

in **RETURN** znake (koda 13), pa sprejme brez problema. Če ga uporabimo z # 3 (ekran), bo GET# stavek prebral znake z ekrana enega za drugim. Vsaka uporaba GET# stavka pomakne kursor za eno mesto na desno. Znak na koncu logične vrstice pa se zamenja s CHR\$(13) (tipka **RETURN**).

Primeri: 5 GET# 1, A\$
10 OPEN 1,3: GET# 1, Z7\$
20 GET# 1, A, B, C\$, D\$

GOSUB

Tip: Stavek

Format: GOSUB <številka vrstice>

Delovanje: To je specifična oblika GOTO stavka, ki pa se od njega razlikuje po tem, da si zapomni, s katerega mesta v programu je bil narejen skok. Ko v programu pridemo do RETURN stavka (to ni **RETURN** tipka!!), se program vrne na stavek, ki sledi GOSUB stavku, s katerim smo skočili v podprogram. Podprogram napišemo takrat, kadar nek del programa uporabimo večkrat v različnih delih programa. Tako si lahko prihranimo mnogo prostora. V tem pogledu je GOSUB stavek podoben DEF FN.

Primeri: Najprej pogledjmo program, ki ne uporablja GOSUB:

```
100 PRINT "TA PROGRAM PIŠE"  
110 FOR L = 1 TO 500: NEXT  
120 PRINT "POČASI NA EKTRAN,"  
130 FOR L = 1 TO 500: NEXT  
140 PRINT "TAKO DA UPORABI PREPROSTO ZANKO"  
150 FOR L = 1 TO 500: NEXT  
160 PRINT "ZA UPOČASNITEV PISANJA!"  
170 FOR L = 1 TO 500 : NEXT
```