

MODIFICADOR DE CADENAS

Muchos de nosotros, en alguna ocasión, nos hemos visto obligados a cambiar una cadena de caracteres que se repetía en varias ocasiones en un programa Basic. Aquellos que hayan vivido esta experiencia, conocerán lo monótona y trabajosa que resulta esta tarea.



A partir de ahora, la modificación de cadenas de un programa Basic, resultará lo más sencillo del mundo si nos ayudamos del programa que aparece listado en este artículo.

Debemos advertir en primer lugar que nuestro programa trabaja únicamente con programas que estén salvados en formato fichero.

Si deseamos trabajar con un programa que no se encuentre en este formato, deberemos hacer lo siguiente:

— Cargar el programa en memoria con:

`LOAD"PROGRAMA`

— Salvar el programa de la forma siguiente:

`SAVE"PROGRAMA",A`

de esta forma nuestra rutina ya podrá trabajar con él, puesto que ahora se encontrará salvado como fichero de caracteres o fichero ASCII.

La rutina encargada de efectuar el trabajo está ubicada a partir de la dirección hexadecimal &A000, y tiene una longitud de 1135 bytes. Por lo tanto no podremos trabajar con programas Basic que ocupen dichas direcciones.

Debemos aclarar que esto no es un gran inconveniente, debido a que apenas ningún programa escrito en Basic llega a ocupar esas direcciones, ya que generalmente no suelen ser demasiado extensos.

De lo dicho anteriormente, se deduce que el programa modificador de cadenas no podrá trabajar con programas Basic que superen los 39 K de longitud.

La solución del peor de los casos

Si en alguna ocasión necesitáramos trabajar con programas de gran extensión, que superarán los 39 K de memoria, no deberemos alarmarnos, puesto que todo tiene solución.

Dicha solución sería partir nuestro programa en dos, es decir, deberíamos cargar el programa en memoria y salvar únicamente la mitad, y a continuación volver a cargarlo en memoria y salvar la otra mitad.

Ahora estaríamos en condiciones de trabajar con cada una de las dos mitades sin problemas de memoria.

Una vez hubiésemos finalizado el trabajo cargaríamos el primer programa y a continuación haríamos un MERGE del segundo programa, con lo cual tendríamos el programa otra vez completo y modificado.

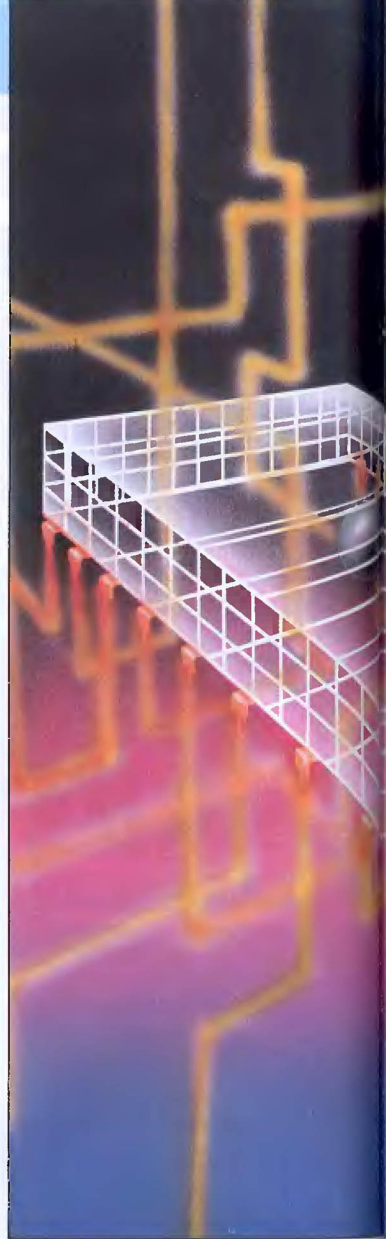
Realizadas estas advertencias y apuntadas las posibles soluciones a los problemas con los que nos podemos encontrar, pasaremos ahora a explicar el manejo del programa modificador de cadenas.

