

!	N	!
!	Argument 1	!
	⋮	
!	Argument N	!

Slika 5.1: Tabela argumentov.

Argumente lahko prenašamo z vrednostjo, z naslovom ali z opisnikom. V prvem primeru zapišemo v eno dolgo besedo samo vrednost podatka, v drugem primeru zapišemo naslov, na katerem se podatek začne, v tretjem pa zapišemo naslov opisnika, ki je sestavljen iz 4 polj s skupno dolžino 8 bytov. V prvih dveh bytih opisnika je zapisana dolžina podatka v bytih, v naslednjih dveh bytih sta podatka o tipu in vrsti podatka, sledi pa dolga beseda z naslovom, na katerem se začne podatek. Natančnejši opis standarda je v dodatku C knjige Architecture Handbook.

Vsi višji programski jeziki prenašajo argumente z naslovom, če prenašajo tekste pa z opisnikom. Z ukazi, ki so v različnih programskih jezikih različni, lahko zahtevamo, da se prenašajo argumenti na katerikoli od zgoraj omenjenih treh načinov.

Za primer si oslejmo, kako pripravimo argumente in pokličemo proceduro VAJA, ki ima za prva dva argumenta števili, ki ju prenesemo z naslovom, za tretjega pa tekst, ki ga določimo z opisnikom.

Klic s CALLS:

```

PUSHAQ TEKST           ; Tretji argument je naslov
                        ; opisnika teksta.
PUSHAL STEVILO2        ; Drugi argument je naslov
                        ; števila,
PUSHAL STEVILO1        ; prav tako prvi argument.

```

; Klic procedure VAJA. Prvi podatek je število argumentov in se  
; zapiše na sklad nad ostale argumente, drugi podatek je naslov  
; procedure.

```
CALLS #3,VAJA
```

Klic s CALLG:

; Med podatki definiramo tabelo z argumenti:

PODATKI:

```

.LONG 3                 ; Število argumentov.
.ADDRESS STEVILO1      ; Prvi argument - naslov števila.
.ADDRESS STEVILO2      ; Drugi argument - naslov števila.
.ADDRESS TEKST         ; Tretji argument - naslov
                        ; opisnika teksta.

```