



```

; Med ukazi je klic procedure VAJA z ukazom CALLG. Prvi podatek
; je naslov tabele argumentov, ki se bo prekopiral v register AF,
; drugi podatek je naslov procedure.
CALLG  PODATKI,VAJA

```

5.2.2 Format procedure

Proceduro začnemo tako kot glavni program z definicijo vstopne točke z ukazom `.ENTRY`. Ime, ki ga pišemo v tem ukazu, je ime procedure, ki ga uporabimo v ukazih `CALLS` ali `CALLG` ali pri klicu te procedure iz višjega programskega jezika. Drugi parameter pri ukazu `.ENTRY` je 16-bitna maska, ki pove, katere registre želimo shraniti. Bit 0 pomeni register R0, bit 1 register R1 itd. V masko zapišemo registre, ki jih uporabljamo v proceduri. Pri klicu procedure se vrednosti teh registrov zapišejo na sklad in ob izhodu iz procedure dobijo registri spet tiste vrednosti, ki so jih imeli pred klicem procedure. Kličočemu programu procedura torej ne pokvari stanja v registrih.

Proceduro končamo tako, da v register R0 ali v R0 in R1 zapišemo status, s katerim se je izvajanje procedure končalo, če uporabljamo proceduro kot funkcijo, pa v teh registrih vrnemo vrednost funkcije. Status uspešno ima simbolično ime `SS$NORMAL`, številska vrednost tega simbola pa je 1. Status zapišemo v register R0 z ukazom

```
MOVL  #SS$_NORMAL,R0
```

pred ukazom `RET`, s katerim vrnemo kontrolo v kličoči program.

Začetek in konec procedure sta enaka kot pri glavnem programu, ki je tudi neka procedura. Kaj je glavni program in kaj podprogram določimo z imenom pri ukazu `.END`. To ime pove, na kateri vstopni točki se bo začel izvajati program, ko ga aktiviramo z ICL ukazom `RUN`. Ukaz `.END` mora biti na koncu vsake datoteke z makro programom ali delom makro programa, vendar pišemo ime vstopne točke le pri enem.

5.2.3 Uporaba argumentov v proceduri

Pri klicu procedure se zapiše naslov, na katerem je začetek tabele argumentov v register AF. Do podatkov pridemo torej v proceduri preko registra AF. Če želimo argumente procedure VAJA prenesti v registre tako, da bosta števili, ki sta prva argumenta, v registrih R2 in R3, naslov teksta pa v R4, bomo to naredili tako:

```

; Procedura VAJA
; Argumenti so:      začetna pozicija izbranega podteksta
;                   dolžina izbranega podteksta
;

```