

Diskr. i komb. met. za rač. gr. 1→15, 2→5, 3→10, 4→10, 5→5, 6→15, 7→10, 8→10.

Data je procedura stepen koja za graf smešten u niz listi susedstva $G[]$ sa n čvorova nalazi stepen svakog čvora $s[]$.

```

1 void stepen(grana G[], int n,
2   int s[])
3 {
4   int i;
5   grana gr;
6   for(i=0; i<n; i++){
7     gr = G[i];
8     s[i] = 0;
9     while(gr){
10      (s[i])++;
11      gr = gr->next;
12    }
13  }
14
15 }
```

1. Naći vreme izvršavanja $T(n, m)$ procedure stepen u zavisnosti od broja čvorova grafa n i broja grana grafa m i vremena c_k izvršavanja linije k .

Dati asimptotsku ocenu za $T(n, m)$.

2. Dati definiciju malog o ponašanja. Da li je za sve $f = f(n)$ i $g = g(n)$ tačno:
 $f = o(g) \Rightarrow f = O(g)$?
 $f = \Theta(g) \Rightarrow f = O(g)$?
 $f = \Omega(g) \Rightarrow f = \Theta(g)$?
3. Napisati algoritam za sortiranje umetanjem, takozvani BUBBLE SORT.
4. Za algoritam BUBBLE SORT iz prethodnog zadatka naći broj zamena za ulazni niz [9, 8, 3, 2, 1, 5, 3].

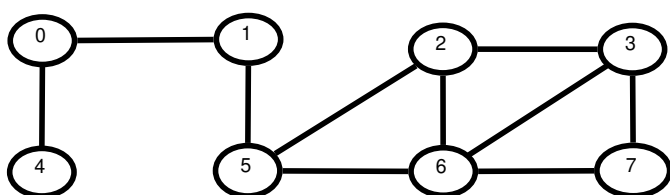
5. Napisati u programskom jeziku C procedure push i printstack iz implementacije ADT stack preko povezanih listi.

```

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.

Označiti tipove grana (TBFC).

7. Za graf iz prethodnog zadatka dati reprezentaciju nizom listi susedstva.

Primeniti na graf sa iz prethodnog zadatka BFS algoritam, polazeći od čvora 2, dati tabelu udaljenosti d i prethodnika p .

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

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

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