

Задатак 1.

Написати програм на језику C++ који:

a) дефинише структуру типа student чији су чланови:

ime (низ карактера)

smer (низ карактера)

br_indeksa (цео број)

b) декларише динамички (ев. статички) низ за 10 структура типа student, чита са тастатуре и уписује у текст датотеку studenti.txt запис о свакој структури чији је br_indeksa мањи од 100.

c) чита структуре из датотеке studenti.txt и приказује на екрану само оне које имају вредност податка smer тражену са тастатуре, понавља овај поступак за задате највише три међусобно различите вредности податка smer и након тога се завршава.

Решење 1.

```
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <string.h>
#define MAX 5
using namespace std;
struct student{ char ime[MAX+1], smer[MAX+1]; int br_indeksa; };
int main()
{
    int i, j, n=0, istih=0, razlicitih=0;
    char ime[MAX+1], smer[MAX+1], *smerovi[MAX];
    fstream f;
    student *niz;

    niz=new student[MAX]; if(niz==NULL){cout<<"Nije uspela dodela";exit(1);}
    f.open("studenti.txt", ios::out | ios::in | ios::trunc); if(!f){cout<<"Greska datoteke";exit(1);}

    cout<<"\nUneti za svakog studenta u posebnom redu: ime, smer, broj ind"<<endl;
    for(i=0; i<MAX; i++)
    {
        cin>>niz[i].ime>>niz[i].smer>>niz[i].br_indeksa;
        if(niz[i].br_indeksa<=100)
        {
            f<<niz[i].ime<<' '<<niz[i].smer<<' '<<niz[i].br_indeksa<<endl; n++;
        }
    }delete niz;

    niz=new student[n]; if(niz==NULL){cout<<"Greska dodele";exit(1);}
    for(i=0; i<n; i++)
    {
        smerovi[i]=new char[MAX+1]; if(smerovi[i]==NULL){cout<<"Greska";exit(1);}
        f.seekg(0);
        for(i=0; i<n; i++)
        {
            cout<<"\nsmer: ";cin>>smer;strcpy(smerovi[i],smer);

            for(j=0; j<i; j++) if(strcmp(smerovi[j],smer)==0){istih=1;continue;}
            if(istih){istih=0;continue;}

            f>>niz[i].ime>>niz[i].smer>>niz[i].br_indeksa;
            if(strcmp(niz[i].smer,smer)==0)
                cout<<niz[i].ime<<' '<<niz[i].smer<<' '<<niz[i].br_indeksa<<endl;
            if(++razlicitih==3)break;
        }
    }
    f.close(); delete niz; for(i=0; i<n; i++) delete smerovi[i]; return 0;
}
```

Задатак 2.

Написати програм на језику C++ који:

- a) чита са тастатуре $n \leq 20$ речи (реч је низ знакова до празног знака),
- b) уписује речи са тастатуре у текст датотеку prva.txt,
- c) уписује исте речи, без речи које не почињу словом, у бинарну датотеку druga.bin,
- d) чита речи из датотеке druga.bin и само речи у задатом са тастатуре опсегу редних бројева речи, уписује у текст датотеку treca.txt.

Решење 2.

```
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#define BROJ 10
#define MAX 80
using namespace std;
int main()
{
    int i, l, a, b, n, n2=0; char rec[MAX+1];
    fstream f1, f2, f3;

    f1.open("prva.txt",ios::out);if(!f1){cout<<" Neuspelo otvaranje dat!";exit(1);    }
    f2.open("druga.bin",ios::out | ios::binary);if(!f2){cout<<" Neuspelo otvar dat!";exit(1);}

    do{    cout<<"\nUnesite n: ";cin>>n;cin.get();    }while(n<1 || n>BROJ);

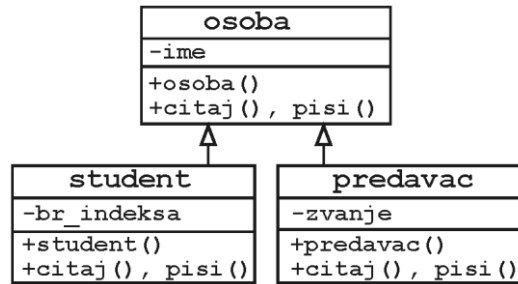
    cout<<"\nUnesite reci:\n";
    for(i=0; i<n; i++)
    {
        cin>>rec;
        f1<<rec<<' ';
        l=strlen(rec);
        if( !isalpha(rec[0]) )
        {
            f2.write((char *)&l, sizeof(int));
            f2.write((char *)rec, l+1);
            n2++;
        }
    }
    f1.close();f2.close();

    f2.open("druga.bin",ios::in | ios::binary);if(!f2){cout<<" Neuspelo otvaranje dat!";exit(1);}
    f3.open("treca.txt",ios::out);if(!f3){cout<<" Neuspelo otvaranje dat!";exit(1);    }
    do
    {
        cout<<"\nUnesite opseg: \n"; cin>>a>>b;
    }while(a<1 || a>n2 || b<1 || b>n2 || a>b);
    for(i=0; i<(a-1); i++)
    {
        f2.read((char *)&l, sizeof(int));
        f2.read((char *)&rec, l+1);
    }
    for(i=0; i<=(b-(a-1)); i++)
    {
        f2.read((char *)&l, sizeof(int));
        f2.read((char *)&rec, l+1);
        f3<<rec<<' ';
    }
    f2.close(); f3.close(); return 0;
}
```

