

Primer: 100 REM IZPISATI ČRKE OD A DO G  
 110 A=65 : REM CHR\$ KODA ZA ČRKO A  
 120 REPEAT  
 130 : PRINT CHR\$ (A);  
 140 : A=A+1  
 150 UNTIL A > 71  
 160 END

### 13.3. RCOMP

Format: RCOMP (ukazi če je pogoj izpolnjen): ELSE (če ni izpolnjen)

Namen: Nadaljevati IF .. THEN .. ELSE v naslednji vrsti.

Delovanje: RCOMP uporabljamo takrat ko želimo razširiti IF stavek v naslednjo vrsto in pri tem ne želimo ponovno pisati isti IF .. THEN ukaz.

Primer: 100 INPUT A  
 110 IF A=0 THEN PRINT "+A = 0" : ELSE : PRINT "A <> 0"  
 120 RCOMP: PRINT "(NIČ)" : ELSE: PRINT "(NI NIČ)"  
 (RCOMP je uporabljen namesto IF A=0 THEN)

### 13.4. LOOP .. EXIT IF .. END LOOP

Format: LOOP (niz ukazov) EXIT IF (pogoj) END LOOP

Namen: Ponavljati del programa dokler ni izpolnjen določen pogoj.

Delovanje: Ta instrukcija se uporablja, kadar je potrebno ponoviti del programa večkrat, število ponovitev pa ni znano.

Primer: 100 LOOP :REM PRIČETEK ZANKE  
 110 : GET A\$ :REM PREBERI TASTATURO  
 120 : EXIT IF A\$  
 ="" :REM ČE JE SPACE KONČAJ ZANKO  
 130 : PRINT A\$ :REM IZPIŠI A\$  
 140 : END LOOP :REM KONEC ZANKE  
 150 : PRINT "NADALJEVANJE"

### PROCEDURE

Za olajšanje pisanja strukturiranih programov nam Simon's Basic omogoča pisanje t.i. podprogramov in procedur. To so programi, ki jih kličemo z imenom in ne s številko vrstice kot je to v standardnem Basic-u z ukazom GOSUB.

### 13.5. PROC

Format: PROC labela

Namen: Definirati simbolični naslov (labelo)