

Решавање на проблеми со податоци од тип структура

Пр. да се напише програма со која се составува список од n ученици. Учениците да се внесат како структура составена од име, презиме, оценка и просек, а оценка е структура во која се внесуваат оценките на учениците добиени од 3 теста по програмски јазици. Да се подреди и испечати список на ученици подреден по просечната оценка во опаѓачки редослед. Внесување, подредување и печатење на податоци да се реализира со функции

```
#include <iostream>
#include <string>
using namespace std;
struct oценка
{
    int t1;
    int t2;
    int t3;
};

struct ucеник
{
    char име[50];
    char презиме[50];
    oценка ocена;
    float прос;
};

void vнеси(ucеник spisок[1000], int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cout<<"vнеси име"<<endl;
        cin>>spisок[i].име;
        cout<<"vнеси презиме"<<endl;
        cin>>spisок[i].презиме;
        cout<<"vнеси oценка 1"<<endl;
        cin>>spisок[i].ocена.t1;
        cout<<"vнеси oценка 2"<<endl;
        cin>>spisок[i].ocена.t2;
        cout<<"vнеси oценка 3"<<endl;
        cin>>spisок[i].ocена.t3;
        spisок[i].прос=(float)(spisок[i].ocена.t1+spisок[i].ocена.t2+spisок[i].ocена.t3)/3;
    }
}

void подреди(ucеник spisок[1000], int n)
{
    ucеник p;
```

```

    int i,j;
    for(i=0;i<n-1;i++)
    {
        for(j=i+1;j<n;j++)
        {
            if(spisok[i].pros<spisok[j].pros)
            {
                p=spisok[i];
                spisok[i]=spisok[j];
                spisok[j]=p;
            }
        }
    }
}

void pecati(ucenik spisok[1000], int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cout<<spisok[i].ime<<" "<<spisok[i].prezime<<" "<<spisok[i].pros<<endl;
    }
}

int main()
{
    int i,j,n;
    struct ucenik spisok[1000];
    struct ucenik p;
    cout << "vnesi broj na ucenici" << endl;
    cin>>n;
    vnesi(spisok, n);
    podredi(spisok, n);
    cout<<"podreden spisok "<<endl;
    pecati(spisok, n);
    return 0;
}

```

Пр. да се напише програма со која се составува список од n вработени. Вработените да се внесат како структура составена од име, презиме, работен стаж и плата. Платата е структура во која се внесуваат основата и коефициентот на сложеност на работното место. Платата се пресметува како основата помножена со коефициентот на сложеност и се додава работниот стаж помножен со 0.5. Работниот стаж се пресметува 2020-godinanavработuvanje. Да се направи мени за избор дали да се печати список на сите вработени подредени висина на плата во опаѓачки редослед или по години на стаж во растечки редослед. Внесување на податоци, пресметување и подредување да се реализира со функции

```
#include <iostream>
```

```

#include <string>
using namespace std;

struct plata
{
    float osnova;
    float koef;
};
struct vraboten
{
    char ime[50];
    char prezime[50];
    int s;
    plata prim;
    float plati;
};

void vnesi(vraboten spisok[1000], int n)
{
    int i,g;
    for(i=0;i<n;i++)
    {
        cout<<"vnesi ime "<<endl;
        cin>>spisok[i].ime;
        cout<<"vnesi prezime "<<endl;
        cin>>spisok[i].prezime;
        cout<<"vnesi god. na vrabotovanje "<<endl;
        cin>>g;
        cout<<"vnesi osnova na plata"<<endl;
        cin>>spisok[i].prim.osnova;
        cout<<"vnesi slozenost na rab. mesto "<<endl;
        cin>>spisok[i].prim.koef;
        spisok[i].s=2020-g;
        spisok[i].plati=spisok[i].prim.osnova*spisok[i].prim.koef+spisok[i].s*0.5;
    }
}

void podrediplata(vraboten spisok[1000], int n)
{
    vraboten p;
    int i,j;
    for(i=0;i<n-1;i++)
    {
        for(j=i+1;j<n;j++)
        {
            if(spisok[i].plati<spisok[j].plati)
            {
                p=spisok[i];
                spisok[i]=spisok[j];
            }
        }
    }
}

```

```

        spisok[j]=p;
    }
}
}

void pecatiplata(vraboten spisok[1000], int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cout<<spisok[i].ime<<" "<<spisok[i].prezime<<" "<<spisok[i].plati<<endl;
    }
}

void podredistaz(vraboten spisok[1000], int n)
{
    vraboten p;
    int i,j;
    for(i=0;i<n-1;i++)
    {
        for(j=i+1;j<n;j++)
        {
            if(spisok[i].s<spisok[j].s)
            {
                p=spisok[i];
                spisok[i]=spisok[j];
                spisok[j]=p;
            }
        }
    }
}

void pecatistaz(vraboten spisok[1000], int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cout<<spisok[i].ime<<" "<<spisok[i].prezime<<" "<<spisok[i].s<<endl;
    }
}

int main()
{
    int i,j,n,m,g;
    struct vraboten spisok[1000];
    struct vraboten p;

```

```

cout << "vnesi broj na vraboteni" << endl;
cin>>n;
    vnesi(spisok, n);

cout<<"vnesi 1 ako podreduvas po plata, 2 ako podreduvas po raboten staz"<<endl;
cin>>m;
switch(m)
{
    case 1:
        {
            cout<<"podreden spisok po plata"<<endl;
            podrediplata(spisok, n);
            pecatiplata(spisok, n);
        }
break;
    case 2:
        {
            cout<<"podreden spisok po godini na raboten staz"<<endl;
            podredistaz(spisok, n);
            pecatistaz(spisok, n);
        }

}
return 0;
}

```

Прашања поврзани со наставните единици може да се испраќаат на email: anetastojceska@gmail.com