

Vježbe 6. Nasljeđivanje:

Pr. 1a

```
#include <iostream.h>

class Zivotinja {
protected:
    int godine;
    float tezina;
public:
    Zivotinja(): godine(2), tezina (15) {
        cout<<"Konstruktor Zivotinja....\n";}
    Zivotinja(int god, float tez) : godine(god), tezina (tez) {
        cout<<"Konstruktor Zivotinja sa parametrima....\n";}
    ~Zivotinja() {
        cout<<"Destruktor Zivotinja ....\n";}
    void inicijal (int god, float tez){
        godine = god;
        tezina =tez;
    }
    int DajGod() {return godine; }
    float DajTez() { return tezina; }
};

class Pas : public Zivotinja {
private:
    char pedigre;
public:
    Pas(): pedigre('d'){
        cout<<"Konstruktor Psa....\n";}
    ~Pas() {
        cout<<"Destruktor Pas ....\n";}

    void UnosPedigre(char ped){ pedigre = ped;}
    char ImaPedigre () { return pedigre; }
    void Glasanje() { cout<< "Vau-vau, vau-vau";}
};

void main() {
    Zivotinja ziv(1,10);
    cout<<"\n Starost ... " << ziv.DajGod() << endl;
    cout<<"\n Tezina ... " << ziv.DajTez() << endl;

    Pas Nola;
    cout<<"\n Starost ... " << Nola.DajGod() << endl;
    cout<<"\n Tezina ... " << Nola.DajTez() << endl;
    cout<<" Pedigre ?"<< Nola.ImaPedigre()<< endl;
    cout<<"Ima li pas pedigre? (d/n)";
    char p;
    cin>> p;
    Nola.UnosPedigre(p);
    cout<<" Pedigre ?"<< Nola.ImaPedigre()<< endl;
}
}
```

Pr.1b - sa konstruktorom

```
#include <iostream.h>

class Zivotinja {
protected:
    int godine;
    float tezina;
public:
    Zivotinja(): godine(2), tezina (15) {
        cout<<"Konstruktor Zivotinja....\n";}
    Zivotinja(int god, float tez) : godine(god), tezina (tez) {
        cout<<"Konstruktor Zivotinja sa parametrima....\n";}
    ~Zivotinja() {
        cout<<"Destruktor Zivotinja ....\n";}

    void inicijal (int god, float tez){
        godine = god;
        tezina =tez;
    }
    int DajGod(){return godine; }
    float DajTez(){ return tezina; }
};

class Pas : public Zivotinja {
private:
    char pedigre;
public:
    Pas(): pedigre('d'){
        cout<<"Konstruktor Psa....\n";}

    Pas(int god, float tez): Zivotinja (god, tez), pedigre('n') {
        cout<<" Konstruktor Psa izveden iz zivotinje";
    }

    Pas(int god, char ped): Zivotinja (god, (float)god), pedigre(ped) {
        cout<<" Konstruktor Psa izveden iz zivotinje";
        tezina = 10;
    }

    Pas(int god, float tez, char ped): Zivotinja (god, tez), pedigre(ped){
        cout<<" Konstruktor Psa izveden iz zivotinje 2";
    }

    ~Pas() {
        cout<<"Destruktor Pas ....\n";}

    void UnosPedigre(char ped){ pedigre = ped;}
    char ImaPedigre () { return pedigre; }
    void Glasanje() { cout<< "Vau-vau, vau-vau";}
};

void main() {
```

```

Zivotinja ziv(1,10);
cout<<"\n Starost ... " << ziv.DajGod() << endl;
cout<<"\n Tezina ... " << ziv.DajTez() << endl;

Pas Nola;
cout<<"\n Starost ... " << Nola.DajGod() << endl;
cout<<"\n Tezina ... " << Nola.DajTez() << endl;
cout<<" Pedigre ?"<< Nola.ImaPedigre() << endl;
Nola.Glasanje();

Pas Nola2(2,3, 'd');
cout<<"\n2 Starost ... " << Nola2.DajGod() << endl;
cout<<"\n Tezina ... " << Nola2.DajTez() << endl;
cout<<" Pedigre ?"<< Nola2.ImaPedigre() << endl;
Nola2.Glasanje();
cout<< "\n kraj " << endl;

}

