram1
12670  IF question ("Quick redraw required?","No")
12680   quick_dra=1
12690   time_in_sec=1
12700   PRINT #2,"Keys are Pause, ESCape, Fast, Slow."
12710   PAUSE 150
12720  ELSE
12730   quick_dra=0
12740  END IF
12750 IF set_mode%=4: MODE 4: ELSE : MODE 8: END IF
12760 LET start=reserved_start
12770 WINDOW #1,512,256,0,0:PAPER #1,black:CLS #1
12780 REMark rotate globe
12790 FOR count=1 TO 21
12800 LET key$=INKEY$
12810 IF key$=="p" THEN GO TO 12800
12820 IF NOT quick_dra THEN time_delay: END IF
12830 save_name$='flp1_'&count&'_globe'
12840 COMLOAD save_name$
12850 IF quick_dra THEN SBYTES "ram1_temp"&count,131072,32768: END IF
12860 END FOR count
12870 IF NOT quick_dra
12880  GO TO 12780
12890 ELSE
12900  REPeat loop
12910   FOR count=1 TO 21
12920    LBYTES "ram1_temp"&count,131072
12930    LET key$=INKEY$((time_in_sec*50)+1)
12940    IF key$=="s": time_in_sec=((time_in_sec*50)+2)/50: END IF
12950    IF key$=="f": time_in_sec=((time_in_sec*50)-2)/50: END IF
12960    IF key$=CHR$(27): EXIT loop: END IF
12970    IF key$=="p": GO TO 12930: END IF
12980   END FOR count
12990  END REPeat loop
13000 END IF
13010 init
13020 END DEFine Rot_globe
13030 :
13040 DEFine PROCedure time_delay
13050  LET seconds$=DATE$: LET seconds$=seconds$(19 TO 20)
13060  key$=INKEY$: IF key$=CHR$(27) THEN GO TO 12290
13070  IF seconds$ < 15 THEN GO TO 13050
13080  SDATE 1991,2,13,16,0,0
13090 END DEFine
13100 :
13110 DEFine PROCedure expanding_globe
13120  print_title ("EXPANDING GLOBE")
13130  INPUT #2,"View at longitude"!view_long
13140  LET earth_age=200
13150  WINDOW #1,512,256,0,0:PAPER #1,black:CLS #1
13160 FOR expand=21 TO 0 STEP -1
13170  IF earth_age THEN pick_up_positions
13180  draw_globe
13190  print_age
13200  save_name$="flp1_"&(22-expand)&"_globe"
13210  DELETE save_name$
13220  COMSAVE save_name$
13230  LET earth_age=earth_age - 10
```

```
13240 END FOR expand
13250  Rot_globe
13260 END DEFine
13270 :
13280 DEFine PROCedure print_age
13290  CSIZE #1,0,0: PAPER #1,black
13300  LET temp$=age$
13310  WINDOW #1,100,250,400,4
13320  OVER #1,1: CURSOR #1,3,29:INK #1,blue: PRINT #1,"Age";:PRINT #1;"
Dia"
13330  CURSOR #1,0,30: INK #1,green: PRINT #1,"Age";: INK #1,white: PRINT
#1;" Dia"
13340  BLOCK #1,72,6,15,17,magenta
13350  IF earth_age=0 THEN
13360   BLOCK #1,70,2,15,19,red
13370  ELSE
13380   BLOCK #1,70,2,15,19,white
13390  END IF
13400  SCROLL #1,5:AT #1,2,4: OVER #1,1: INK #1,green: PRINT
#1,"0":SCROLL #1,-5
13410  FOR age=0 TO 200 STEP 2
13420   LET age$=age
13430   set_plate_colour
13440   OVER #1,0: PAPER #1,blue: AT 0,0: PRINT ,"_"
13450   PAPER #1,black: SCROLL #1,2
13460   FOR counter=10 TO 200 STEP 10
13470    IF age=counter THEN
13480    IF age<100: pos=3: ELSE : pos=2: END IF
13490     OPEN_IN #6,"ram2_positions_"&age
13500     INPUT #6,temp$
13510     IF age=earth_age THEN
13520      BLOCK #1,72,6,15,17,magenta
13530      BLOCK #1,temp$/90,2,15,19,red
13540     ELSE
13550      BLOCK #1,72,6,15,17,magenta
13560      BLOCK #1,temp$/90,2,15,19,white
13570     END IF
13580     CLOSE #6
13590    SCROLL #1,-5: OVER #1,1
13600     INK #1,green: CSIZE #1,0,0: AT #1,1,pos: PRINT age
13610    SCROLL #1,5: OVER #1,0
13620    END IF