El programa

Con lo primero que nos encontramos al ejecutar el programa, es con un menú de ayuda en el que se encuentran las siguientes opciones:

ORDENES DEL MODIFICADOR

B — Basic
L — Listar
C — Cargar (LOAD)
S — Salvar (SAVE)
F — Cambiar cadenas



La primera de las opciones tiene como único objeto, permitirnos retornar al Basic.

La opción "**LISTAR**", nos permitirá listar el programa que se encuentre en memoria en ese momento. Si no hemos cargado ningún programa, al elegir esta opción, aparece un mensaje indicándonos que no existen ningún programa en memoria.

Cuando se elija la opción **CARGAR**, se nos pedirá el nombre del programa que deseamos cargar en memoria. Una vez escrito dicho nombre, deberemos pulsar "**ENTER**".

Mientras se esté ejecutando la carga del programa, aparecerán unas líneas en la pantalla, pero eso no nos debe asustar, ya que dichas rayas son debidas a que se ha elegido dicha zona de memoria como buffer de carga.

Se ha hecho de esta forma con el fin de ahorrar memoria ya que dicho buffer necesita: 2k.

A continuación se nos preguntará cuál es la cadena que debe sustituir a la anterior.

Cuando se hayan dado estos parámetros, el programa buscará una cadena idéntica a la primera que hayamos dado, y la cambiará por la última.

Cuando haya finalizado el trabajo, se imprimirá un mensaje indicando que dichas cadenas se han modificado. En el caso de que no encuentre ninguna cadena de caracteres igual a la que hemos introducido, nos lo indicará con otro mensaje.

Funcionamiento y rutinas principales

Vamos a ver ahora cuáles son las rutinas más importantes de que se compone nuestro programa.

Para hacernos una idea de conjunto del programa realizaremos una especie de diagrama de flujo con el cual podremos ver globalmente los pasos a seguir.

lectura del teclado. La rutina permanecerá en dicho bucle hasta que no sea pulsada alguna de las teclas predeterminadas.

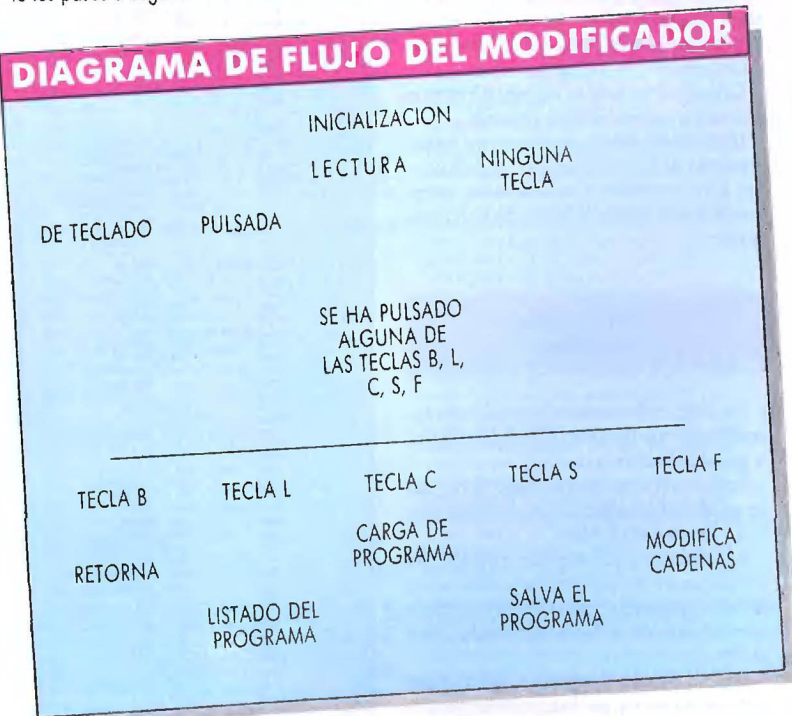
Cuando se detecte la pulsación de alguna de las teclas que indican al programa que se ha elegido una opción, se enviará el control a la rutina que corresponda, y una vez finalizado el trabajo se retornará al bucle principal de lectura de teclado.

La primera subrutina con la que nos encontramos en el programa, es la encargada de efectuar la carga.