```

Pr. 1c – nadjačavanje

```

#include <iostream.h>

class Zivotinja {
protected:
    int godine;
    float tezina;
public:
    Zivotinja(): godine(2), tezina (15) {
        cout<<"Konstruktor Zivotinja...\n";}
    Zivotinja(int god, float tez) : godine(god), tezina (tez) {
        cout<<"Konstruktor Zivotinja sa parametrima...\n";}
    ~Zivotinja() {
        cout<<"Destruktor Zivotinja ....\n";}

    void inicijal (int god, float tez){
        godine = god;
        tezina =tez;
    }
    int DajGod() {return godine; }
    float DajTez() { return tezina; }
    void Glasanje() { cout<< "Životinja se glasa ?? ";}
};

class Pas : public Zivotinja {
private:
    char pedigre;
public:
    Pas(): pedigre('d'){
        cout<<"Konstruktor Psa...\n";}

    Pas(int god, float tez): Zivotinja (god, tez), pedigre('n') {
        cout<<" Konstruktor Psa izveden iz zivotinje";
    }
};

```

```

    }

    Pas(int god, char ped): Zivotinja (god, (float)god), pedigre(ped) {
        cout<<" Konstruktor Psa izveden iz zivotinje";
        tezina = 10;
    }

    Pas(int god, float tez, char ped): Zivotinja (god, tez), pedigre(ped){
        cout<<" Konstruktor Psa izveden iz zivotinje 2";
    }

    ~Pas() {
        cout<<"Destruktor Pas ....\n";}

    void UnosPedigre(char ped){ pedigre = ped;}
    char ImaPedigre () { return pedigre; }
    void Glasanje() { cout<< "Vau-vau, vau-vau";}
};

class Maca : public Pas{
public:
    void Glasanje() { cout<< "Miau-miau, miau-miau!!!";}
};

void main() {
    Zivotinja ziv(1,10);
    cout<<"\n Starost ... " << ziv.DajGod() << endl;
    cout<<"\n Tezina ... " << ziv.DajTez() << endl;
    ziv.Glasanje();
    cout<< endl;

    Pas Nola;
    cout<<"\n Starost ... " << Nola.DajGod() << endl;
    cout<<"\n Tezina ... " << Nola.DajTez() << endl;
    cout<<" Pedigre ?"<< Nola.ImaPedigre() << endl;
    Nola.Glasanje();

    Pas Nola2(2,3, 'd');
    cout<<"\n2 Starost ... " << Nola2.DajGod() << endl;
    cout<<"\n Tezina ... " << Nola2.DajTez() << endl;
    cout<<" Pedigre ?"<< Nola2.ImaPedigre() << endl;
    Nola2.Glasanje();
    cout<< "\n kraj " << endl;

    Maca Suzi;
    cout<<"\n Starost ... " << Suzi.DajGod() << endl;
    cout<<"\n Tezina ... " << Suzi.DajTez() << endl;
    cout<<" Pedigre ?"<< Suzi.ImaPedigre() << endl;
    Suzi.Glasanje();
    Suzi.Zivotinja::Glasanje(); //pristup funkciji u gl. klasi
    cout<< endl;
}