13630   END FOR counter
13640  END FOR age
13650  SCROLL 7
13660  OVER #1,1: INK #1,white: CSIZE #1,3,1: CURSOR #1,2,0:  PRINT
#1;earth_age!"Ma"
13670  INK #1,red: CSIZE #1,3,1: CURSOR #1,0,1:  PRINT #1;earth_age!"Ma"
13680 END DEFine
13690 :
13700 DEFine PROCedure backup_to_flp1
13710   DELETE flp1_continents
13720   COPY ram1_continents TO flp1_continents
13730   OPEN_IN #4,ram1_continents
13740   REPeat move_con
13750    IF EOF(#4) THEN EXIT move_con
13760    INPUT #4,continent$,age$
13770    DELETE "flp1_"&continent$&"_"&age$
```

```
13780    COPY "ram2_"&continent$&"_"&age$ TO "ram1_"&continent$&"_"&age$
13790   END REPeat move_con
13800   FOR age=10 TO 200 STEP 10
13810    DELETE "flp1_positions_"&age
13820    COPY "ram1_positions_"&age TO "flp1_positions_"&age
13830   END FOR age
13840   CLOSE #4
13850 END DEFine
13860 :
13870 DEFine PROCedure Info
13880 MODE 4
13890 INK #1,black: PAPER #1,white
13900 INK #3,black: PAPER #3,white
13910 INK #2,black: PAPER #2,white: CLS #1:CLS#3:CLS#2
13920 PRINT #1,"                    THE EXPANDING EARTH"
13930 PRINT #1\\"The earth was a very different place"!"when the
dinosaurs dominated"!"the land.  About 200 million years ago
the"!"continents of Europe,"!"North America, South America, Africa"!"and
Antarctica were joined"!"together into one gigantic"!"supercontinent.  As
the earth aged"!"the continents separated until they had moved to their
present-day"!"positions, forming the ocean floor in the process. Today
the"!"oldest ocean floor is closest to these ancient continents
whilst"!"new ocean floor is still forming in the centre of the ocean."
13940 CLS #2: contin
13950 CLS #1: PRINT #1,"                    THE EXPANDING EARTH"
13960 PRINT #1\\"This program allows the continents of the earth to be
repositioned"!"as they were up to 200 million years ago.  After choosing
a"!"time in the past, each continent of that time can be moved
and"!"rotated to refit the continents in their ancient positions.
The"!"whole sequence of continental movement can then be animated
to"!"show the continuous movement of the continents.  A range
of"!"continental movement has already been entered.  The ocean
crust"!"surrounding a continent block is assumed to be attached to
the"!"continent so as a sequence is run the newly formed ocean
floor"!"appears at the ocean ridges.  The ocean floor is not drawn if
it"!"has not formed at that time."
13970 CLS #2: contin
13980 CLS #1
13990 PRINT #1,"                    THE EXPANDING EARTH"
14000 PRINT #1\\"The further back in time the smaller the area of the
ocean floor:"!"one theory which attempts to explain this is that the
earth has"!"expanded in size during this time.  This is the Expanding
Earth"!"theory. It is this expansion which has torn the continents
apart."!"The continental positions supplied with this program take
account"!"of this expansion by allowing the diameter of the earth to
be"!"progressively increased in size as time approaches the
present."!"The concept of the Expanding Earth is described in more detail
in"!"the book 'Dinosaurs and the Expanding Earth' available from"!"One-
off publishing. (Get it from the library)."\\"There is no more on-line
help."