Para averiguar cuál es el nombre del programa que se desea cargar, efectuamos una llamada a la rutina INPUT, la cual nos pedirá el nombre del fichero que deseamos introducir en la memoria del ordenador.

Hecho esto se retorna a la rutina LOAD para efectuar dicha carga, utilizando las rutinas del firmware capaces de leer un programa en forma de fichero ASCII.

Una vez finalizada su tarea, reinicializa la pantalla y devuelve el control del programa al bucle principal para detectar la elección de otras opciones.



Otra de las opciones que posee es la que permite salvar el programa que tenemos en memoria en disco o cinta. En esta opción volverán a aparecer las rayas mencionadas anteriormente, debido a que se utiliza el mismo buffer de memoria.

Por último, se nos ofrece la posibilidad de modificación de cadenas de caracteres. Esta opción nos permitirá cambiar una cadena por otra de una longitud no superior a 20 caracteres. Si la longitud de la cadena de caracteres que se desea modificar es superior a dicha cifra, entonces se deberá efectuar el cambio en dos pasadas, es decir, deberemos modificar en primer lugar los primeros 20 caracteres y a continuación los restantes. Cuando deseemos modificar alguna cadena, lo primero que hará el programa será preguntarnos cuál es la cadena que se desea cambiar; una vez introducida, deberemos pulsar "ENTER".

En primer lugar hacemos una inicialización de todos los parámetros que utilizaremos durante la ejecución del programa, seguidamente preparamos la pantalla eligiendo el modo en el que vamos a trabajar.

Activamos una ventana, en la cual imprimiremos el menú de opciones. En este caso se ha elegido una situada en la parte superior de la pantalla.

Una vez hecho esto, entramos en el bucle principal del programa, donde se produce la

Rutina de listar

Otra de las rutinas que componen el programa es la encargada de producir un listado en pantalla. Para ello se toma la dirección inicial del programa, y a partir de ahí va tomando los caracteres que encuentra en memoria y los imprime en pantalla.

Cuando se detecta que se ha llegado al final del programa, retorna al bucle principal.

A continuación nos encontramos con la rutina más importante de nuestro programa, se trata de la rutina encargada de modificar las cadenas de caracteres.

En primer lugar, se llama a la rutina INPUT, para conseguir la información de qué cadena debe modificar y por cuál ha de sustituirla. Cuando ha conseguido dichos datos, intenta detectar una cadena idéntica a la dada, sino lo consigue devuelve el control al bucle principal imprimiendo el mensaje de que no ha encontrado ninguna cadena.

En caso de que se encuentre con una cadena de caracteres idéntica a la que se ha dado, entonces se encarga de sustituirla por la nueva, reservando el espacio de memoria necesario para la nueva cadena de caracteres a introducir, y eliminando al mismo tiempo el espacio de memoria que ocupaba la antigua.

Una vez realizada la misma operación con todas las cadenas de caracteres idénticas a la dada, imprime el mensaje correspondiente y retorna al bucle de lectura de teclado.

Por último nos queda comentar la rutina de SAVE, ésta hace una llamada a INPUT para saber qué nombre debe dar al nuevo fichero, y a continuación se almacena en cinta o disco.

Después de finalizar su trabajo, inicializa la pantalla y retorna al bucle principal.

Una vez realizada la descripción del funcionamiento de la rutina y dadas las instrucciones para su correcto funcionamiento, únicamente queda copiar el listado de dicho programa.

Manipulación del listado

Para ello, deberemos copiar el listado ensamblador que aparece al final del artículo, y salvarlo en cinta o disco.

También ofrecemos un cargador Basic, para aquellos que prefieran teclearlo directamente en forma de DATAS.

Aquellos que elijan esta última opción, deberán ejecutar el programa cargador una vez teclado, y en el caso de que aparezca algún mensaje de error, se deberán revisar las líneas DATA.

Si dicho mensaje no aparece, indicará que todo ha ido bien, y por lo tanto estaremos en condiciones de salvarlo en cinta o disco.