```

Pr. 2 – sa stringovima

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

```
using namespace std;
```

```
class Sportas {
private:
    int SerBroj;
    char Ime[30], Prezime[50];
    int Godine, Visina, Tezina;
public:
    Sportas(int SB = 0, char *i = "", char *p = "", int G = 0, int V = 0, int T = 0);
    Sportas(const Sportas &);
    void Unos();
    void Promjena();
    void Ispis();
    void Ispis(bool Zaglavlje);
};
```

```
class Tenisac : public Sportas {
private:
    int Pozicija;
    int OsvojenoTur;
public:
    void Unos();
    void Promjena();
    void Ispis();
};
```

```
// *** Funkcije klase Sportas
```

```
// ** Konstruktori
```

```
Sportas::Sportas(int SB, char *i, char *p, int G, int V, int T){
    SerBroj = SB;
    strncpy(Ime, i, 20);
    strncpy(Prezime, p, 50);
    Godine = G;
    Visina = V;
    Tezina = T;
}
```

```
Sportas::Sportas(const Sportas &cpSportas){
    SerBroj = cpSportas.SerBroj;
    strcpy(Ime, cpSportas.Ime);
    strcpy(Prezime, cpSportas.Prezime);
    Godine = cpSportas.Godine;
    Visina = cpSportas.Visina;
    Tezina = cpSportas.Tezina;
}
```

```
// **
```

```
void Sportas::Unos(){  
    cout << "Unesite serijski broj: ";  
    cin >> SerBroj;  
    cout << "Unesite ime: ";  
    cin >> Ime;  
    cout << "Unesite prezime: ";  
    cin >> Prezime;  
    cout << "Unesite godine: ";  
    cin >> Godine;  
    cout << "Visina: ";  
    cin >> Visina;  
    cout << "Tezina: ";  
    cin >> Tezina;  
}
```

```
void Sportas::Promjena()  
{  
    cout << "Unesite serijski broj (trenutno = " << SerBroj << ") : ";  
    cin >> SerBroj;  
    cout << "Unesite ime (trenutno = " << Ime << ") : ";  
    cin >> Ime;  
    cout << "Unesite prezime (trenutno = " << Prezime << ") : ";  
    cin >> Prezime;  
    cout << "Unesite godine (trenutno = " << Godine << ") : ";  
    cin >> Godine;  
    cout << "Visina (trenutno = " << Visina << ") : ";  
    cin >> Visina;  
    cout << "Tezina (trenutno = " << Tezina << ") : ";  
    cin >> Tezina;  
}
```

```
void Sportas::Ispis()  
{  
    cout << "Serijski broj: " << SerBroj << endl;  
    cout << "Ime: " << Ime << endl;  
    cout << "Prezime: " << Prezime << endl;  
    cout << "Godine: " << Godine << endl;  
    cout << "Visina: " << Visina << endl;  
    cout << "Tezina: " << Tezina << endl;  
}
```

```
void Sportas::Ispis(bool Zaglavlje)  
{  
    if (Zaglavlje)  
    {  
        cout << setw(8) << "SER.BR." << setw(20) << "IME" << setw(30) << "PREZIME"  
            << setw(7) << "GODINE" << setw(7) << "VISINA" << setw(7) <<  
"TEZINA" << endl;  
    }  
  
    cout << setw(8) << SerBroj << setw(20) << Ime << setw(30) << Prezime  
        << setw(7) << Godine << setw(7) << Visina << setw(7) << Tezina << endl;  
}
```

```

// ***

// ***** Funkcije klase Tenisac
void Tenisac::Unos()
{
    Sportas::Unos();
    cout << "Pozicija na rang listi: ";
    cin >> Pozicija;
    cout << "Broj osvojenih turnira: ";
    cin >> OsvojenoTur;
}

void Tenisac::Promjena()
{
    Sportas::Promjena();
    cout << "Pozicija na rang listi (trenutno = " << Pozicija << ") : ";
    cin >> Pozicija;
    cout << "Broj osvojenih turnira (trenutno = " << OsvojenoTur << ") : ";
    cin >> OsvojenoTur;
}

void Tenisac::Ispis()
{
    Sportas::Ispis();
    cout << "Pozicija na rang listi: " << Pozicija << endl;
    cout << "Broj osvojenih turnira: " << OsvojenoTur << endl;
}

// *****

// GLAVNI PROGRAM
int main()
{
    // Sportas

    // prvi sportas
    Sportas VarSportas1;
    VarSportas1.Ispis();
    VarSportas1.Unos();
    VarSportas1.Ispis();

    // drugi sportas
    Sportas VarSportas2(123);
    VarSportas2.Ispis();

    // treci sportas
    Sportas VarSportas3(1, "Janica", "Kostelic", 21, 180, 81);

    // cetvrti sportas
    Sportas VarSportas4(VarSportas1);
    VarSportas4.Promjena();

    // Ispis Svih sportasa

```

```
VarSportas1.Ispis(true);
VarSportas2.Ispis(false);
VarSportas3.Ispis(false);
VarSportas4.Ispis(false);

// Tenisac

// prvi tenisac
Tenisac VarTenisac1;
VarTenisac1.Unos();
VarTenisac1.Ispis();
VarTenisac1.Promjena();
VarTenisac1.Ispis();

return 0;
}
// *****
```