



CODE@FEIT

ШАС 2: РЕЛАЦИОНИ И ЛОГИЧКИ ОПЕРАТОРИ, IF/ELSE, SWITCH, ФУНКЦИИ



ТИПОВИ НА ПРОМЕНЛИВИ - ПРОШИРУВАЊЕ

Кастирање помеѓу `char` и `int`

- `char ch = (char)65.25; // декадно 65 се доделува на ch`
`System.out.println(ch); // ch е знакот A`
- `int i = (int)'A'; // Кодот на знакот A во Unicode табелата се доделува на i`
`System.out.println(i); // i сега е 65`

Сите аритметички операции може да се извршат над `char` променливи

`char` операнд автоматски се кастира во број, ако другиот операнд е број или знак

Ако другиот операнд е стринг, тогаш знакот се прилепува на стрингот



ТИПОВИ НА ПРОМЕНЛИВИ - ПРОШИРУВАЊЕ

```
int i = '2' + '3';           // (int)'2' e 50, a (int)'3' e 51
System.out.println("i e " + i); // i e 101
int j = 2 + 'a';           // (int)'a' e 97
System.out.println("j e " + j); // j e 99
System.out.println(j + " e Unicode za znakot" + (char)j);
System.out.println("Den" + '2');
```

Излез:

101

99

e Unicode za znakot c

en2



ТИПОВИ НА ПРОМЕНЛИВИ - ПРОШИРУВАЊЕ

При конкатенација на стрингови, ако некој од членовите не е стринг, автоматски е конвертиран во таков

Конкатенација на три стринга

```
String message = "Welcome " + "to " + "Java";
```

Стрингот `Chapter` се конкатенира со бројот `2`

```
String s = "Chapter" + 2; // s станува Chapter2
```

Стрингот `Supplement` се конкатенира со знакот `B`

```
String s1 = "Supplement" + 'B'; // s1 станува SupplementB
```



СПЕЦИЈАЛНИ ЗНАЦИ

`\t` таб

`\n` нов ред

`\\` коса црта

`'` единечни наводници

`"` двојни наводници

```
System.out.println("He said \"Java is fun\"");
```



ГРЕШКИ

```
public class ShowSyntaxErrors {  
public static void main(String[] args) {  
    i=30; //недекларирана променлива  
    System.out.println(i + 4);  
}
```

```
public class ShowRuntimeErrors {  
public static void main(String[] args) {  
    int i = 1/0 ; //не е дозволено делење со 0  
}
```



ГРЕШКИ

```
public class Test {  
    public void main(string[] args) {  
        int i;  
        int k = 100.0; //не може да се смести реална вредност во целобројна променлива  
        int j = i + 1;  
        System.out.println("j ima vrednost " + j + " a k ima vrednost" + k);  
    }  
}
```



КОЈ Е ИЗЛЕЗОТ ОД СЛЕДНИВЕ НАРЕДБИ?

```
System.out.println("1" + 1); //11
```

```
System.out.println('1' + 1); //50
```

```
System.out.println("1" + 1 + 1); //111
```

```
System.out.println("1" + (1 + 1)); //12
```

```
System.out.println('1' + 1 + 1); 52
```

Unicode вредноста на '1' е 49



РЕЛАЦИОНИ ОПЕРАТОРИ

>	поголемо
>=	поголемо и еднакво
<	помало
<=	помало и еднакво
==	еднакво
!=	различно

Внимавајте, `==` е оператор за споредба, додека `=` е оператор за доделување на вредност

Релациите враќаат само две **boolean** вредности, точно (1, **true**) или неточно (0, **false**)

`1==2` `//0`-грешно, но

`1!=2` `//1`-точно



ЛОГИЧКИ ОПЕРАТОРИ

&& (логичко И)

|| (логичко ИЛИ)

! (негација)

$(\text{expr1}==1)\&\&(\text{expr2}==1)=1$, инаку 0

$(\text{expr1}==0)\|\|(\text{expr2}==0)=0$, инаку 1

$!0=1$, $!1=0$

Логичките оператори, секоја вредност различна од нула, ја сметаат за единица



ЛОГИЧКИ ОПЕРАТОРИ

Сите логички изрази се пресметуваат одлево надесно

Пресметувањето се врши се додека „не сме сигурни“ за вредноста на изразот

Пример:

- за $i=11$ при пресметување на изразот $(i < 10) \ \&\& \ (i > 5)$ ќе се пресмета само вредноста на изразот $(i < 10)$ и бидејќи истата е 0, пресметувањето на целиот израз ќе прекине



РЕЛАЦИОНИ И ЛОГИЧКИ ОПЕРАТОРИ

```
i = 5; j = 0; k = -1;  
lzraz = i && j || k;
```

```
lzraz=(5&&0)||-1
```

```
lzraz=(1&&0)||1
```

```
lzraz=0||1
```

```
lzraz=1
```

```
i = 5; k = 10;
```

```
System.out.println((i==4)&&(k=5), (i!=4)||k==5);
```

0

1



КОЈ Е ИЗЛЕЗОТ ОД СЛЕДНИВЕ НАРЕДБИ?

`(true) && (3 > 4)`

`!(x > 0) && (x > 0)`

`(x > 0) || (x < 0)`

`(x != 0) || (x == 0)`

`(x >= 0) || (x < 0)`

`(x != 1) == !(x == 1)`

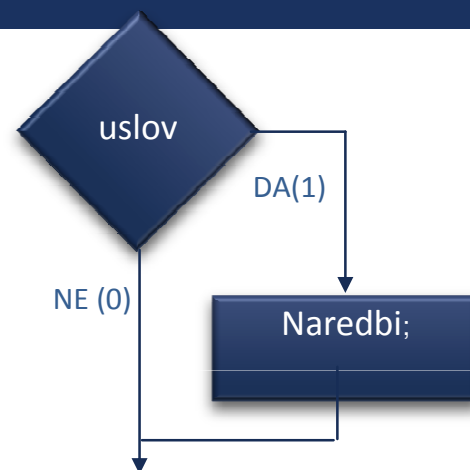
За $x=1$



F – СТРУКТУРА ЗА РАЗГРАНУВАЊЕ

(<logichki_uslov>)

naredbi...



Логичкиот услов мора да врати `boolean` вредност и мора да е затворен со мали загради

При извршување на повеќе наредби во рамки на условот користете големи загради `{ }`


После условот нема `;`



ВНИМАВАЈТЕ!

; после условот означува празна наредба (не прави ништо)

Наредбата после ; секогаш се извршува



```
if (i>0);  
System.out.println("Vnesen e pozitiven broj");
```

Влез: i= -8

Излез: Vnesen e pozitiven broj



F – СТРУКТУРА ЗА РАЗГРАНУВАЊЕ

```
public static void main(String[] args)
```

```
int x=5;
```

```
if (x > 5)
```

```
{
```

```
System.out.println(x " > " + "5");
```

```
}
```




F – СТРУКТУРА ЗА РАЗГРАНУВАЊЕ

```
if (x > 6) {
```

```
    if (x < 9) { ... }
```

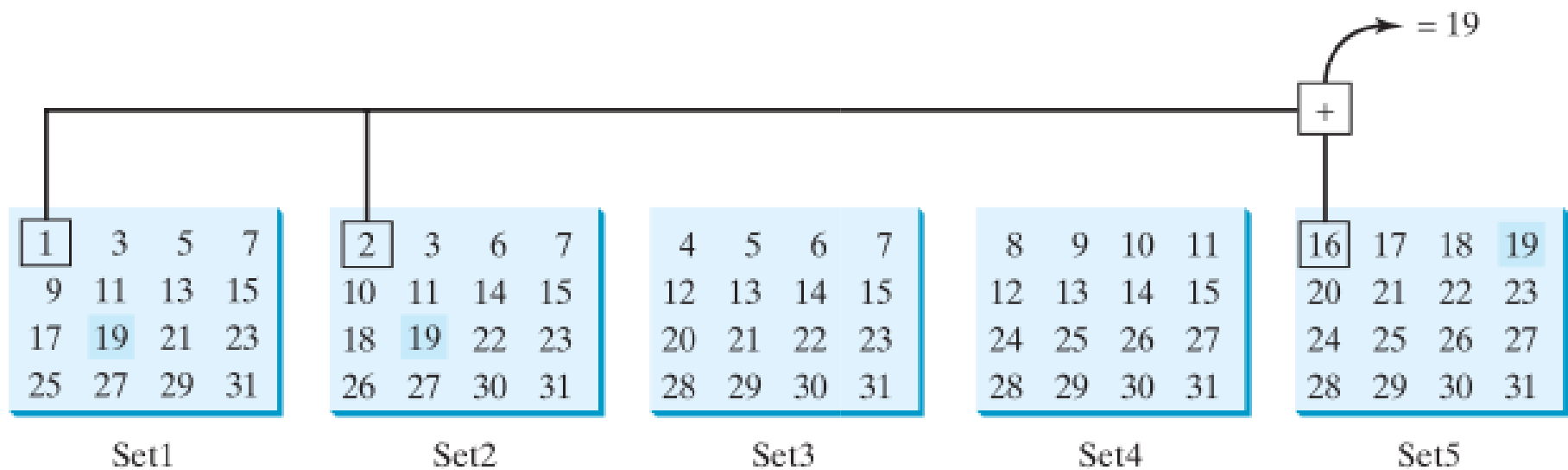
```
}
```

```
if ( x > 6 && x < 9) { ... }
```



