

## Yksinkertainen aliohjelma

Kirjoitetaan symbolisella konekielellä pääohjelma ja aliohjelma. Aliohjelman nimi on KulutaAikaa ja sillä on yksi arvoparametri, joka on kokonaisluku. Aliohjelman pseudokoodi Javan kaltaisella kielellä on:

```
void KulutaAikaa(int Lkm) {  
    int i;  
    for (i = 0; i <= Lkm; i++)  
        Tulosta(i);  
}
```

Pääohjelma vain kutsuu aliohjelmaa kahdesti arvoilla 1000 ja 5. Koska pääohjelma ei käytä rekistereitä, niitä ei tarvitse viedä pinoon ennen aliohjelman suoritusta.

### Pääohjelma:

	<b>PUSH SP, =1000</b>	Pushataan pinoon arvoparametri ”1000”.
	<b>CALL SP, KulutaAikaa</b>	Kutsutaan aliohjelmaa. CALL-käsky automaattisesti laittaa vanhan PC:n arvon ja FP:n arvon pinoon. Se asettaa PC:n arvoksi aliohjelman osoitteen (ihan tavallinen hyppykäsky). FP:n ja SP:n arvot päivitetään myös uutta tilannetta vastaavaksi.
	<b>PUSH SP, =5</b>	Pushataan pinoon arvoparametri ”5”.
	<b>CALL SP, KulutaAikaa</b>	Kutsutaan aliohjelmaa.
	<b>SVC SP, =HALT</b>	Ohjelma loppuu.

### Aliohjelma:

<b>KulutaAikaa</b>	<b>LOAD R1, =1</b>	Aliohjelma alkaa. Jos pääohjelma käyttäisi rekistereitä, ne pitäisi tallettaa käyttämällä PUSH- <del>HR</del> käskyä ennen kuin aliohjelma alkaa käyttää niitä.  Ladataan rekisteriin R1 arvo 1.
<b>Vielä</b>	<b>COMP R1, -2(FP)</b>	Aliohjelmalle annettu arvoparametri on osoitteessa ”-2(FP)” eli FP - 2 (ks. alla oleva kuva). Verrataan sitä rekisterin R1 arvoon, koska halutaan ”kuluttaa aikaa” niin monta kertaa kuin käyttäjä antoi parametrina.

Luentomonistetta ”Tietokoneen toiminta, Auvo Häkkinen, 1998” mukaillen.

	<b>JGRE Pois</b>	Jos ollaan kulutettu tarpeeksi aikaa, eli rekisterin R1 arvo on suurempi kuin parametri, hypätään kohtaan ”Pois”.
	<b>OUT R1, =CRT</b>	Tulostetaan rekisterin R1 arvo näytölle.
	<b>ADD R1, =1</b>	Lisätään rekisterin R1 arvoon 1.
	<b>JUMP Viela</b>	Hypätään silmukan alkuun.
<b>Pois</b>	<b>EXIT SP, =1</b>	<p>Poistutaan aliohjelmasta. EXIT-käsky saa aikaan, että</p> <ul style="list-style-type: none"> <li>- kaikki mitä pinossa on FP:n jälkeen vapautetaan</li> <li>- FP:n vanha arvo kopioidaan FP:hen</li> <li>- PC:n vanha arvo kopioidaan PC:hen</li> <li>- Pinosta poistetaan FP, PC ja EXIT-käskyn parametrin osoittama määrä muistipaikkoja (eli käytännössä siirretään stack pointeria sopiva määrä)</li> </ul> <p>Jos pääohjelma olisi käyttänyt rekistereitä ja ne olisi tallennettu pinoon, ne täytyisi palauttaa ennen kuin EXIT-käskyä kutsutaan.</p>

Ohjelma + aliohjelma	
1000 (arvoparametri)	← FP - 2
Vanha PC tallennettu tänne	← FP - 1
Vanha FP tallennettu tänne	← SP, FP