

Java

Spremenljivke, prireditveni stavek

DIRI 2003 – Programski jeziki

Spremenljivke

- Prostor, kjer hranimo vrednosti
- Ime
 - Znak, številka, _
 - Presledkov v imenu ne sme biti!
- Tip spremenljivke
 - int (cela števila)
- Vse spremenljivke napovemo
 - `int sirina;`
 - `int starost, st_cevljev;`

Tip int

- int
- Cela števila
- Omejen obseg
 - +2 mrd : -2mrd
 - Sami moramo poskrbeti, da so rezultati operacij znotraj tega obsega
- Operacije: +, -, *, /, %
- `int x = 10;`
- V spremenljivki `x` hranimo cela števila in začetna vrednost je 10.

int - operacije

- +, -, * (seštevanje, odštevanje, množenje)
- $2 + 3 \rightarrow 5$
- $2 - 3 \rightarrow -1$
- $2 * 3 \rightarrow 6$
- Prioriteta operacij
 - $2 + 3 * 4 \rightarrow 14$
- / (celoštevilsko: $12 / 7 = 1$)
- % ostanek pri deljenju ($12 \% 7 = 5$)

Prireditveni stavek

- V spremenljivko shranimo vrednost
 - `x = 10;`
 - `starost = 25 + 2 * 8;`
- `ime_spremenljivke = izraz;`
- Izračuna se vrednost izraza. Dobljena vrednost se shrani v spremenljivko.
- Če spremenljivka nastopa v izrazu – vrednost, ki jo hranimo v spremenljivki
 - `x = 10; // v x smo shranili 10`
 - `y = 3 * x + 5; // izračunamo izraz: 3 krat število, ki // je shranjeno v x in to povečamo za // 5. Dobljeni rezultat shranimo v y.`
- `x = x + 1;`
- Vrednost shranjeno v x povečamo za 1!
- Zakaj:
 - Izračunamo izraz: Tisto, kar je shranjeno v x, povečamo za 1.
 - Dobljeni rezultat spet shranimo v x

Tip double

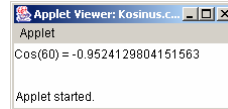
- Realna števila (decimalna števila)
- Decimalna pika.
- Nenatančnost
 - Dvojiški zapis
 - Končno število decimalk
 - 0.1 se ne da napisati točno v dvojiškem sistemu
- Operacije: `+`, `-`, `*`, `/`
- Funkcije
 - Razred `Math`
- `double x = 10.2;`
- V spremenljivki `x` hranimo decimalna števila in začetna vrednost je 10.2.

Standardne funkcije

- Kako izračunamo elementarne funkcije, kot so logaritem, eksponentna funkcija, sinus, kosinus, ipd.?
- Vse te funkcije so priskrbljene v standardni knjižnici, v razredu Math.

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

public class Kosinus extends Applet
{
    public void paint(Graphics g) {
        g.drawString("Cos(60) = " + Math.cos(60.0), 0, 15);
    }
}
```



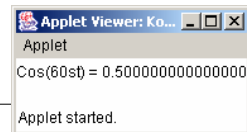
Razred Math

- Različne metode
 - sqrt (kv. koren)
 - abs (absolutna vrednost)
 - sin (sinus – argument v radianih)
 - Ostale si oglej npr. na <http://haka.fmf.uni-lj.si/java/docs/api/java/lang/Math.html>
- Konstanti Math.PI in Math.E
- Kličemo jih npr. z Math.sin(izraz)
- Math.abs(12-15) je torej 3.
- Rezultati (in argumenti) so tipa double!

Kotne funkcije

- `Math.sin`, `Math.cos`, `Math.tan`,
`Math.asin`, `Math.atan`
- Argumenti tipa `double` pomenijo kote v radianih
- `Math.toRadians` pretvori kot iz stopinj v radiane
- `Math.toDegrees` počne obratno

