Listing 1

```
1           ORG   #4000
2
3
4  START:   CP    2                ;ako uz CALL ne stoje dva argumenta
5           RET   NZ               ;vrati se u bejzik
6           LD    E,(IX+0)         ;DE=završna
7           LD    D,(IX+1)         ;adresa
8           LD    L,(IX+2)         ;HL=početna
9           LD    H,(IX+3)         ;adresa
10          LD    (ZAVR+1),DE      ;zavr. adresu na mjesto za testiranje
11          PUSH  HL               ;IX=HL tj. IX sadrži početnu adresu
12          POP   IX               ;djela memorije koju treba pretvoriti
13          SBC   HL,DE            ;ako je početna adresa veća od završne
14          RET   NC               ;vrati se u bejzik
15          LD    HL,#170          ;HL=početna adresa bejzika
16
17 LINIJA:  LD    A,(HL)           ;uzmi dužinu tekuće bejzik DATA linije
18          OR    A                ;ako je dužina nula,tada više nema
19          RET   Z                ;bejzik linija, pa se vrati u bejzik
20          SUB   10               ;izračunaj broj bajtova#3 u toj liniji
21          LD    B,A              ;u tu DATA liniju ide B/3 bajtova
22          LD    DE,6             ;preskoči dužinu i broj bejzik linije,
23          ADD   HL,DE            ;token za DATA (&8c) i space bajt (&20)
24          LD    E,D              ;postavi brojač sume (DE) na nulu
25
26 XPUTA:   LD    A,(IX)           ;uzmi bajt kojeg treba pretvoriti
27          LD    C,A              ;i privremeno ga smjesti u C
28          ADD   A,E              ;dodaj prethodnom
29          LD    E,A              ;stanju brojača sume
30          JR    NC,NOTINC        ;vrijednost upravo
31          INC   D                ;očitanog bajta
32 NOTINC:  LD    A,C              ;vrati vrijednost očitanog bajta u A
33          CALL  BYTASC           ;i pretvori ga u ASCII format
34          INC   HL               ;preskoči zarez u bejzik DATA liniji
35          INC   IX               ;pointer na slijedeći bajt
36          DEC   B                ;čitav ovaj postupak ponovi B/3 puta,
37          DEC   B                ;tj. onoliko puta koliko ima mjesta
38          DJNZ  XPUTA            ;u DATA liniji koju popunjavaš
39
40          LD    A,D              ;uzmi viši bajt sume
41          CALL  LVDS             ;i pretvori mu niža 4 bita u ASCII
42          LD    A,E              ;uzmi niži bajt sume
43          CALL  BYTASC           ;i pretvori ga u ASCII
44          INC   HL               ;preskoči linijski terminator (0)
45
46          EX    DE,HL            ;HL sačuvaj u DE
47          PUSH  IX               ;prebaci sadržaj IX registra
48          POP   HL               ;u HL registar
49 ZAVR:    LD    BC,0             ;BC sadrži završnu adresu pretvorbe
50          SBC   HL,BC            ;provjeri da li je ona premašena
51          EX    DE,HL            ;vrati vrijednost HL registra
52          JR    C,LINIJA         ;ako završna adresa nije premašena
53          RET                    ;nastavi pretvorbu, inače u bejzik
54
55 BYTASC:  LD    C,A              ;privremeno sačuvaj A u C
56          RLCA                   ;u A registru
57          RLCA                   ;zamjeni mjesta
58          RLCA                   ;bitovima 0-3
59          RLCA                   ;sa bitovima 4-7
60          CALL  LVDS             ;pretvori u ASCII viša 4 bita
61          LD    A,C              ;isto to uradi i za 4 niža bita
62
63 LVDS:    AND   #0F              ;uzmi u obzir samo bitove 0-3
64          CP    #0A              ;da li je broj ili slovo
65          JR    C,NIJSLO         ;ako nije slovo ne dodaji 39
66          ADD   A,#27            ;ako želite velika slova dodajte #07
67 NIJSLO:  ADD   A,"0"            ;dodaj ASCII kod nule
68          LD    (HL),A           ;i upiši u DATA liniju
69          INC   HL               ;povećaj pointer DATA linije
70          RET                    ;vrati se iz potprograma
```

Listing 2

```
>A
Table size:
Options: 4
Hisoft GENA3.1 Assembler. Page      1.

Pass 1 errors: 00


Pass 2 errors: 00

Table used:    111  from    399

>
>B
Ready
new
Ready
list
Ready
auto 100,10
100 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
110 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
120 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
130 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
140 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
150 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
160 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
170 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
180 data **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
190 *Break*
Ready
call &4000,&4000,&4066
Ready
list
100 DATA fe,02,c0,dd,5e,00,dd,56,01,dd,6e,02,dd,66,03,ed,7af
110 DATA 53,49,40,e5,dd,e1,ed,52,d0,21,70,01,7e,b7,c8,d6,8f3
120 DATA 0a,47,11,06,00,19,5a,dd,7e,00,4f,83,5f,30,01,14,3ac
130 DATA 79,cd,51,40,23,dd,23,05,05,10,ec,7a,cd,5a,40,7b,65c
140 DATA cd,51,40,23,eb,dd,e5,e1,01,66,40,ed,42,eb,38,cc,8d4
150 DATA c9,4f,07,07,07,07,cd,5a,40,79,e6,0f,fe,0a,38,02,54b
160 DATA c6,27,c6,30,77,23,c9,00,00,00,00,00,00,00,00,00,346
170 DATA **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
180 DATA **,**,**,**,**,**,**,**,**,**,**,**,**,**,**,**,***
Ready
170
180
10 lin=90:for i=&4000 to &4066 step 16
20 lin=lin+10:sum=0:for j=i to i+15
30 read a$:a=val("&"+a$)
40 poke j,a:sum=sum+a:next:read sum$
50 if sum=val("&"+sum$) then next:end
60 print "Greška u liniji";lin:end
70 '
80 '
90 '
list
10 lin=90:FOR i=&4000 TO &4066 STEP 16
20 lin=lin+10:sum=0:FOR j=i TO i+15
30 READ a$:a=VAL("&"+a$)
40 POKE j,a:sum=sum+a:NEXT:READ sum$
50 IF sum=VAL("&"+sum$) THEN NEXT:END
60 PRINT "Greška u liniji";lin:END
70 '
80 '
90 '
100 DATA fe,02,c0,dd,5e,00,dd,56,01,dd,6e,02,dd,66,03,ed,7af
110 DATA 53,49,40,e5,dd,e1,ed,52,d0,21,70,01,7e,b7,c8,d6,8f3
120 DATA 0a,47,11,06,00,19,5a,dd,7e,00,4f,83,5f,30,01,14,3ac
130 DATA 79,cd,51,40,23,dd,23,05,05,10,ec,7a,cd,5a,40,7b,65c
140 DATA cd,51,40,23,eb,dd,e5,e1,01,66,40,ed,42,eb,38,cc,8d4
150 DATA c9,4f,07,07,07,07,cd,5a,40,79,e6,0f,fe,0a,38,02,54b
160 DATA c6,27,c6,30,77,23,c9,00,00,00,00,00,00,00,00,00,346
Ready
run
Ready
```