



Pravila

Rješenja zadaće će biti izlagana na vježbama 10.1.2022. Rok za predaju je 9.1.2022. u 12:00 sati. Sav kod zadaće predajte na GitHub u repozitorij MORAS, u mapu MORAS/zadaca.3.

Zadatak 1 (10). Implementacija parsera assemblera napravljena na vježbama assemblyersku liniju

```
(NEKA_OZNAKA) A + B + C + D
```

neće smatrati pogrešnom. Naime, ta implementacija će ignorirati sve što se nalazi nakon deklaracije oznake. Promijenite parser na način da takvu situaciju smatrate pogrešnom. Dakle, bilo kakav neprazni string koji dolazi nakon deklaracije oznake smatrajte pogreškom, prekinite parsiranje i javite poruku o pogrešci.

Zadatak 2 (10 + 10 + 10 + 20). Parser assemblera implementiran na vježbama proširite makro naredbama. Svaka makro naredba počinje znakom "\$" nakon čega slijedi ime naredbe te njezina lista argumenata.

```
$NAREDBA (ARG1, ARG2, ...)
```

Argumenti makro naredbi su numeričke vrijednosti i predstavljaju adrese u memoriji (RAM) ili nazive varijabli/oznaka. Implementirajte sljedeće makro naredbe.

- (a) Naredba "\$MV(A, B)" u B sprema sadržaj A.
- (b) Naredba "\$SWP(A, B)" mijenja sadržaj A i B.
- (c) Naredba "\$SUM(A, B, D)" sumira vrijednosti A i B i sprema ih u D.
- (d) Naredba "\$WHILE(A)" nakon koje dolazi blok naredbi terminiran s "\$END". Taj se blok naredbi ponavlja sve dok $RAM[A]$ ne bude postavljen na vrijednost 0.

Napomena: Preporučeno je makro naredbe pretvoriti u assemblyerski kod nakon što su uklonjeni komentari i znakovi razmaka počevši s naredbom *WHILE*. Pripazite na ugnježdavanje while petlje.

Primjer 1.1. Računamo sumu brojeva od 1 do n . Pretpostavimo da su već alocirane i inicijalizirane varijable n i $s = 0$.

```
$WHILE (n) // dok je n razlicit od 0
@n
M = M - 1
$ADD (s, n, s) // s = s + n
$END
```



Zadatak 3 (15 + 25). Implementirajte sljedeće asemblerske programe. Preporučeno je koristiti prošireni assembler iz prethodna dva zadatka.

- (a) Program koji u $R2$ sprema rezultat potencije čija se baza nalazi u $R0$, a eksponent u $R1$. Eksponent i baza će uvijek biti nenegativni brojevi.
- (b) Neka je u $R0$ dana duljina polja cijelih brojeva koje počinje na adresi 100 u RAM-u. Npr. ako je $RAM[0] = 7$ polje se proteže od $RAM[100]$ do $RAM[106]$. To polje sortirajte koristeći proizvoljan algoritam.