

Diskretne i kombinatorne metode za računarsku grafiku

```

1: procedure NEKI SORT( $A$ )
2:    $n \leftarrow \text{length}(A)$ 
3:   for  $i \leftarrow 1$  to  $n - 1$  do
4:      $i_{\min} \leftarrow i$ 
5:     for  $j \leftarrow i + 1$  to  $n$  do
6:       if  $A[j] < A[i_{\min}]$  then
7:          $i_{\min} \leftarrow j$ 
8:       end if
9:     end for
10:    if  $i \neq i_{\min}$  then
11:      swap( $A[i]$ ,  $A[i_{\min}]$ )
12:    end if
13:    writeln( $A$ )
14:  end for
15: end procedure

```

1. Primeniti algoritam NEKI SORT na ulaz $A = [2, 3, 4, 5, 1]$ i ispisati stanje niza A koje se ispisuje u liniji 13.
2. Za ulazni niz $[2, 3, 4, 5, 1]$, koliko će puta upoređivanje u liniji 6 biti izvršeno, a koliko puta zamena (swap) u liniji 11?
3. Za obrnuto sortirani ulazni niz A dimenzije n , koliko će puta upoređivanje u liniji 6 biti izvršeno, a koliko puta zamena (swap) u liniji 11?
4. Dati definiciju "velikog O" ponašanja i pokazati da je $\frac{2}{3}n^2 + 16n = O(n^2)$.
Da li je $P(n) = \Theta(n^2)$, gde je $P(n)$ broj poređenja algoritma NEKI SORT za ulazni niz dužine n ?

5. Napisati u Programskom jeziku C proceduru koja za graf smešten u Adjacency list G vraća stepen $s[i]$ svih čvorova $G[i]$, $i = 1, 2, \dots, n$.

Graf je smešten u niz listi susedstva

```

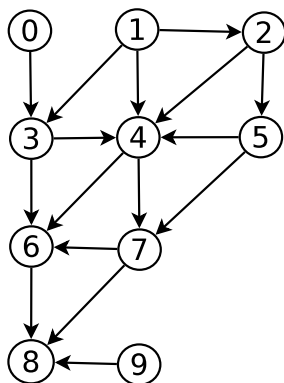
typedef int cvor;
typedef struct _node gnode;
typedef gnode *grana;

```

```

struct _node
{
    cvor data;
    gnode *next;
};
void stepen(grana G[], int n,
             int s[])
{
    // G[], n su ulaz, s[] je izlaz
    // Ovde ide trazeni kod
}

```



6. Primeniti DFS algoritam na graf sa slike. Pored čvorova ispisati d i f vrednosti, na granama napisati tip grane kada se prvi put otkrije: T = tree, F = forward, C = cross, B = back.

Čvorove i grane uzimati leksikografski.

7. Da li je graf sa slike usmereni aciklični graf (DAG)? Obrazložiti.

Ako se ignorišu usmerenja grana, da li je graf sa slike drvo? Obrazložiti.

Ako je graf sa slike DAG, napisati re-dosled čvorova koji daje topološko sortiranje dobijeno primenom DFS algoritma iz prethodnog zadatka.

8. Rešiti problem angažovanja radnika A, B, C, D na poslove 1, 2, 3, 4, 5.

	1	2	3	4	5
A	14	22	8	12	14
B	14	14	16	14	15
C	19	11	14	17	15
D	12	8	11	18	11
E	13	9	17	20	11

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