Para ello haremos lo siguiente:

SAVE "CADENAS", B, &A000, 1135

Cuando se quiera trabajar con el modificador de cadenas, deberemos escribir un programa BASIC como el que se indica a continuación:

```
10 MEMORY &9FF
20 LOAD "CADENAS"), &A000
30 CALL &A000
```

No queda nada más que decir, sólo desear que este programa os sea de gran utilidad.

```

10 MODIFICADOR DE CADENAS
20 ORG &A000
30 LD HL, 0
40 LD (FINPRO), HL
50 CALL VENTAN
60 JP TEC
70 VENTAN: LD A, 2
80 CALL &BC0E
90 LD A, 1
100 CALL &BB84
110 LD HL, 0
120 LD DE, &5001
130 CALL &BB44
140 LD A, 1
150 CALL &BB9A
160 XOR A
170 CALL &BB90
180 CALL &BB4C
190 LD HL, TXT
200 DTRPD: LD A, (HL)
210 CP 255
220 JR Z, FUERA
230 CALL &BB5A
240 INC HL
250 JR DTRPD
260 FUERA: XOR A
270 CALL &BB84
280 LD HL, &A002
290 LD DE, &5019
300 CALL &BB44
310 RET
320 I
330 TEC: LD A, 54
340 CALL &BB1E
350 RET NZ
360 LD A, 34
370 CALL &BB1E
380 CALL &BB1E
390 JR Z, PAST1
400 CALL IMPRE
410 PAST1: LD A, 60
420 CALL &BB1E
430 JR Z, PAST2
440 CALL &BB1E
450 PAST2: LD A, 62
460 CALL &BB1E
470 JR Z, PAST3
480 CALL LOAD
490 PAST3: LD A, 53
500 CALL &BB1E
510 JR Z, TEC
520 CALL BUSCA
530 JR TEC
540 I
550 LOAD: CALL &BB4C
560 CALL I_LOAD
570 CALL &BB4C
580 LD A, (L_LOAD)
590 LD B, A
600 LD HL, N_LOAD
610 LD DE, &C0A0
620 CALL &BC77
630 JP NC, ERROR
640 LD HL, 5000
650 BULEC: CALL &BC90
660 JR NC, FINIT
670 LD (HL), A
680 INC HL
690 JR BULEC
700 JR BULEC
710 FINIT: DEC H
720 LD (FINPRO), HL
730 CALL &BC7A
740 CALL VENTAN
750 LD HL, &B0B0
760 LD DE, TXTPL
770 CALL PRINT
780 RET
790 IMPRE: CALL &BB4C
800 LD HL, (FINPRO)
810 LD A, H
820 OR L
830 JP Z, NOPRO
840 LD DE, 5000
850 DTRA: PUSH DE
860 LD A, 64
870 CALL &BB1E
880 CALL NZ, PAUSA
890 POP DE
900 LD A, (DE)
910 CALL &BB5A
920 INC DE
930 LD HL, (FINPRO)
940 LD A, H
950 CP B
960 JR NZ, QTRA
970 LD A, L
980 CP E
990 JR NZ, QTRA
1000 RET
1010 PAUSA: LD BC, 4000
1020 PAUS: DEC BC
1030 LD A, B
1040 OR C
1050 JR NZ, PAUS
1060 CALL &BB03
1070 CALL &BB18
1080 RET
1090 I
1100 BUSCA CADENA
1110 I
1120 BUSCA: LD HL, (FINPRO)
1130 LD A, H
1140 OR L
1150 JP Z, NOPRO
1160 XOR A
1170 LD (NUMCA), A
1180 CALL &BB4C
1190 CALL NOMCA1
1200 CALL &BB4C
1210 CALL NOMCAN
1220 CALL &BB4C
1230 LD HL, CADENV
