

Listing 1

```

100 ' *****
105 ' *   reloc   LOADER   *
110 ' *                                     *
115 ' * (C) by   IB & SOFT *
120 ' *****
125 '
130 ON ERROR GOTO 200
135 MEMORY 41999
140 '
145 start=HIMEM-8
150 check=0
155   WHILE check=0: start=start+8
160     ch=0
165     FOR w=1 TO 8
170       READ a$:a=VAL("&" + a$)
175       ch=ch+a:POKE start+w,a
180     NEXT w:READ ch$
185     check=ch XOR VAL("&" + ch$)
190   WEND
195 PRINT:PRINT"Data error in line ";
    (start-41999)/8*5+215:END
200 '
205 IF ERR<>4 THEN PRINT"Error in line "
    ;ERR:STOP
210 RESUME 295
215 DATA 00,00,00,00,00,00,66,69,0CF
220 DATA 6C,65,20,2E,62,69,6E,3E,296
225 DATA 30,32,1A,A4,06,09,21,16,166
230 DATA A4,1',10,27,CD,77,BC,21,30D
235 DATA 10,27,22,14,A4,CD,83,BC,31D
240 DATA CD,7A,BC,3E,31,32,1A,A4,362
245 DATA 06,09,21,16,A4,11,54,0B,15A
250 DATA CD,77,BC,21,10,27,22,12,28C
255 DATA A4,CD,80,BC,30,12,47,2A,360
260 DATA 14,A4,7E,B8,C4,6C,A4,2A,3EC
265 DATA 14,A4,23,22,14,A4,18,E9,2B6
270 DATA CD,7A,BC,C9,A7,2A,14,A4,455
275 DATA 11,10,27,ED,52,EB,2A,12,2AE
280 DATA A4,2B,72,2B,73,22,12,A4,2B7
285 DATA 2A,14,A4,23,22,14,A4,CD,2AC
290 DATA 80,BC,C9,00,00,00,00,00,205
295 '
296 STOP
300 SAVE"reiko.bin",b,42000,123

```

Listing 2

```

100 '
110 '   Z80  RELOCATOR
120 '
130 '   (c) by IB & SOFT
140 '   MAY 1987
150 '
160 MODE 2:INK 0,0:BORDER 0:INK 1,0:FOR i=1 TO 9:PRINT STRING$(79,207):NEXT
170 LOCATE 35,3:PRINT CHR$(22)+CHR$(1):TAB(33)"Z80  RELOCATOR"TAB(32):CHR$(10)
   "(C) by IB & SOFT":CHR$(22):CHR$(0)
180 INK 1,23
190 '
200 WINDOW=3,1,80,10,14:PEH=3,1:PAPER=3,0
210 WINDOW=0,1,79,16,25:PEH=0,0:PAPER=0,1:CLS
220 '
230 DEF FN deek(a)=PEEK(a)+256*PEEK(a+1)
240 DEF FN low(a)=a-INT(a/256)*256
250 DEF FN high(a)=INT(a/256)
260 '
270 LOCATE*3,31,2:PRINT*3,"Loading files ...";
280 MEMORY 41999:LOAD"reiko.bin",42000
290 CLS*3:LOCATE*3,31,2:PRINT*3,"Analysing file !";
300 CALL 42015:CLS*3
310 '
320 CLS:LOCATE 2,2:INPUT"Special relocating adress (relative to start) : ",a$:
   adress=VAL(a$)
330 IF adress<>0 THEN CLS:adr=FN deek(42002):adr=adr-2:POKE adr,FN low(adress):
   POKE adr+1,FN high(adress):POKE 42002,FN low(adr):POKE 42003,FN high(adr):
   GOTO 320
340 '
350 CLS:LOCATE 2,2:INPUT"Initial routine (No or relative to start) : ",a$:
   adress=VAL(a$):IF UPPER$(a$)="H" THEN adress=65535
360 adr=FN deek(42002):adr=adr-2:POKE adr,FN low(adress):
   POKE adr+1,FN high(adress):POKE 42002,FN low(adr):POKE 42003,FN high(adr)
370 '
380 lenght=(9998-FN deek(42002))/2
390 adr=FN deek(42002):adr=adr-2:POKE 42002,FN low(adr):
   POKE 42003,FN high(adr):POKE adr,FN low(lenght):POKE adr+1,FN high(lenght)
400 '
410 CLS:LOCATE 2,2:PRINT STRING$(76,154)
420 PRINT TAB(20)"Program lenght"STRING$(60-POS(*0),46):FN deek(42004)-10000
430 PRINT TAB(20)"Table lenght"STRING$(60-POS(*0),46):10000-FN deek(42002)
440 PRINT TAB(20)"Total lenght"STRING$(60-POS(*0),46):FN deek(42004)-adr+47
   " + reloc"TAB(20)"Initial routine"STRING$(60-POS(*0),46):
450 a=FN deek(FN deek(42002)+2):IF a=65535 THEN PRINT " No" ELSE PRINT a
460 PRINT TAB(2)STRING$(76,154)
470 '
480 DATA 21,10,27,01,CF,00,C5,E5
490 DATA 21,70,25,4E,23,46,23,D1
500 DATA E5,EB,E5,A7,09,4E,23,46
510 DATA 2B,D1,E5,D5,EB,09,EB,C1
520 DATA E1,73,23,72,E1,D1,1B,7A
530 DATA B3,A7,C8,D5,C5,1B,DC,00
540 FOR i=adr-47 TO adr-1:READ a$:POKE i,VAL("&"a$):NEXT
550 POKE adr-43,FN low(lenght):POKE adr-42,FN high(lenght)
555 PRINT:INPUT" Name for file : ",a$:SAVE a$,b,adr-47,FN deek(42004)-adr+47
560 PEH 1:PAPER 0:MODE 2