```
g.drawString("Cos(60st) = " +  
    Math.cos(Math.toRadians(60.0)), 0, 15);
```



Zgledi "problemov" - double

- Izračunajmo vrednosti izrazov $\sin(\pi/6)$, $\sqrt{1225}$
in $\ln(e)$.
 - `Math.sin(Math.PI / 6)`
 - `Math.sqrt(1225)`
 - `Math.log(Math.E)`
- 0.499999 namesto 0.5
- Zakaj 35.0 in ne 35?

Povzetek

- Glej npr.
 - <http://zaversnik.fmf.uni-lj.si/Gradiva/Java/teorija/spremenljivke.htm>
 - <http://haka.fmf.uni-lj.si/pra-racunalninstvo-1/lekcija02/index.html>
- Vsako spremenljivko je potrebno napovedati (le enkrat!)
- Vrednosti prirejamo s prireditvenim stavkom

Zgled

- Spremeni naslednja navodila v ukaze v Javi
 - Deklariraj celoštevilčno spremenljivko x z začetno vrednostjo 12.
 - Deklariraj celoštevilčno spremenljivko y z začetno vrednostjo -715.
 - Deklariraj celoštevilčno spremenljivko z z začetno vrednostjo 0.
 - Nastavi z na vsoto spremenljivk x in y .
 - Odštej 7 od x .
 - Nastavi y na produkt x in z .

Izpis vrednosti spremenljivke

- `System.out.println(x);`
- Izpiši vrednost izraza!
- Vrednost izraza `x` je vrednost spremenljivke `x`.
- `g.drawString(x, 10, 5);`

Zgled: Iz števila 38 naredimo 83!

- Shranimo število
`stevalo = 38;`
- Določimo enice
`enice = stevalo % 10;`
- Določimo desetice
`desetice = stevalo / 10;`
- Naredimo novo število
`novo_stevalo = enice * 10 + desetice;`

Zložimo v program/programček

- Okostje za program
 - main ...
 - SteviloP.java
- Okostje za programček
 - import, Applet, paint, ...
 - Stevilo.java
- Komentarji
- Izgled

Nova uporaba

- Kako iz 27 narediti 72?
- Le zamenjamo prireditveni stavek
- `stevilo = 27;`

Kvadrat s stranico a

- Naloga: Nariši kvadrat s stranico $a = 100$ in levim zgornjim ogliščem v točki $(10, 10)$
- Ideja:
 - Uporabimo metodo `drawRect`, kjer sta dolžina in višina enaka a
 - `g.drawRect(10, 10, a, a)`

Program

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

public class Kvadrat extends Applet
/*
  Narisali bomo kvadrat s stranico a
*/
{
    public void paint (Graphics g)
    {
        int a; // Stranica kvadrata

        a = 100; // V spremenljivko a shranimo celo stevilo 100
        // Narisemo kvadrat
        g.drawRect(10, 10, a, a);
    }
}
```

Kvadrat s stranico 70

- Spremeniti prireditev
 - `a = 70;`
- Dvakrat večji kvadrat
 - `a = 2 * a;`
 - V spremenljivko `a` shranimo dvakratno vrednost, kot jo je imela spremenljivka `a` prej
- [Kvadrat.java](#)
- Program?
 - [KvadratP.java](#)
 - S sedanjim znanjem ne gre!

Menjava vrednosti dveh spremenljivk

- Pogost opravek v programiranju je menjava vrednosti dveh spremenljivk.
- Denimo, da imamo deklarirani dve celoštevilski spremenljivki `x` in `y`. Programerji začetniki pogosto mislijo, da se njuni vrednosti zamenja takole:
 - `x = y;`
 - `y = x;`
- Kaj se zares zgodi, ko se izvedeta ta ukaza?
- Kako se zamenja vrednost dveh spremenljivk?
- Namig: uporabi tretjo, pomožno spremenljivko.
- <http://zaversnik.fmf.uni-lj.si/Gradiva/Java/primeri/Zamenjava/Zamenjava.htm>

