

Branje v programčkih

DIRI 2003 – Programski jeziki

Branje v metodi paint ni dobra ideja!

```
import java.awt.*;
import java.applet.*;
import javax.swing.*;

public class Vsota_dveh extends Applet
{
    public void paint(Graphics g)
    {
        int a, b;
        int vsota;
        String a_s, b_s;

        // Preberimo stevili
        a_s = JOptionPane.showInputDialog("Prvo stevilo ");
        a = Integer.parseInt(a_s);
        b_s = JOptionPane.showInputDialog("Drugo stevilo ");
        b = Integer.parseInt(b_s);

        vsota = a + b;

        g.drawString("Vsota " + a + " in " + b + " je " + vsota, 20, 20);
    }
}
```

Spremenimo okno v katerem se izvaja programček!

Maija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Programčki – novo "okostje"

```
import java.applet.*;
import java.awt.*;

public class MojRazred extends Applet
{
    public void init()
    {
        // Kaj naj se zgodi ob zagonu
    }
    public void paint(Graphics okno)
    {
        // Kaj naj se prikaže
    }
}
```

Maija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Vrstni red izvajanja

- Najprej `init`, nato `paint`
- `init` se izvede le enkrat
- `paint` se izvede vedno, ko se na površini, kjer je grafično območje programčka kaj zgodi
- skupne spremenljivke
 - Tiste, ki jih potrebuje več metod
 - Deklariramo izven metod – v razredu
 - Spremenljivke, deklarirane v metodi, lahko uporablja le ta metoda

Maija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Dostop do spremenljivk

```
class xxx ....
{
    "razredne" spremenljivke
    public void init()
    {
        spremenljivke metode init
        ...
    }
    public void paint(Graphics g)
    {
        spremenljivke metode paint
        ...
    }
}
```

- V metodi `paint` lahko uporabljamo razredne spremenljivke in spremenljivke, deklarirane v metodi `paint`
- V metodi `init` lahko uporabljamo razredne spremenljivke, spremenljivko `g` in spremenljivke, deklarirane v metodi `init`
- Metoda `paint` "ne pozna" spremenljivk metode `init` in obratno
- Načeloma imamo lahko znotraj metode `init` enako poimenovane spremenljivke kot v `paint` – so pa to povsem ločene spremenljivke in nimajo nobene povezave druga z drugo!

Maija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Dva kroga drug v drugem

- Znotraj rdečega kroga bi radi narisali manjši moder krog
- Skupno središče
- V metodi `init` preberemo oba radija
- V metodi `paint` kroga narišemo
- Radija sta potrebna v obeh metodah
 - skupni spremenljivki
 - deklariramo v razredu
- Določiti je potrebno levi zgornji kot obeh krogov

Maija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Dva kroga drug v drugem

```
import java.awt.*;
import java.applet.*;
import javax.swing.*;

public class DvaKroga extends Applet
{
    int r_manjsi, r_vecji;

    public void init()
    {
        String a_s, b_s;

        // Preberimo radija

        a_s = JOptionPane.showInputDialog("Radij manjšega kroga ");
        r_manjsi = Integer.parseInt(a_s);
```

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Dva kroga drug v drugem

```
        b_s = JOptionPane.showInputDialog("Radij večjega kroga ");
        r_vecji = Integer.parseInt(b_s);
    }

    public void paint(Graphics g)
    {
        g.setColor(Color.red);
        g.fillOval(0, 0, 2 * r_vecji, 2 * r_vecji);
        g.setColor(Color.blue);
        g.fillOval(r_vecji - r_manjsi, r_vecji - r_manjsi,
            2 * r_manjsi, 2 * r_manjsi);
    }
}
```

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Dostop do spremenljivk

- Spremenljivki `a_s` in `b_s` sta potrebni le v metodi `init`.
- V metodi `paint` ti dve spremenljivki sploh NE obstajata (ju ne moremo uporabiti!)

- Pokažimo, da se `paint` res izvede vedno ob "obnovi" okna
 - Barva kroga naj bo naključna
 - [DvaKrogaBarva.java](#)

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Zgled - čas

- Sestavimo programček, ki mu podamo čas izmerjen v sekundah. Pretvorimo ga v dneve, ure, minute in sekunde
- $21344 \text{ sekund} = 355 \text{ min } 44 \text{ sek} = 5 \text{ h } 55 \text{ min } 44 \text{ sek} = 0 \text{ dni } 5 \text{ h } 55 \text{ min } 44 \text{ sek}$
- Če čas delimo s 60 – dobimo minute
- Ostanek pri deljenju s 60 so "prave" sekunde
- Minute sedaj delimo s 60 in dobimo ure
- Ostanek pri deljenju s 60 so "prave" minute
- Ure delimo s 24 in dobimo dneve
- Ostanek pri deljenju s 24 so "prave" ure

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Zgled - čas

- `cas = Integer.parseInt(cas_v_sek);`
- `sekunde = cas % 60; // "prave" sekunde`
- `cas = cas / 60; // cas v minutah`
- `minute = cas % 60;`
- `cas = cas / 60; // cas v urah`
- `ure = cas % 24;`
- `dnevi = cas / 24;`

- Skupne spremenljivke (tiste, potrebne za izpis!):
 - `sekunde, minute, ure, dnevi, cas_v_sek`

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Zgled – čas - program

```
import java.applet.*;
import java.awt.*;
import javax.swing.*;

public class Cas extends Applet
{
    String cas_v_sek;
    int sekunde, minute, ure, dnevi;
```

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Zgled – čas - program

```
public void init()
{
    int cas;

    cas_v_sek = JOptionPane.showInputDialog("Vnesi čas v
sekundah: ");
    cas = Integer.parseInt(cas_v_sek);
    sekunde = cas % 60; // "prave" sekunde
    cas = cas / 60; // cas v minutah
    minute = cas % 60;
    cas = cas / 60; // cas v urah
    ure = cas % 24;
    dnevi = cas / 24;
}
```

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003

Zgled – čas - program

```
public void paint(Graphics g)
{
    String izpis;

    izpis = cas_v_sek + " sekund = ";
    izpis = izpis + dnevi + " dni ";
    izpis = izpis + ure + " ur ";
    izpis = izpis + minute + " minut ";
    izpis = izpis + sekunde + " sekund ";
    g.drawString(izpis, 10, 10);
}
```

Marija Lokar,
Fakulteta za matematiko in fiziko

DIRI 2003
