

Curs I

!!! Se poate veni și cu seria c la curs/laborator + nu se face prezență.

I Introducerea în OOP

- Deosebită OOP

Descompunem sistemul într-un set de obiecte.

Datele și structura algoritmului sunt separate în mai multe clase și structuri.

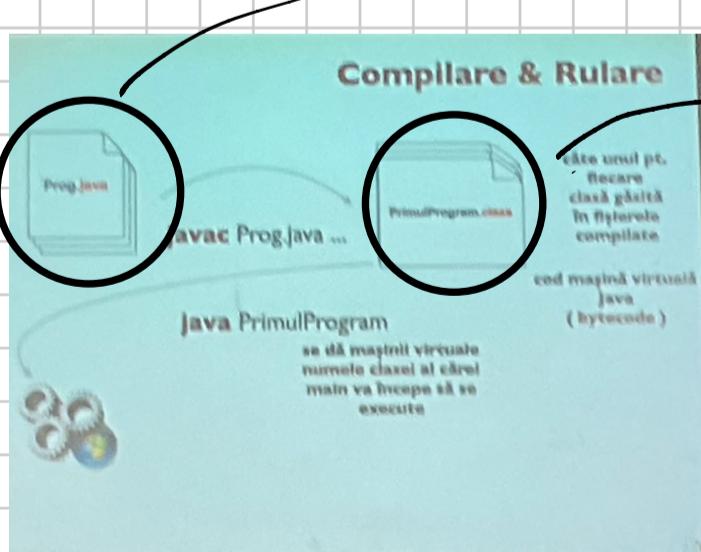
Def: Este o metodă de implementare a programelor în care acestea sunt organizate ca și colecții de obiecte.

Limbajul folosit este Java

- De ce Java?

Ește limbaj de top și mai simplu pentru introducerea în OOP.

Bytecode → cod maximă virtuală Java



program main

program secundar. Apelăm o clăsă din programul secundar în main.

Există tipuri de date primitive.

Ex: short int, long long, etc

Variabilele pot începe cu: literă, -, \$

Să declară variabile locale în clase, și toate valorile trebuie initializate.

int a;

System.out.println(a++);

System.out.println(a);

char c = "b";

System.out.println(c--);

System.out.println(c);

} \Rightarrow

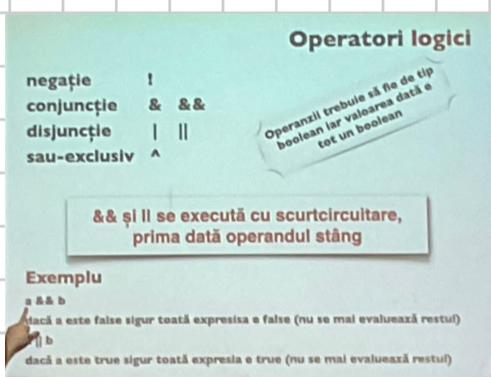
afisare

1
b
a

Dacă unul dintre operanți e string, automat și celalalt e convertit la string.

int a = 10;

System.out.println(a + "a"); \Rightarrow [11a]



Ex operator de atribuire

a = a + 2 \Rightarrow a += 2

a = 2 b = 10 c

c = a = b \Rightarrow c = 10 a = 10 b = 10

c = a += b \Rightarrow a = 17 b = 10 c = 17

La fiecare clase se pună sfârșit de execuție
"Clase și obiecte"

Un obiect este o instanță a unei clase.

O clasă definește un set de obiecte.

Clasa definește implementarea unui obiect.

class clock {

 private int ora, minut, secunda;

}

4

În Java nu trebuie eliberată memoria, de acasă se ocupa colectorul de deșeuri.

- Îndosarea operatorilor

Clock clasa1;

clasa1 = new Clock();

clasa1.setTime(10, 0, 0);

setTime este o funcție din clasă. Dacă funcția este pe public.

Variabile din clock pot fi modificate doar dacă sunt pe public.

Într-o clasă în fața fiecărei variabile nu este foarte să pui this.

casting -> prelucrarea datelor dintr-un obiect și punerile în alt obiect.

```
Clock clear1, clear2;  
clear1 = new Clock();  
clear2 = new Clock();  
clear1.setTime(10, 0, 0);  
clear2 = clear1; => aliasing
```

Mai multi pointere care referă la fel.

Constructor:

1. au același nume ca și clasa
2. nu au tip returnat
3. se folosește la început pentru a inițializa obiectul

Clock clear;

clear = new Clock();

Fără constructor

Clock clear;

clear = new Clock(10, 0, 0);

Cu constructor

1. Pentru a defini un constructor, acesta trebuie să aibă același nume ca și clasa.
2. Constructorul nu returnează nimic, tipul de date o să fie void.
3. Constructorul se folosește pentru a inițializa obiectul.

Cum se definește un constructor?

class student {

private int numar;

public void student(int nr) {

 numar = nr;