1240 LD DE, 5000
1250 LD IX, 1000
1260 VULV: LD B, 0
1270 BUC: LD A, (LONCV)
1280 CP B
1290 JR Z, FIN
1300 PUSH HL
1310 LD HL, (FINPRO)
1320 LD A, D
1330 CP H
1340 JR NZ, SIGU
1350 LD A, E
1360 CP L
1370 JR Z, FINRU
1380 POP HL
1390 SIGU: LD A, (DE)
1400 LD (IX+0), A
1410 CP (HL)
1420 INC DE
1430 JR Z, INCCO
1440 LD B, 0
1450 LD B, 0
1460 LD HL, CADENV
1470 JR BUC
1480 INCCO: INC B
1490 INC HL
1500 JR BUC
1510 FIN: LD (DIREC), DE
1520 DTRDE: DEC IX
1530 DJNZ DTRDE
1540 LD A, (NUMCA)
1550 INC A
1560 LD (NUMCA), A
1570 LD A, (LONCN), A
1580 LD B, A
1590 LD HL, CADENN
1600 LD A, (HL)
1610 LD (IX+0), A
1620 INC IX
1630 INC HL
1640 DJNZ PONBU
1650 LD HL, CADENV
1660 JR VULV
1670 FINRU: POP HL
1680 PUSH IX
1690 POP HL
1700 LD DE, 4000
1710 ADD HL, DE
1720 LD (FINPRO), H
1730 LD DE, 5000
1740 SCF
1750 EBC
1760 CDF HL, DE
1770 LD B, H
1780 LD C, L
1790 LD HL, 1000
1800 LD DE, 5000
1810 LD IR
1820 LD A, (NUMCA)
1830 OR C
1840 JR NZ, SIGCA
1850 LD HL, &B0B0
1860 LD DE, TXTNOC
1870 CALL PRINT
1880 RET
1890 SIGCA: LD HL, &B0B0
1900 LD DE, TXTSIC
1910 CALL PRINT
1920 RET
1930 SAVE: CALL &BB4C
1940 LD HL, (FINPRO)
1950 LD A, H
1960 OR L
1970 JP Z, NOPRO
1980 CALL I_SAVE
1990 CALL &BB4C
2000 LD A, (L_SAVE)
2010 LD B, A
2020 LD HL, N_SAVE
2030 LD DE, &C0A0
2040 CALL &BB4C
2050 LD DE, 5000
2060 DTRCA: LD A, (DE)
2070 CALL &BC95
2080 LD HL, (FINPRO)
2090 LD A, H
2100 CP D
2110 JR NZ, PAST1
2120 LD A, L
2130 CP E
2140 JR Z, FINSA
2150 INC DE
2160 JR DTRA
2170 FINSA: CALL &BC9F
2180 CALL &BC9F
2190 CALL VENTAN
2200 RET
2210 NOMCA1: LD A, 20
2220 LD (MIRLON+1), A
2230 LD HL, &B0B0
2240 LD DE, TXT0
2250 CALL PRINT
2260 CALL INPUT
2270 LD A, (LONG)
2280 LD (LONCV), A
2290 LD DE, CADENV
2300 CALL TRASP
2310 RET
2320 NOMCAN: LD A, 20
2330 LD (MIRLON+1), A
2340 LD HL, &B0B0
2350 LD DE, TXT1
2360 CALL PRINT
2370 CALL INPUT
2380 LD A, (LONG)
2390 LD (LONCN), A
2400 LD DE, CADENN
2410 CALL TRASP
2420 RET
2430 I_LOAD: LD A, B
2440 LD (MIRLON+1), A
2450 LD HL, &B0B0
2460 LD DE, TXT2
2470 CALL PRINT
2480 CALL INPUT
2490 LD A, (LONG)
2500 LD (L_LOAD), A
2510 LD DE, N_LOAD
2520 CALL TRASP
2530 RET
2540 I_SAVE: LD A, B
2550 LD (MIRLON+1), A
2560 LD HL, &B0B0
```



```

2570 LD DE,TEXT3
2580 CALL PRINT
