**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 i = 0, sum = 0;

while(i < 2 \* n) {

sum += niz[i];

i++;

}

int j = 0;

while (j < n \* n) {

sum += niz[j];

j++;

}

return sum;

}

1. Procijeniti složenost O() algoritma koji računa operaciju za jedan skup i još N skupova dužine. Prvi skup je niz dužine M, a ostalih N skupova su nizovi dužine K. Algoritam iz prvog skupa ispisuje sve elemente koji se pojavljuju u ostalih N skupova. Algoritam se može napisati na dva načina:
2. Nijedan niz nije sortiran. Prolazimo kroz prvi niz i za svaki broj provjeravamo da se pojavljuje u N preostalih nizova.
3. Sortiramo N nizova. Zatim prolazimo kroz prvi niz i za svaki broj provjeravamo da se pojavljuje u N preostalih nizova binarnom pretragom.

Složenost izraziti u N, M i K. Možete pretpostaviti da se za sortiranje u (b) koristi quicksort.

(20 bodova)

1. Napisati funkciju koja prima jednostruko vezanu listu (svaki element sadrži broj) i jedan broj. Funkcija će ispisati sve elemente liste veće od tog broja. (25 bodova)
2. Napisati funkciju koja prima vezanu listu (svaki element sadrži broj i svi su različiti) i jedan dodatni broj. Funkcija će izbrisati element sa tim brojem. (35 bodova)
3. Prikazati quicksort algoritam na nizu { 4, 2, 6, 1, 7, 4, 1, 3, 5 }. (10 bodova)