

Diskretne i kombinatorne metode za računarsku grafiku

1. Data je funkcije u programskom jeziku C koja kvadratnu matricu smeštenu u niz po vrstama transponuje "u mestu".

```

1 void transposeA(double *A, int n)
2 {   int i, j;   double temp;
3     for (i=0; i<n; i++){
4         for (j=i+1; j<n; j++){
5             temp = A[i*n+j];
6             A[i*n+j] = A[j*n+i];
7             A[j*n+i] = temp;
8         }
9     }
10 }
```

Dodeliti vreme izvršavanja linijama 2, 3, 4, 5, 6, 7, redom $c_2, c_3, c_4, c_5, c_6, c_7$ i odrediti vreme izvršavanja procedure transposeA, $T(n)$, u zavisnosti od formata ulazne matrice A, $n \times n$.

2. Za matricu

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 4 & -7 & -16 \\ 5 & 9 & -8 \end{bmatrix}$$

napisati izgled transponovane matrice u nizu.

Za matricu A odrediti broj izvršavanja linije 6.

3. Dati definiciju "velikog O" ponašanja i pokazati da je $n \ln n + n = O(n^2)$.

Da li je $n \ln n + n = O(n)$?

Da li je $\frac{3}{4}n^2 + 3n \ln n = O(n^2)$?

4. Neka je u zadatku 1, $c_2 = c_3 = c_4 = c_5 = c_6 = c_7 = 1$. Izračunati $T(n)$ u zavisnosti od n i pokazati da je $T(n) = O(n^2)$.

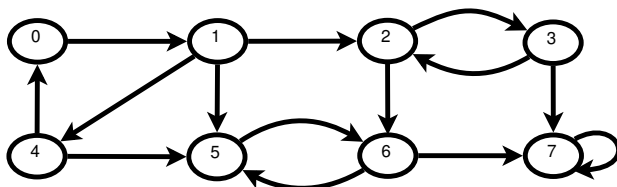
5. Napisati u programskom jeziku C procedure push i printstack iz implementacije ADT stack preko povezanih listi.

```

typedef char listdata;
typedef struct _node node;
typedef node *stack;
```

```

struct _node
{
    listdata data;
    node *next;
};
int push(stack *, listdata);
void printstack(stack);
```



6. Primeniti algoritam DFS na graf sa slike, uzimajući čvorove i grane leksikografski.

Pored čvorova napisati d i f vrednosti.

Napraviti tabelu zagrada. Nacrtati šumu ovog DFS. Označiti tipove grana kad se prvi put otkrije.

7. Da li je graf iz prethodnog zadatka usmereni aciklični graf? Obrazložiti.

Odrediti komponente jake povezanosti grafa sa slike.

8. Rešiti problem angažovanja radnika A, B, C, D na poslove 1, 2, 3, 4, ako su u tabeli vremena potrebna radniku da obavi odgovarajući posao.

	1	2	3	4
A	11	8	9	5
B	12	15	13	10
C	8	11	14	8
D	15	13	12	9

Bodovi: 1→10, 2→10, 3→10, 4→10, 5→15, 6→15, 7→10, 8→10,