

Objektno orijentirano programiranje V03 – 04



Određivanje objekata

Prilikom izrade objektno-orijentiranog rješenja potrebno je prvo analizirati danu situaciju kako bi prepoznali i identificirali objekte koji sudjeluju.

Nakon toga pristupamo izradi nacrtu objekata koje još nazivamo i klase (engl.class). Znači, klasa je samo nacrt objekta kojeg treba u nekom trenutku napraviti i koristiti.

Vrlo slično kako radimo nacrt (engl. *blueprint*) kuće, pa po tom nacrtu (nazovimo ga *ObičnaKuća*) možemo napraviti više kuća i svaku od njih koristiti. Svaka od kuća je onda *instanca klase ObičnaKuća*.

Definiranje klase

```
class ObicnaKuca is a kind of Kuca {  
    property Povrsina is a double  
    property BrojProzora is a integer  
    property BrojVrata is a integer  
    property Adresa is a string  
    method StartHeating() {  
        //instrukcije za paljenje grijanja  
    }  
    event DoorBellPressed() {  
        //instrukcije za zvonjenje  
    }  
}
```

Analiza i prepoznavanje klasa

Križić-Kružić je igra za dva igrača. Svaki igrač koristi ikonu koja ga predstavlja: križić \times ili kružić \circ .

Ploča na kojoj se igra je podijeljena u 9 područja (kvadratića) organiziranih u 3 redka po tri stupca. Svako područje može sadržavati najviše jednu ikonu.

Igrači naizmjenice povlače po jedan potez stavljanjem svoje ikone u točno jedno područje. Ukoliko se tri ikone istog tipa pojave horizontalno, okomito ili dijagonalno, igrač čija je to ikona je pobjednik.

Ukoliko su sva područja popunjena, a nema pobjednika, igra se proglašava nerješenom. Ukoliko netko od igrača odbije napraviti sljedeći potez, drugi igrač se smatra pobjednikom.

Kada se igra pokrene, sustav treba omogućiti naizmjenično igranje protivnika.

Analiza i prepoznavanje klasa

Identificiranje klasa iz projekta:

Jedan način kako to možete učiniti je da prepoznate sve imenice u tekstu projekta, ispišete ih u jednini, te izbacite one koje su redundantne, preopćenite, nejasne, nisu bitne za projekt ili su atributi nekog objekta.

Zadatak 1

- a) U programskom jeziku C#, napišite definiciju klase Window koja modelira prozor u 'windowsima' s svojstvima: Naslov, Oznaka, Koordinate (zbog iscrtavanja, npr. (x1, y1), (x2,y2)), Boja (prema RGB modelu), te svojstvo za status je li prozor aktivan ili ne.
- b) U Main()-u kreirajte jedan objekt tipa Window, postavite mu svojstva, te ispišite detalje na ekran.
- c) U klasu dodajte metodu koja vraća string koji sadrži naslov, oznaku, status, te koordinate (kao uređene parove).
- d) d) U klasu dodajte metodu Width() koja računa širinu prozora, te metodu Height() koja računa visinu prozora.

Zadatak 1

- e) U klasu dodajte konstruktor koji kao parametar uzima naziv, te u Main-u kreirajte još jedan objekt korištenjem tog konstruktora.
- f) U klasu dodajte konstruktor bez argumenata. Prozor se defaultno treba zvati „Window1”, a dimenzije su mu 80x25.
- g) U klasu dodajte konstruktor koji kao parametre uzima naziv, koordinate te boju
- h) U klasu dodajte metodu Area() koja računa površinu prozora
- i) U klasu dodajte metodu Perimeter() koja računa opseg prozora.
- j) U klasu dodajte metodu Draw() koja ‘crtá’ taj prozor na konzoli simulirajući pravi prozor

Zadatak 2

- a) Kreirajte klasu Student sa sljedećim poljima: Ime, Prezime, GodinaStudija, Prosjek. Razmislite o prikladnom podatkovom tipu za svako pojedino polje. (12)
- b) Prethodnoj klasi dodajte metodu koja vraća podatke o studentu. (12)
- c) Kreirajte konstruktor klase Student koji ima dva parametra tipa String(ime i prezime). Konstruktor treba inicijalizirati ime i prezime Studenta . (12)
- d) Klasi Student dodajte konstruktor bez argumenata. Prilikom kreiranja novog studenta bez navedenog imena i prezimena, student se inicijalno mora zvati Pero Perić. (12)

Zadatak 3

Razmislite kako bi implementirali razlomak.

Napišite program koji učitava 2 razlomka i ispisuje rezultat 4 osnovne matematičke operacije s njima. (12 –željeni ishodi učenja)