ЗАДАЧА 1

Погодете го роденденот!





ЗАДАЧА 1

```
public static void main(String[] args) {  
    Definirame pet string promenlivi  
    String Mnozh1 =  
        "3 5 7\n" +  
        "11 13 15\n" +  
        "19 21 23\n" +  
        "27 29 31";  
    String Mnozh2 =  
        "3 6 7\n" +  
        "11 14 15\n" +  
        "19 22 23\n" +  
        "27 30 31";  
    String Mnozh3 =  
        "5 6 7\n" +  
        "13 14 15\n" +  
        "21 22 23\n" +  
        "29 30 31";  
}
```



ЗАДАЧА 1

string Mnozh4 =

8 9 10 11\n" +

2 13 14 15\n" +

24 25 26 27\n" +

28 29 30 31";

String Mnozh5 =

6 17 18 19\n" +

20 21 22 23\n" +

24 25 26 27\n" +

28 29 30 31";



ЗАДАЧА 1

```
int den=0;
```

```
Scanner input = new Scanner(System.in);
```

Barame od korisnikot da odgovori na slednive prashanja

```
System.out.print("Dali vashiot rodenden e vo Mnozha1?\n");
```

```
System.out.print(Mnozha1);
```

```
System.out.print("\nVnesete 0 za NE i 1 za DA: ");
```

```
int odgovor = input.nextInt();
```

```
if(odgovor==1)
```

```
den += 1;
```



ЗАДАЧА 1

```
System.out.print("Dali vashiot rodenden e vo Mnozh2?\n");
```

```
System.out.print(Mnozh2);
```

```
System.out.print("\nVnsetete 0 za NE i 1 za DA: ");
```

```
odgovor = input.nextInt();
```

```
if(odgovor==1)
```

```
den += 2;
```

```
System.out.print("Dali vashiot rodenden e vo Mnozh3?\n");
```

```
System.out.print(Mnozh3);
```

```
System.out.print("\nVnsetete 0 za NE i 1 za DA: ");
```

```
odgovor = input.nextInt();
```

```
if(odgovor==1)
```

```
den += 4;
```



ЗАДАЧА 1

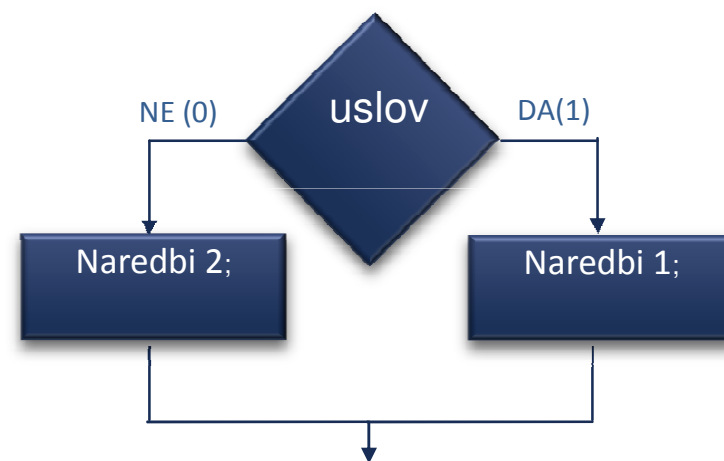
```
System.out.print("Dali vashiot rodenden e vo Mnozh4?\n");
System.out.print(Mnozh4);
System.out.print("\nVnsete 0 za NE i 1 za DA: ");
odgovor = input.nextInt();
if(odgovor==1)
    den += 8;
System.out.print("Dali vashiot rodenden e vo Mnozh5?\n");
System.out.print(Mnozh5);
System.out.print("\nVnsete 0 za NE i 1 za DA: ");
odgovor = input.nextInt();
if(odgovor==1)
    den += 16;
System.out.println("Datumot na vashiot rodenden e "+den);
```



