

Zadaci sa rešenjima dostupnim na sajtu

1. Napisati asemblerski program koji određuje sumu elemenata podniza:
 - a. Rezervisati memorijski prostor za dve promenljive tipa long – ind1 i ind2; sabrati elemente koji se nalaze između ova dva indeksa
 - b. Koristiti indeksno adresiranje
2. Napisati asemblerski program koji nalazi minimalni i maksimalni element niza označenih brojeva:
 - a. Rezervisati memorijski prostor za dve promenljive – min i max, koje će po završetku programa sadržati minimalni i maksimalni element niza
 - b. Koristiti indirektno adresiranje
3. 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 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
3. 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