**Ime:**

**Prezime :**

**Matični broj:**

1. Za idući kôd odrediti vremensku složenost T(N) i vremensku klasu složenosti O(f(N)) (10 bodova):

int f(int \*niz, int n) {

int j = 1;

int i = 0, sum = 0;

while (j < 2 \* n) {

sum += niz[j];

j++;

}

while(i < n \* n) {

sum += niz[j];

i++;

}

return sum;

}

1. Procijeniti složenost O() algoritma koji iz neke datoteke čita elemente (brojeve) u jednostruko vezanu listu. Broj pročitanih elemenata je N i ne koriste se nikakvi posebni pokazivači (npr. na kraj liste). Algoritam se može napisati na dva načina:
2. Svaki novi element se dodaje na početak liste.
3. Svaki novi element se dodaje na kraj liste.

Složenost izraziti u N.

(20 bodova)

1. Napisati funkciju koja prima jednostruko vezanu listu (svaki element sadrži broj) i provjerava da li se svi elementi sa parnim brojem nalaze između dva elementa sa neparnim brojem. Prvi i zadnji element sadrži neparni broj, a primljena lista ima barem 3 elementa. Funkcija vraća 1 ako jesu ili 0 ako nisu. (25 bodova)
2. Napisati funkciju koja prima vezanu listu (svaki element sadrži broj) i jedan dodatni broj. Funkcija će dodati novi element sa tim brojem prije najvećeg broja u listi. (35 bodova)
3. Prikazati quicksort algoritam na nizu { 9, 3, 6, 7, 10, 4, 11, 5, 7 }. (10 bodova)