14010 CLS #2: contin
14020 MODE 8
14030 REinit
14040 END DEFine
14050 :
14060 DEFine PROCedure contin
14070 CLS #2
14080 INK #2,red: AT #2,2,2: PRINT #2,"PRESS SPACE BAR TO CONTINUE"
```

```
14090 LET key$=INKEY$(100)
14100 IF key$=" " THEN GO TO 14150
14110 INK #2,black: AT #2,2,2: PRINT #2,"PRESS SPACE BAR TO CONTINUE":
14120 LET key$=INKEY$(100)
14130 IF key$=" " THEN GO TO 14150
14140 GO TO 14080
14150 CLS #2
14160 END DEFine
14170 :
14180 DEFine PROCedure CLEar_ram1
14190  REMark ram1 is pictures
14200  LET backed_up=1
14210  REMark INK #1,blue: PAPER #1,blue: OVER #1,0
14220  FORMAT ram1_1
14230  REMark create ram2 space for data
14240  FORMAT ram2_600
14250  REMark OVER #1,1: INK #1,black
14260 END DEFine
14270 :
14280 DEFine PROCedure CLEAr_ram2
14290  REMark ram2 is data
14300  LET backed_up=0
14310  REMark INK #1,blue: PAPER #1,blue: OVER #1,0
14320  FORMAT ram2_1
14330  REMark create ram1 space for pictures
14340  FORMAT ram1_1400
14350  REMark OVER #1,1: INK #1,black
14360 END DEFine
14370 :
14380 DEFine PROCedure create_ram1
14390  CLEAr_ram2
14400 END DEFine
14410 :
14420 DEFine PROCedure create_ram2
14430  CLEar_ram1
14440 END DEFine
14450 :
14460 DEFine PROCedure help
14470 INK #1,black: PAPER #1,white: CLS #1
14480 INK #3,black: PAPER #3,white: CLS #3
14490 CLS #2
14500 PRINT #1\\'       HELP'\\'Choose a menu you wish help with'
14510 print_menu
14520 LET key$=INKEY$(-1)
14530 CLS #1
14540 IF key$=='w' THEN
14550 PRINT#1\'     WINDOW'\
14560 END IF
14570 IF key$=='o' THEN
14580 PRINT#1\'    OUTLINE'
14590 END IF
14600 IF key$=='r' THEN
14610 PRINT #1\'    REDRAW'
14620 END IF
14630 IF key$=='f' THEN
14640 PRINT #1\'    FULL SCREEN'
14650 END IF
14660 IF key$=='c' THEN
```

```
14670 PRINT #1\'    CLS'
14680 END IF
14690 IF key$=='g' THEN
14700 PRINT #1\'    GLOBE'
14710 END IF
14720 IF key$=='m' THEN
14730 PRINT #1\'    MERCATOR'
14740 END IF
14750 IF key$=='d' THEN
14760 PRINT #1\'    DELETE'
14770 END IF
14780 IF key$=='ESCape' THEN
14790 PRINT #1\'    ESCAPE'
14800 END IF
14810 IF key$=='p' THEN
14820 PRINT #1\'    POSITION'
14830 END IF
14840 IF key$=='i' THEN
14850 PRINT #1\'    PRINT'
14860 END IF
14870 IF key$=='x' THEN
14880 PRINT #1\'    EXPORT'
14890 END IF
14900 IF key$=='n' THEN
14910 PRINT #1\'    PAINT'
14920 END IF
14930 IF key$=='s' THEN
14940 PRINT #1\'    EASEL'
14950 END IF
14960 IF key$=='a' THEN
14970 PRINT #1\'    ACTIVE'
14980 END IF
14990 IF key$=='h' THEN
15000 PRINT #1\'    HELP'
15010 END IF
15020 IF key$=='b' THEN
15030 PRINT #1\'    BACKUP'
15040 END IF
15050 IF key$=='q' THEN
15060 PRINT #1\'    QUIT'
15070 END IF
15080 CLS #2: contin
15090 REinit
15100 END DEFine
15110 :
15120 DEFine PROCedure REinit
15130  MODE 8
15140  WINDOW #0,130,60,375,190
15150  WINDOW #3,512,256,0,0 : PAPER #3,white : INK #3,red
15160  WINDOW #1,430,160,60,10 :PAPER #1,blue
15170  WINDOW #2,460,54,30,200 :PAPER #2,red : INK #2,white
15180  CLS#3:CLS#1:CLS#2
15190  BORDER #2,2,black
15200  SCALE #1,180,-180,-90
15210  print_header
15220  load_mercator_map
15230 END DEFine
```