2590 CALL INPUT
2600 LD A,(LONG)
2610 LD (L_SAVE),A
2620 LD DE,N_SAVE
2630 CALL TRASP
2640 RET
2650 INPUT: CALL #B803
2660 CP 1
2670 LD (LONG),A
2680 LD DE,NAME
2690 LD HL,#090F
2700 CALL #B875
2710 LD A," "
2720 CALL #B85A
2730 L_BUC: CALL #B818
2740 LD B,A
2750 CP 13
2760 RET 2
2770 CP 127
2780 JR Z,DELETE
2790 CP 32
2800 JR C,I_BUC
2810 CP 129
2820 JR NC,I_BUC
2830 LD A,(LONG)
2840 MIRONL: CP 4
2850 JR Z,I_BUC
2860 LD A,B
2870 CALL #B85A
2880 LD A,B
2890 CALL #B85A
2900 LD A," "
2910 CALL #B85A
2920 LD A,(LONG)
2930 INC A
2940 LD (LONG),A
2950 LD A,B
2960 LD (DE),A
2970 INC DE
2980 JR I_BUC
2990 DELET: LD A,(LONG)
3000 CP 0
3010 JR Z,I_BUC
3020 DEC A
3030 LD (LONG),A
3040 DEC DE
3050 LD A,32
3060 LD (DE),A
3070 LD A,B
3080 CALL #B85A
3090 LD A,32
3100 CALL #B85A
3110 LD A,B
3120 CALL #B85A
3130 LD A,B
3140 CALL #B85A
3150 LD A,32
3160 CALL #B85A
3170 LD A,B
3180 CALL #B85A
3190 LD A," "
3200 CALL #B85A
3210 JR I_BUC
3220 TRASP: LD A,(LONG)
3230 LD C,A
3240 LD B,0
3250 LD HL,NAME
3260 LDIR
3270 RET
3280 PRINT: CALL #B875
3290 PBUD: LD A,(DE)
3300 CP 255
3310 RET
3320 CALL #B85A
3330 INC DE
3340 JR PBUD
3350 RET
3360 NOPRD: LD HL,#B0B0
3370 LD DE,TEXTPR
3380 CALL PRINT
3390 RET
3400 ERROR: LD HL,#B0B0
3410 LD DE,TEXTR
3420 CALL PRINT
3430 RET
3440 TXT0: DEFM "NOMBRE DE LA CADENA A CAMBIAR"
3450 DEFB 255
3460 TXT1: DEFM "NOMBRE DE LA NUEVA CADENA"
3470 DEFB 255
3480 TXT2: DEFM "NOMBRE DEL PROGRAMA A CARGAR"
3490 DEFB 255
3500 TXT3: DEFM "NOMBRE DEL PROGRAMA A SALVAR"
3510 DEFB 255
3520 N_LOAD: DEFS 10
3530 L_LOAD: DEFS 1
3540 N_SAVE: DEFS 10
3550 L_SAVE: DEFS 1
3560 CADENA: DEFS 20
3570 LONCV: DEFS 1
3580 CADENA: DEFS 20
3590 LONCV: DEFS 1
3600 TXTPR: DEFM "NO HAY PROGRAMA EN MEMORIA"
3610 DEFB 255
3620 TXTPL: DEFM "PROGRAMA EN MEMORIA"
3630 DEFB 255
3640 TXTNOC: DEFM "NO EXISTE ESTA CADENA"
3650 DEFB 255
3660 TXTSIC: DEFM "CADENAS CAMBIADAS"
3670 DEFB 255
3680 TXTER: DEFM "ERROR"
3690 DEFB 255
3700 TXT: DEFM " B - BASIC L - LISTAR
C - CARGAR DEFM "S - SALVAR F - CAMBIAR CADENAS"
3710 DEFB 255
3730 FINPRO: DEFW 0
3740 NUMCA: DEFB 0
3750 DIREC: DEFS 2
3760 LONG: DEFS 1
3770 NAME: DEFS 25
00
LEC A093 BUSCA A0EB
DENV A37A DELET A790
ROR A2EB FIN #137
NPRO A44C FINRU /A158