```

Listing 3

```
100 '
110 ' Auto relocater
120 '
121 c=HIMEM
130 MODE 2:INPUT"Load adress : ",a:
    INPUT"Lenght of program : ",l
140 INPUT"Name of file : ",b$
141 IF a=0 THEN a=c-l
142 MEMORY a-l:OPENOUT"dummy":LOAD b$,a
150 '
160 st=a+51+(PEEK(a+4)+256*PEEK(a+5))*2
170 POKE a+1,st-INT(st/256)*256:
    POKE a+2,INT(st/256)'start programa
180 b=a+51:POKE a+9,b-INT(b/256)*256:
    POKE a+10,INT(b/256)'start tabele
190 CALL a
200 b=PEEK(a+49)+256*PEEK(a+50):
    IF b<>65535 THEN CALL st+b
205 CLOSEOUT:CLOSEIN
206 MEMORY st-l
```

```

10 *OPENOUT "REN.bin"
20 #LIST ON
30 ;
;*****
; Relocator
;*****
;
40 relko: LD HL,10000 ; start programa - sem se nalozi program
50 LD BC,0 ; koliko naslovov
PUSH BC
PUSH HL
LD HL,0 ; start tabele
; zgornje naslove-byte popravlja BASIC
;
60 ;
loop : LD C,(HL)
INC HL
LD B,(HL) ; BC=odmik od zacetka programa
INC HL
70 ;
POP DE
PUSH HL ; naslov tabele na stacku
EX DE,HL ; start naslov
PUSH HL ; start naslov na stack
AND A
ADD HL,BC ; start programa + odmik
80 ;
LD C,(HL)
INC HL
LD B,(HL) ; BC=relocativni byte
DEC HL
;
POP DE
PUSH HL ; naslov relokativnega bytea
PUSH DE ; start programa
EX DE,HL ; HL=start naslov
90 ;
ADD HL,BC ; HL=pravilni (POPRAVLJEN) naslov
EX DE,HL
POP BC ; start programa
POP HL
LD (HL),E
INC HL
LD (HL),D ; pravi naslov shranjen
100 ;
POP HL ; naslednji naslov tabele
POP DE ; koliko naslov ?
DEC DE
LD A,D
OR E
AND A ; CP 0 (je zadnji ?)
RET Z
;
110 PUSH DE ; koliko jih je se ostalo !
PUSH BC ; start naslov,
; HL=naslednji naslov tabele
JR loop
;
120 *CLOSEOUT

```