

Dodatne predefinisane operacije

```
1 ...  
int a = 10;  
String b = "AB";  
cwl(a+b);
```

ispis: 10AB

```
cwl(b+a);
```

ispis: AB10

↳ postoji operator koji prima String i int, bilo kojim redom, odnosno pri operaciji sabiranja Stringa i int-a, program će da int prebaci u String i da nalepi ova dva string-a onda.

```
2 ...  
int a = 20;  
String b = "CB";  
var c = a+b; // dakle c će biti String tipa sad do kraja izvršenja prog.  
cwl(c);
```

ispis: 20CB

Operatori minus (-), puta (*) i podeljeno (/) rade samo sa vrednosnim tipovima

```
PR.1 ...  
int a = 10; int b = 20;  
var c = a+b; // c je int  
cwl(c); // 30
```

↳ Dakle promenljiva c će zauzeti veću memoriju i dobiti vrednost klasične operacije koja je u pitanju

```
PR.2 ...  
int a = 10; long b = 20;  
var c = a+b; // c je long  
cwl(c); // 30
```

```
PR.2 ...  
int a = 10; double d = 20;  
var c = d/a; // c je double  
cwl(c); // 2
```

Neke nove cike sa rokova

1 public void Metoda(double a, int b) { ... }

poziv:

Metoda(b: 10, 5);

↳ Nije ispravan, kada se pošalje vrednost po nazivu za neki parametar (ovde je za b) onda svi naredni MORAJU isto po nazivu

↳ ispravno bi bilo Metoda(b: 10, a: 5);

2 enum Tip { ⁰ Biznis, ¹ Privatni }

Tip tip = Tip.Biznis; // isto što i index 0

tip--; // 0-- odnosno -1, u enumu

owl(tip); ne postoji ni jedan element sa ovim indexom
i zato je tip = -1 sad



ispis: -1

Dat je kod

int a;

main {

...

int b = 0; var c = 1;

...

}

Pitanje: Da li je a globalna promenljiva?

↳ Ako je u pitanju C++ jezik → ovde ne postoji koncept glob. prom.

↳ Ako je C i C++ → ovo je globalna promenljiva (svuda vidljiva)

↳ Ako je JAVA → statičke promenljive se posmatraju kao globalne

Obratiti pažnju kod zadatka sa metodama koje mogu biti kreirane u istoj klasi, ako metoda ima povratnu vrednost (npr. int) a ne vraća ništa, onda ona nije ispravno kreirana (važi i obrnuto).

SET PROPERTY i GET PROPERTY
ima jednu reč value jer postavlja vrednost promenljive

PR1 Class Osoba {

private String ime; // privatnoj promenljivoj ne možemo da pristupimo u main-u ako on nije u istoj klasi, ali ako imamo get i set metode (PROPERTY) onda možemo

GET public String ime {
get { return ime; } ili get => ime;
}

SET public String ime {
set { ime = value; }
}
}

VARIJACIJE SET PROPERTY-A

PR2. public String ime => ime = value; // GREŠKA X
↳ on sad očekuje neku promenljivu koja se zove value

PR3. public String ime(String value) => ime = value; // ISPRAVNO ✓