

Osnovi računarskih sistema

Beleške sa vežbi

Smer *Računarstvo i informatika*
Matematički fakultet, Beograd

Jelena Tomašević

November 28, 2005

Sadržaj

```
#include <iostream>
#include <string>

using namespace std;

// ako radimo sa pokazivacima, kao sto moramo,
// onda moramo da se staramo o brisanju nepotrebnih objekata

class Vozilo
{
public:
    virtual ~Vozilo()
    {}

    virtual string Vrsta() const = 0;
    virtual int BrojVrata() const = 0;
    virtual int BrojTockova() const = 0;
    virtual int BrojSedista() const = 0;
    virtual int CenaPranja() const = 0;
};

class Automobil : public Vozilo
{
public:
    string Vrsta() const
    { return "Automobil"; }

    int BrojVrata() const
    { return 4; }

    int BrojTockova() const
    { return 4; }

    int BrojSedista() const
    { return 4; }

    int CenaPranja() const
    { return 200; }
```

```
};
```

```
class Kamion : public Vozilo
{
public:
    string Vrsta() const
        { return "Kamion"; }

    int BrojVrata() const
        { return 2; }

    int BrojTockova() const
        { return 6; }

    int BrojSedista() const
        { return 3; }

    int CenaPranja() const
        { return 550; }
};
```

```
// klase hijerarhije Vozilo moraju se upotrebljavati
// ISKLJUCIVO putem pokazivaca ili referenci
```

```
class Perionica
{
public:
    Perionica()
        : prvoVozilo(0), prvoSlobodnoMesto(0), dnevniPazar(0)
        {}

    ~Perionica()
    {
        while( ImaVozilaURedu() )
            delete IzdvojiPrvoVozilo();
    }

    void DodajVoziloURed( Vozilo* v )
    {
        int narednoSlobodnoMesto = (prvoSlobodnoMesto + 1) % max_vozila;
        if( prvoVozilo == narednoSlobodnoMesto )
            cout << "Nema vise mesta, vozilo je otislo u drugu perionicu." << endl;
        else{
            red[prvoSlobodnoMesto] = v;
            prvoSlobodnoMesto = narednoSlobodnoMesto;
        }
    }
};
```

```
bool ImaVozilaURedu() const
{ return prvoVozilo != prvoSlobodnoMesto; }

Vozilo* IzdvojiPrvoVozilo()
{
    Vozilo* v = red[prvoVozilo];
    prvoVozilo = (prvoVozilo + 1) % max_vozila;
    return v;
}

void OperiPrvoVozilo()
{
    if( ImaVozilaURedu() ){
        Vozilo* v = IzdvojiPrvoVozilo();
        cout << "Na redu je jedan " << v->Vrsta() << "." << endl;
        cout << "Prvo peremo vrata, ima ih " << v->BrojVrata() << endl;
        cout << "Zatim prelazimo na tockove, ima ih " << v->BrojTockova() << endl;
        cout << "Sada su na redu sedista, njih " << v->BrojSedista() << endl;
        cout << "Gotovo, za sada." << endl;
        dnevniPazar += v->CenaPranja();
        delete v;
    }
}

int DnevniPazar() const
{ return dnevniPazar; }

private:
    static const int max_vozila = 300;
    Vozilo* red[max_vozila];
    int prvoVozilo;
    int prvoSlobodnoMesto;
    int dnevniPazar;
};

main(){
    Perionica kodZike;
    kodZike.DodajVoziloURed( new Automobil() );
    kodZike.DodajVoziloURed( new Kamion() );
    while( kodZike.ImaVozilaURedu() ){
        kodZike.OperiPrvoVozilo();
        cout << endl;
    }
    kodZike.DodajVoziloURed( new Automobil() );

    cout << "Danas je Zika zaradio " << kodZike.DnevniPazar() << " dinara." << endl;

    return 0;
}
```