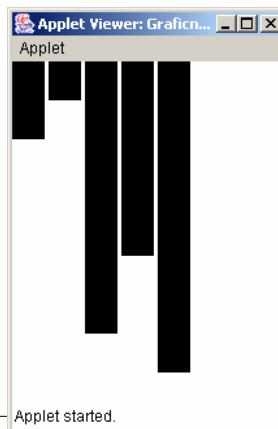
Poraba bencina

- Ko sem se zjutraj peljal v Ljubljano, sem moral natočiti gorivo
- Prevozil sem 712 km in natočil 52,2 l goriva
- Koliko je bila poraba v l / 100km?

- Spremenljivke
 - `prevozeni_km : int`
 - `litri, poraba : double`

Grafični prikaz števila

- Na sliki je prikaz števila 21647
- Vsaka številka – ustrezno velik stolpec



Postopek

- Predpostavke:
 - Število je celo in pozitivno
 - Je največ 5 mestno
- Postopek
 - Število razbijemo na enice, desetice, stotice, tisočice in desetisočice
 - Narišemo ustrezno velike pravokotnike

Grafični prikaz števila - števke

- Določitev števka števila
- Določimo zadnjo števko (enice)
 - $\% 10$
- "Odrežemo" enice
 - $/ 10$
 - V novem številu so "stare" desetice enice
- Ponovimo enak postopek

Grafični prikaz števila - številke

```
// določimo številke števila
enice = stevilo % 10;
stevilo = stevilo / 10;
desetice = stevilo % 10;
stevilo = stevilo / 10;
stotice = stevilo % 10;
stevilo = stevilo / 10;
tisocice = stevilo % 10;
dt = stevilo / 10;
```

Izris

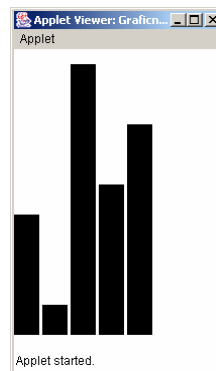
- Pravokotnik širok za `sirina` in visok za številko (enice, desetice, ...), pomnoženo z ustreznim faktorjem
 - `stevka * faktor`
 - `g.fillRect(kotx, koty, sirina, faktor * dt);`
- Spreminja se koordinata levega zgornjega kota pravokotnika, ki prikazuje številko
 - `kotx = kotx + sirina + razmik;`
- Koordinata `y` je ves čas enaka (denimo 10)

Izris

- Najprej narišemo desetisočice
- Spremenimo levi zgornji kot pravokotnika
- Narišemo tisočice
- ...
- StevkeGraficno.java

Grafični prikaz števila

- "Obrnimo" sliko
- Na sliki je prikaz števila 41957



Grafični prikaz

- Drugačne so samo y koordinate levega zgornjega kota
- 0 ima koordinate “spodnjega” roba
- Vsaj $9 * faktor + začetni\ y \rightarrow$ osnova
- Številka 3
 - Ustrezno zmanjšana (y koordinate naraščajo od zgoraj navzdol!)
 - $osnova - 3 * faktor$
- [StevkeGraficnoVar2.java](#)

Poraba bencina

- Izračunamo porabo
- $poraba = litri / prevozeni_km * 100;$
- Izpišemo rezultat

Poraba bencina - program

```
public class Poraba {
    public static void main(String[] g)
    {
        String izpis;
        double poraba;
        double litri = 52.2;
        int prevozeni_km = 712;

        poraba = litri / prevozeni_km * 100;
        izpis = "Če si prevozil " + prevozeni_km + "km in porabil ";
        izpis = izpis + litri + " goriva, je poraba " + poraba;
        izpis = izpis + " l na 100km";
        System.out.println(izpis);
    }
}
```

- String
- V teh spremenljivkah hranimo nize
- Stikanje nizov