```

```

10 REM MODIFICADOR DE CADENAS
20 REM PROGRAMA CARGADOR
30 FOR N=#A000 TO #A46E
40 READ A:SUMA=SUMA+A
50 POKE N,A
60 NEXT
70 IF SUMA<109212 THEN PRINT "ERRO
REN DATAS"
80 DATA 33,0,0,34,76,164,205
90 DATA 12,160,195,71,160,62,2
100 DATA 205,14,188,62,1,205,180
110 DATA 187,33,0,0,17,1,80
120 DATA 205,102,187,62,1,205,150
130 DATA 187,175,205,144,187,205,10
8
140 DATA 187,33,1,164,126,254,255
150 DATA 40,6,205,90,187,35,24
160 DATA 245,175,205,180,187,33,2
170 DATA 0,17,25,80,205,102,187
180 DATA 201,62,54,205,30,187,192
190 DATA 62,36,205,30,187,40,3
200 DATA 205,176,160,62,60,205,30
210 DATA 187,40,3,205,146,161,62
220 DATA 62,205,30,187,40,3,205
230 DATA 119,160,62,53,205,30,187
240 DATA 40,213,205,232,160,24,208
250 DATA 205,108,187,205,11,162,205
260 DATA 108,187,58,110,163,71,33
270 DATA 100,163,17,160,192,205,119
280 DATA 188,210,232,162,33,136,19
290 DATA 205,128,188,48,4,119,35
300 DATA 24,247,43,34,76,164,205
310 DATA 122,188,205,12,160,33,13
320 DATA 8,17,191,163,205,208,162
330 DATA 201,205,108,187,42,76,164
340 DATA 124,181,202,222,162,17,136
350 DATA 19,213,62,66,205,30,187
360 DATA 196,217,160,209,26,205,90
370 DATA 187,19,42,76,164,124,186
380 DATA 32,234,125,187,32,230,201
390 DATA 1,64,156,11,120,177,32
400 DATA 251,205,3,187,205,24,187
410 DATA 201,42,76,164,124,181,202
420 DATA 222,162,175,50,78,164,205
430 DATA 108,187,205,207,161,205,10
8
440 DATA 187,205,237,161,205,108,18
7
450 DATA 33,122,163,17,136,19,221
460 DATA 33,232,3,6,0,58,142
470 DATA 163,184,40,34,229,42,76
480 DATA 164,122,188,32,4,123,189
490 DATA 40,58,225,26,221,119,0
500 DATA 190,19,221,35,40,7,6
510 DATA 0,33,122,163,24,220,4
520 DATA 35,24,216,237,83,79,164
530 DATA 221,43,16,252,58,78,164
540 DATA 60,50,78,164,58,163,163
550 DATA 71,33,143,163,126,221,119
560 DATA 0,221,35,35,16,247,33
570 DATA 122,163,24,178,225,221,229
580 DATA 225,17,160,15,25,34,76
590 DATA 164,17,136,19,55,63,237
600 DATA 82,68,77,33,232,3,17
610 DATA 136,19,237,176,58,78,164
620 DATA 254,0,32,10,33,13,8
630 DATA 17,211,163,205,208,162,201
640 DATA 33,13,8,17,233,163,205
650 DATA 208,162,201,205,108,187,42
660 DATA 76,164,124,181,202,222,162
670 DATA 205,41,162,205,108,187,58
680 DATA 121,163,71,33,111,163,17
690 DATA 160,192,205,140,188,17,136
700 DATA 19,26,205,149,188,42,76
710 DATA 164,124,186,32,4,125,187
720 DATA 40,3,19,24,238,205,143
730 DATA 188,205,146,188,205,12,160
740 DATA 201,62,20,50,115,162,33
750 DATA 13,8,17,242,162,205,208
760 DATA 162,205,71,162,58,81,164
770 DATA 50,142,163,17,122,163,205
780 DATA 196,162,201,62,20,50,115
790 DATA 162,33,13,8,17,16,163
800 DATA 205,208,162,205,71,162,58
810 DATA 81,164,50,163,163,17,143
820 DATA 163,205,196,162,201,62,8
830 DATA 50,115,162,33,13,8,17
840 DATA 42,163,205,208,162,205,71
850 DATA 162,58,81,164,50,110,163
860 DATA 17,100,163,205,196,162,201
870 DATA 62,8,50,115,162,33,13
880 DATA 8,17,71,163,205,208,162
890 DATA 205,71,162,58,81,164,50
900 DATA 121,163,17,111,163,205,196

