

Napomene:

1. Obavezno pročitati **SVE** napomene.
2. Zadatak snimiti pod imenom **zad.S**. Ovo je jedini fajl koji će biti pregledan.
3. Obavezno upisati **ime, prezime i broj indeksa** u komentar na početku fajla. Nepotpisani zadaci neće biti pregledani.
4. Rešenje koje ne može da se kompajlira nosi **0** poena.
5. Obavezno iskomentarisati kod.
6. Koristiti praktikum iz ispitnih materijala (**ne** kopirati ga kod sebe u direktorijum).
7. Za vreme testa nema pauze. Izlazak iz učionice se smatra završetkom testa.
8. Asistenti za vreme testa pružaju informacije vezane isključivo za razumevanje teksta zadatka i pomoć oko korišćenja programskih alata.
9. Nazivi promenljivih koje se inicijalno nalaze u zad.S fajlu se **ne smeju menjati**. Njihove vrednosti se mogu (i trebaju) menjati, da se ispitaju razni ulazi za program. Sadržaji stringova koji su zadati u zad.S fajlu se takođe **ne smeju menjati**. Ukoliko je potrebno, mogu se dodavati nove promenljive.
10. **Test se snima.**
11. Test traje 2 sata i 45 minuta.

Zadatak:

Napisati assemblerski program koji računa sumu brojeva koji se nalaze u datom opsegu, kao i sumu brojeva koji se nalaze van datog opsega. Manju od te dve sume treba ispisati na ekran u heksadecimalnom formatu.

Format unosa je sledeći:

- U uglastim zagradama je naveden opseg. Opseg je inkluzivan—donja i gornja granica ulaze u opseg. Granice su zadate kao neoznačeni 32-bitni celi brojevi u dekadnom brojnom sistemu, gde iza svakog broja sledi znak zarez „,“. Smatrati da se neće unositi vrednosti tako da donja granica bude veća od gornje.
- Posle opsega sledi par vitičastih zagrada u kojima se nalazi niz neoznačenih 32-bitnih celih brojeva u dekadnom brojnom sistemu. Iza svakog broja sledi znak zarez „,“.
- Karakteri koji stoje iza zatvorene vitičaste zgrade se ne posmatraju.
- Maksimalna dužina unosa je 100 karaktera. Smatrati da neće biti uneto više od 10 elemenata niza. Smatrati da će neophodne zgrade i zarezi uvek biti prisutni.

Na primer:

[5,10,]{1,3,5,7,10,25,30,}ostatak se ne posmatra

Ukoliko dođe do prekoračenja prilikom računanja sume, treba ispisati poruku **greska1**.

Ukoliko dođe do greške prilikom konverzije brojeva iz stringa u interni oblik, treba ispisati poruku **greska2**.

Izlazni kod programa treba da bude **0** ako nije bilo greške, odnosno **1** ako jeste.

Primer interakcije sa programom:

Unesite izraz: [5,10,]{1,3,5,7,10,25,30,}dodatak

Suma brojeva u opsegu je manja.

Suma: 0x16

Unesite izraz: [1,2000000000,]{2000000000,2000000000,294967295,4294967295,}

Sume su jednake.

Suma: 0xFFFFFFFF

Unesite izraz: [5,10,]{1,3,5,7,10,25,429496729**5**,}dodatak

GRESKA: Prekoracenje prilikom racunanja sume.

Unesite izraz: [5,10,]{1,3,5,7,10,25,429496729**6**,}dodatak

GRESKA: Problem prilikom konverzije brojeva.

Bodovanje zadatka će u velikoj meri zavisi od procenta uspešnih testova.

Napomena: ukoliko testovi koji ne treba da izazovu grešku ne prolaze, testovi koji treba da izazovu grešku se ne uzimaju kao validni.

Pored testova koji su unapred dati (automatizovano testiranje sa **./testiraj.sh zad.S**), prilikom pregledanja rešenje će se testirati sa još dodatnih testova, te je potrebno testirati i sa drugim ulazima.

Napomena: **testiraj.sh** ima smisla pokretati tek kada je zadatak završen.

Za kompletno odrađen zadatak se dobija 30 poena.