

Zadaci sa rešenjima dostupnim na sajtu

1. Napisati asemblerski program koji određuje sumu elemenata podniza:
 - a. Proširiti kod sa 6. slajda sa prezentacije za 3. nedelju vežbi
 - b. Rezervisati memorijski prostor za još dve promenljive tipa *long* – *ind1* i *ind2*; sabrati elemente koji se nalaze između ova dva indeksa
 - c. Koristiti indeksno adresiranje, kao na pomenutom 6. slajdu
2. Napisati asemblerski program koji nalazi minimalni i maksimalni element niza označenih brojeva:
 - a. Rezervisati memorijski prostor za još dve promenljive – *min* i *max*, koje će po završetku programa sadržati minimalni i maksimalni element niza
 - b. Koristiti indirektno adresiranje, kao na 7. slajdu sa prezentacije za 3. nedelju vežbi
3. U asemblerskom jeziku napisati program koji izbacuje razmake sa početka stringa:
 - a. Proširiti kod sa 14. slajda sa prezentacije za 3. nedelju vežbi
 - b. Zadržati kod za brisanje razmaka sa kraja stringa, koji se već nalazi na pomenutom slajdu; konačan string treba da bude takav da ne sadrži razmake ni na početku, ni na kraju stringa
 - c. Kao pomoć, koristiti C kod sa 13. slajda sa prezentacije za 4. nedelju vežbi
 - d. **Ideja** – pomerati ceo string za jedno mesto ulevo, sve dok se na početku ne nađe karakter različit od blanko karaktera
4. Napisati asemblerski program za sortiranje niza 16-bitnih označenih vrednosti (u proizvoljnom redosledu):
 - a. Koristiti proizvoljni tip adresiranja
 - b. Dostupno rešenje sortira niz u rastućem redosledu, primenom bubble-sort algoritma

Zadaci bez rešenja dostupnih na sajtu

1. Napisati asemblerski program za određivanje aritmetičke sredine niza
 - a. Koristiti operaciju za deljenje
 - b. Raditi sa vrednostima proizvoljne veličine
 - c. Koristiti proizvoljni tip adresiranja
2. Napisati asemblerski program za sortiranje niza karaktera po abecedi:
 - a. Sortirati u proizvoljnom redosledu
 - b. Rezervisati memorijski prostor i u njega upisati string proizvoljne veličine; sortirati karaktere ovog niza
 - c. Koristiti proizvoljni tip adresiranja
3. Napisati asemblerski program koji određuje pokomponentni zbir dva niza:
 - a. Rezervisati memorijski prostor za dva niza jednake dužine, sa elementima proizvoljne veličine
 - b. Rezultat smestiti u treći niz u memoriji
 - c. Ako se sabiraju nizovi *a* i *b*, a rezultat se smešta u niz *c*, tada za svaki element niza *c* važi da je $c[i] = a[i] + b[i]$
 - d. Koristiti proizvoljni tip adresiranja
4. Napisati asemblerski program koji nalazi indekse svih minimalnih i maksimalnih elemenata niza:
 - a. Početni niz može sadržati duplikate, tako da se na više indeksa u nizu mogu naći elementi sa maksimalnom i minimalnom vrednošću

- b. Za upis rezultata, rezervisati još dva niza, jedan za indekse maksimalnih, a drugi za indekse minimalnih elemenata
- c. Veličine nizova za rezultate mogu biti jednake veličini početnog niza
- d. Koristiti proizvoljni tip adresiranja