F...ELSE – СТРУКТУРА ЗА ИЗБОР ОД ДВЕ МОЖНОСТИ

```
public static void main(String[] args)
```

```
int x=5;  
if (x >= 0) {  
    System.out.println(x + " e pozitiven broj");  
}  
else  
    System.out.println(x + " e negativen broj");
```



Наредбите во else делот се извршуваат доколку условот во if не е точен


При извршување на повеќе наредби во рамки на условот користете големи загради { }




ВГНЕЗДЕНИ IF...ELSE СТРУКТУРИ

`else` се поврзува со најблиското `if` над него

```
int i=100;  
if (i>0)  
    if (i>1000)  
        System.out.println("Vnesen e golem pozitiven broj");  
    else  
        System.out.println("Vnesen e mal pozitiven broj");
```



```
int i=100;  
if (i>0)  
    {if (i>1000)  
        System.out.println("Vnesen e golem pozitiven broj\n");}  
    else  
        System.out.println("Vnesen e negativen broj\n");
```





ПРЕШКИ

```
if (radius >= 0)
    ploshtina= radius * radius * 3.14;
system.out.println("Ploshtinata e" + ploshtina);
if (radius >= 0);
    ploshtina = radius * radius * 3.14;
System.out.println(" Ploshtinata e" + ploshtina);
```



ВНИМАВАЈТЕ!

= VS. ==

```
if (dolg = 0)
    System.out.println("Ne dolzhite nishto. ");
else
    System.out.println("Dolzhite" + dolg + "denari.
");
```

Влез: dolg=0
Излез: Dolzhite 0 denari.

Скратени наредби

if (izraz)

if(izraz!=0)

if (!izraz)

if(izraz==0)



КОЈ Е ИЗЛЕЗОТ ОД СЛЕДНИВЕ НАРЕДБИ?

```
if (x > 2) {  
    if (y > 2) {  
        z = x + y;  
        System.out.println("z e " + z);  
    }  
}
```

```
else  
    System.out.println("x e " + x);
```

x=3, y=2

x=3, y=4

x=2, y=2



ДАЛИ СЛЕДНИВЕ НАРЕДБИ СЕ ЕКВИВАЛЕНТНИ?

```
if (income <= 10000)
    tax = income * 0.1;
else if (income <= 20000)
    tax = 1000 +
        (income - 10000) * 0.15;
```

```
if (income <= 10000)
    tax = income * 0.1;
else if (income > 10000 &&
        income <= 20000)
    tax = 1000 +
        (income - 10000) * 0.15;
```



ЗАДАЧА 2

Што ќе отпечати следниот програмски сегмент?

```
int x = 3;
```

```
if(x
```

```
    System.out.println("DA");
```

```
else
```

```
    System.out.println("NE");
```



ЗАДАЧА 3

Под кои услови ќе се отпечати зборот *Voda* во следниот програмски сегмент?

```
if (T < 0)
    System.out.println("Mraz");
else if (T < 100)
    System.out.println("Voda");
else
    System.out.println("Parea");
```



ЗАДАЧА 4

Да се напише програма што врши квантификација на внесениот цел број: се внесува цел број (x), се проверува и се печати соодветниот текст што го опишува бројот, според следната табела

ако x е поголем или еднаков на 1000	се печати "претерано позитивен"
ако x е помеѓу 999 и 100 (вклучувајќи 100)	се печати "многу позитивен"
ако x е помеѓу 100 и 0 (без 0)	се печати "позитивен"
ако x е еднаков на 0	се печати "нула"
ако x е помеѓу 0 и -100	се печати "негативен"
ако x е помеѓу -100 и -999 (вклучувајќи -100)	се печати "многу негативен"
ако x е помал или еднаков на -1000	се печати "претерано негативен"