Задатак 3.

Написати на језику C++:

а) дефиницију класа на основу дијаграма на слици



- b) дефиниције конструктора класа, који иницијализују сваки атрибут на празан стринг или вредност 0, дефиниције метода основне класе као виртуелних, а предефинисаних у изведеним класама, тако да читају са тастатуре вредности тражених атрибута и уписују у тражену датотеку вредности одговарајућих атрибута класа,
- c) главну функцију која нуди опције 1. Student / 2. Predavac / 3. Крај, у петљи од 10 итерација, креира динамички објекте одговарајућих класа (у свакој итерацији у зависности од изабране опције) и позива над креираним објектима методе, да упишу у датотеку podaci.txt податке за сваког задатог студента и предавача.

Решење 3.

```
#include <iostream>
```

```
#include <fstream>
```

```
#include <string.h>
```

```
using namespace std;
```

```
class osoba {
    char ime[81];
public:
    osoba() { strcpy(ime, ""); }
    virtual void citaj() { cout<<"Ime osobe? "; cin>>ws; cin.getline(ime, 81); }

    virtual void pisi(char *ime_dat)
    {
        fstream fout;
        fout.open(ime_dat, ios::app); if(!fout)exit(1);
        fout<<"\nIme: "<<ime<<" ";
        fout.close();
    }
};

class student : public osoba {
    int br_indeksa;
public:
    student() : osoba() { br_indeksa=0; }
    void citaj() { osoba::citaj(); cout<<"Broj ? "; cin>>ws; cin>>br_indeksa; }

    void pisi(char *ime_dat)
    {
        fstream fout;
        fout.open(ime_dat, ios::app); if(!fout)exit(1);
        osoba::pisi(ime_dat); fout<<"Broj indeksa: "<<br_indeksa<<endl;
        fout.close();
    }
};
```

```

class predavac : public osoba {
    char zvanje[81];
public:
    predavac() : osoba() { strcpy(zvanje, ""); }
    void citaj(){ osoba::citaj(); cout<<"Zvanje? ";    cin>>ws; cin.getline(zvanje, 81); }
    void pisi(char *ime_dat)
    {   fstream fout;
        fout.open(ime_dat, ios::app); if(!fout)exit(1);
        osoba::pisi(ime_dat); fout<<"Zvanje: "<<zvanje<<endl;
        fout.close();
    }
};
//Glavna funkcija: kreiranje objekata, realizacija programa
int main()
{   osoba *objekat[10];
    int i, n=0, izbor;

    cout<<"\n\nNapraviti izbor: ";
    while(1)
    {   cout <<"\n\nIzbor: 1.Student, 2.Predavac, 3.Izlaz ";
        cin >> izbor; if(izbor==3) break;

        objekat[n]=0;
        switch(izbor)                                // izbor i dinamicko kreiranje objekata
        {   case 1:
                objekat[n]=new student;
                if(objekat[n])                        //provera dinamicke dodele
                {   objekat[n]->citaj();
                    objekat[n++]->pisi("podaci.txt");
                }
                break;
            case 2:
                objekat[n]=new predavac;
                if(objekat[n])                        //provera dinamicke dodele
                {   objekat[n]->citaj();
                    objekat[n++]->pisi("podaci.txt");
                }
                break;
            default:
                break;
        }
        if(n==10)
            break;
    }
    for(int i=0; i<n; i++)
        delete objekat[i];                            //oslobadjanje dinamicki kreiranih
    objekata
    return 0;
}

```