```

```

910 DATA 162,201,205,3,187,175,50
920 DATA 81,164,17,82,164,33,15
930 DATA 9,205,117,187,62,95,205
940 DATA 90,187,205,24,187,71,254
950 DATA 13,200,254,127,40,41,254
960 DATA 32,56,241,254,129,48,237
970 DATA 58,81,164,254,4,40,230
980 DATA 62,8,205,90,187,120,205
990 DATA 90,187,62,95,205,90,187
1000 DATA 58,81,164,60,50,81,164
1010 DATA 120,18,19,24,204,58,81
1020 DATA 164,254,0,40,197,61,50
1030 DATA 81,164,27,62,32,18,62
1040 DATA 8,205,90,187,62,32,205
1050 DATA 90,187,62,8,205,90,187
1060 DATA 62,8,205,90,187,62,32
1070 DATA 205,90,187,62,8,205,90
1080 DATA 187,62,95,205,90,187,24
1090 DATA 152,58,81,164,79,6,0
1100 DATA 33,82,164,237,176,201,205
1110 DATA 117,187,26,254,255,200,20
5
1120 DATA 90,187,19,24,246,201,33
1130 DATA 13,8,17,164,163,205,208
1140 DATA 162,201,33,13,8,17,251
1150 DATA 163,205,208,162,201,78,79
1160 DATA 77,66,82,69,32,68,69
1170 DATA 32,76,65,32,67,65,68
1180 DATA 69,78,65,32,65,32,67
1190 DATA 65,77,66,73,65,82,255
1200 DATA 78,79,77,66,82,69,32
1210 DATA 68,69,32,76,65,32,78
1220 DATA 85,69,86,65,32,67,65
1230 DATA 68,69,78,65,255,78,79
1240 DATA 77,66,82,69,32,68,69
1250 DATA 76,32,80,82,79,71,82
1260 DATA 65,77,65,32,65,32,67
1270 DATA 65,82,71,65,82,255,78
1280 DATA 79,77,66,82,69,32,68
1290 DATA 69,76,32,80,82,79,71
1300 DATA 82,65,77,65,32,65,32
1310 DATA 83,65,76,86,65,82,255
1320 DATA 0,0,0,0,0,0,0,0
1330 DATA 0,0,0,0,0,0,0,0
1340 DATA 0,0,0,0,0,0,0,0
1350 DATA 0,0,0,0,0,0,0,0
1360 DATA 0,0,0,0,0,0,0,0
1370 DATA 0,0,0,0,0,0,0,0
1380 DATA 0,0,0,0,0,0,0,0
1390 DATA 0,0,0,0,0,0,0,0
1400 DATA 0,0,0,0,0,0,0,0
1410 DATA 0,78,79,32,72,65,89
1420 DATA 32,80,82,79,71,82,65
1430 DATA 77,65,32,69,78,32,77
1440 DATA 69,77,79,82,73,65,255
1450 DATA 80,82,79,71,82,65,77
1460 DATA 65,32,69,78,32,77,69
1470 DATA 77,79,82,73,65,255,78
1480 DATA 79,32,69,88,73,83,84
1490 DATA 69,32,69,83,84,65,32
1500 DATA 67,65,68,69,78,65,255
1510 DATA 67,65,68,69,78,65,83
1520 DATA 32,67,65,77,66,73,65
1530 DATA 68,65,83,255,69,82,82
1540 DATA 79,82,255,32,32,32,32
1550 DATA 66,32,45,32,66,65,83
1560 DATA 73,67,32,32,32,76,32
1570 DATA 45,32,76,73,83,84,65
1580 DATA 82,32,32,32,67,32,45
1590 DATA 32,67,65,82,71,65,82
1600 DATA 32,32,32,83,32,45,32
1610 DATA 83,65,76,86,65,82,32
1620 DATA 32,32,70,32,45,32,67
1630 DATA 65,77,66,73,65,82,32
1640 DATA 67,65,68,69,78,65,83
1650 DATA 255,0,0,0,0,0,0,0
1660 DATA 0,0,0,0,0,0,0,0
1670 DATA 0,0,0,0,0,0,0,0
1680 DATA 0,0,0,0,0,0,0,0
1690 DATA 0,0,0,0,0,0,0,0

```

