

## Programiranje 2 - primer teorijskog dela kolokvijuma

Ime, prezime i broj indeksa: \_\_\_\_\_

1. Napisati izraz kojim se komplementira svaki drugi bit promenljive `char x`, počevši od bitova najmanje težine, dok ostali bitovi ostaju neizmenjeni.

2. Dve mane rekurzije su \_\_\_\_\_ i \_\_\_\_\_.

3. Eliminirati repnu rekurziju u narednom kodu.

```
int pretraga(int niz[], int n, int x) {  
    if (n==0) return -1;  
    else if(niz[n-1]==x) return n-1;  
    else return pretraga(niz, n-1, x)  
}
```

4. Prikazati kako se menja stanje na steku prilikom poziva `f(4)` gde je `f` funkcija definisana u zadatku 6. U stek okvirima navesti vrednosti lokalnih promenljivih.

5. U tabeli označiti sve odnose koji važe.

$\in$	$O(n^2)$	$\Omega(n^2)$	$\Theta(n^2)$
$2n + 2$			
$n^3 + 2n^2$			
$3n^2 + 100n$			

6. Vremenska složenost naredne funkcije je  $\Theta(\text{_____})$ , a prostorna je  $\Theta(\text{_____})$ .

```
unsigned f(unsigned n) {  
    if(n<=1) return 1;  
    return f(n-2)+f(n-2);  
}
```

7. Vremenska složenost narednog koda je  $\Theta(\text{_____})$ , a prostorna je  $\Theta(\text{_____})$ .

```
for(i=0; i<n; i++)  
    for(j=0; j<n; j++) {  
        if(j==1) break;  
        s++;  
    }
```

8. Koja vrsta pretrage je primenljiva u većem broju slučajeva? \_\_\_\_\_ Koji je preduslov primene binarne pretrage? \_\_\_\_\_

9. Koji algoritam sortiranja ima optimalnu vremensku složenost najgoreg slučaja? \_\_\_\_\_ Koja je to složenost? \_\_\_\_\_

10. Koja je vremenska složenost algoritma brzog sortiranja u najgorem slučaju? \_\_\_\_\_ Šta se u tom algoritmu postiže particionisanjem?

\_\_\_\_\_  
\_\_\_\_\_