ЗАДАЧА 4

```
public static void main(String[] args)
```

```
    int i;  
    Scanner input = new Scanner(System.in);  
    System.out.print("Vnesete cel broj: \n");  
    i = input.nextInt();  
    if (i >= 1000 || i <= -1000 )  
        System.out.print("preterano ");  
    else if (i >= 100 || i <= -100)  
        System.out.print("mnogu ");  
    if (i > 0)  
        System.out.print("pozitiven\n");  
    else if (i == 0)  
        System.out.print("nula\n");  
    else if (i < 0)  
        System.out.print("negativen\n");  
}
```



ЗАДАЧА 5

Од тастатура се внесуваат координати на една точка. Да се напише програма што ќе одреди од кој квадрант е внесената точка или ќе даде информација доколку се работи за точка од оските



ЗАДАЧА 5

```
public static void main(String[] args) {  
  
    float x,y;  
    Scanner input=new Scanner(System.in);  
    System.out.println("Vnesete kootdinati x i y");  
    x=input.nextFloat();  
    y=input.nextFloat();  
    if(x > 0)  
    {  
        if(y > 0)  
            System.out.println("Prv Kvadrant");  
        else if(y < 0)  
            System.out.println("Cetvrt kvadrant");  
        else System.out.println("Poz. x oska"); //y==0  
    }  
}
```



ЗАДАЧА 5

```
else if(x < 0)
{
    if(y > 0)
        System.out.println("Vtor kvadrant");
    else if(y < 0)
        System.out.println("Tret kvadrant");
    else System.out.println("Neg. x oskan"); //y==0
}
```



ЗАДАЧА 5

```
se //x==0
{
    if(y > 0)
        System.out.println("Poz. y oska");
    else if(y < 0)
        System.out.println("Neg. y oska");
    else System.out.println("Koordinaten pochetok"); //y==0
}
} //main
```



SWITCH

Структурата switch има слична употреба како низа на вгнездени if/else наредби. Општиот облик на структурата е следниот:

```
switch ( izraz ) {
```

```
    e konstanta1 :
```

```
        edbi1;
```

```
    break ;
```

```
    e konstanta2 :
```

```
        edbi2;
```

```
    break ;
```

```
    default:
```

```
        edbid;
```

```
    break ;
```



SWITCH

Со оваа структура, најпрво, се разрешува изразот напишан во заградите веднаш по зборчето **switch**, и потоа се прескокнува кон соодветната **case** константа (лабела) чија вредност е еднаква на вредноста на изразот.

Вредноста на изразот *izraz* мора да биде цел број (`int`), знак (`char`) или енумерички тип.

Не е дозволено да постојат дупли **case** константи.

Константите (лабелите) во **case**, може да бидат наредени по кој било редослед, но мора да бидат константи.

Доколку во **case** константите не постои вредност еднаква со вредноста на изразот, се извршуваат наредбите по лабелата **default**.

Break овозможува **завршување** на наредбата **switch** и продолжување со следната наредба од програмата по неа. Затоа, **break** не смее да се заборави, освен доколку програмерот, има таква намера.



SWITCH

```
control = 0;  
/* bash primer na programerska praksa */  
switch (control) {  
    case 0:  
        System.out.println("Reset\n");  
    case 1:  
        System.out.println("Initializing\n");  
        break;  
    case 2:  
        System.out.println("Working\n");  
}
```




КОЈ Е ИЗЛЕЗОТ?

```
int x = 3, y = 3;  
switch (x + 3) {  
    case 6: y = 1;  
    default: y += 1;  
  
    System.out.println(y);
```



НАПИШЕТЕ ГО СЛЕДНИОВ КОД СО SWITCH

```
(a == 1)
    x += 5;
case if (a == 2)
    x += 10;
case if (a == 3)
    x += 16;
case if (a == 4)
    x += 34;
```



ЗАДАЧА 6

Да се напише програма што ќе работи како едноставен калкулатор: од тастатура ќе чита оператор и два броја и ќе ја изврши наведената операција.



ЗАДАЧА 6

```
public static void main(String[] args) throws IOException {
```

```
    float br1,br2, rezultat;
```

```
    char op;
```

```
    Scanner input=new Scanner(System.in);
```

```
    System.out.println("Vnesete dva broja");
```

```
    br1=input.nextFloat();
```

```
    br2=input.nextFloat();
```

```
    System.out.println("Vnesete operator");
```

```
    op=(char)System.in.read();
```



ЗАДАЧА 6

```
switch (op) {  
    case '+':  
        rezultat = br1 + br2;  
        System.out.println("Rezultatot od operacijata " + br1+op+br2+"e "+ rezultat);  
        break;  
    case '-':  
        rezultat = br1 - br2;  
        System.out.println("Rezultatot od operacijata " + br1+op+br2+"e "+ rezultat);  
        break;  
    case '*':  
        rezultat = br1 * br2;  
        System.out.println("Rezultatot od operacijata " + br1+op+br2+"e "+ rezultat);  
        break;
```



ЗАДАЧА 6

```
    e '/':
    br2 == 0) {
    stem.out.print("Greshka: Delenje so nula. ");
    stem.out.println("Operacijata ke se ignorira");

    e {
    zultat = br1 / br2;
    ystem.out.println("Rezultatot od operacijata " + br1+op+br2+"e " + rezultat);
    }
    ak;
    ault:
    stem.out.println("Nepoznat operator " + op);
    ak;
    witch
    main
```



ФУНКЦИИ (МЕТОДИ)

Функција: дел од програмата кој извршува некаква задача и е одделена во посебна целина

Зошто функции?

Основна функција во Java е main()

```
public static void main(String[] args)
```

```
    System.out.println("Dobredojdovte na FEIT!");
```



ФУНКЦИИ (МЕТОДИ)

Дефинирање на функција:

`<tip_rezultat> <ime_funkcija> (<argumenti>)`

telo na funkcija

Повик на функција:

`<ime_funkcija>(<argumenti>)`



ФУНКЦИИ (МЕТОДИ)

Функции за печатење на екран (`print()`, `println()`) :

```
String ime = "Petko ", prezime = "Petkov";
```

```
System.out.print( ime ); //курсорот останува на истата линија
```

```
System.out.println( prezime ); //курсорот се поместува во нова линија
```



ФУНКЦИИ (МЕТОДИ)

```
public class Funkcii {  
    public static void novRed()                //дефинирање на функција novRed()  
    {  
        System.out.println("");  
    }  
    public static void triNoviRedovi()        //дефинирање на функција triNoviRedovi()  
    {  
        novRed();    novRed();    novRed();    //три пати се повикува функцијата novRed()  
    }  
    public static void main(String[] args) {  
        System.out.print("Zbor 1");  
        triNoviRedovi();                    //повик на функцијата triNoviRedovi()  
        System.out.println("Zbor 2");  
    }  
}
```



ПАРАМЕТРИ НА ФУНКЦИЈА

Дефинирање:

```
public static tip_funkcija ime_funkcija(tip_param1 ime_param1, tip_param2  
ime_param2)
```

```
//наредби
```

Повикување:

```
ime_funkcija(ime_param1, ime_param2)
```



ЗАДАЧА 7: ПРЕСМЕТУВАЊЕ НА КВАДРАТ НА ВРЕДНОСТ

```
public static void печатиКвадрат(int x) //дефинирање на функција со еден аргумент
{
    System.out.println(x*x);
}

public static void main(String[] args) {
    int vrednost=3;
    печатиКвадрат(vrednost); //повик на функција со еден аргумент
    печатиКвадрат(5);
    печатиКвадрат(2*vrednost);
    печатиКвадрат("tri"); //која е грешката во оваа наредба?
}
```



ЗАДАЧА 8: ПРЕСМЕТУВАЊЕ НА ПРОИЗВОД

```
public static void pechatiproizvod(int x, int y) //дефиниција на функција со две променливи
```

```
{
```

```
    System.out.println(x*y);
```

```
}
```

```
public static void main(String[] args) {
```

```
    int vred1=3, vred2=5;
```

```
    pechatiproizvod(vred1,vred2); //повик на функција со две променливи
```

```
    pechatiproizvod(5,10);
```

```
}
```