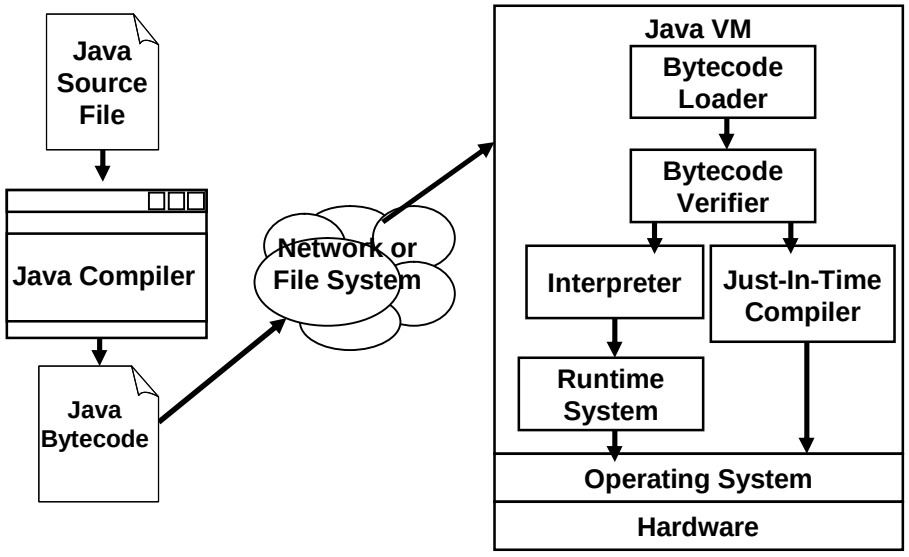
Ievads programmēšanas valodā Java

Java programmas sagatavošana



C++ un Java

- Java programmā viss ir klasēs. Nav globālu mainīgo un ārēju funkciju. Nav struct, union, enum - ir tikai klases.
- Standarttipi :

boolean	True/false
byte	8-bit signed
short	16-bit signed
char	16-bit Unicode
int	32-bit signed
long	64-bit signed
float	32-bit IEEE754
double	64-bit IEEE754

C++ un Java

```
class Point {
   double x;       // data member
   double y;       // data member
   public Point AddPoint(Point p) {
     x += p.x; y += p.y;
     return this;  // reference, not a pointer!
   }
}                  // no semicolon!
```

- Metožu definīcijas ir aprakstītas klasē. Nav preprocesora operatoru #include, tam līdzīga funkcionalitāti nodrošina import, kas pievieno bibliotēku.

```
import java.io;
class HelloWorld {
      static public void main(String args[]) {
           System.out.println("Hello, world.");
      }
}
```

C++ un Java

- Konstruktors pēc noklusēšanas tiek nodrošināts tāpat kā C++. Kopijas konstruktori netiek veidoti.
- **public**, **private** un **protected** jāraksta katram klases loceklim.
- Atvasinātā klase:

```
class MyPoint extends Point {
  // new stuff for MyPoint
  ...
}
```

- Visas Java klases ir atvasinātas, sākot no pamatklases Object.
- Visas metodes ir virtuālas (tāpēc nav atslēgvārda virtual).
- Lai noteiktu, ka metodi nevar aizvietot atvasinātās klases metode, to var aprakstīt kā **final**.

C++ un Java

- Operācija **new**

```
Point p = new Point();
```

- Operācijas new rezultāts ir reference (nevis rādītājs!).
- Rādītāju (pointer) nav!
- Nav operācijas delete. Objektus automātiski likvidē drazu savācēja process (*garbage collection)*.
- Destruktora lomu var pildīt metode **finalize()**. Atšķirībā no C++ destruktora, nav zināms laika moments, kad tā tiks izpildīta. Ja jāaizver faili u.tml., tad labāk veidot īpašu metodi, kuru izsauc tieši.
- Klašu objektus var radīt tikai ar new.

C++ un Java

- Masīvi ir organizēti atšķirīgi no C++ !

```
Point pts [];         // array of Points
pts = new Points[10]; // create array for 10 Points
```

- Kad rada masīvu, tajā nav elementu. Elementus jārada ar new un jāievieto masīvā.

```
int i;
for (i=0; i<10; i++) {
      pts[i]=new Point();
}
```


C++ un Java

- Nav **struct**
- Nav **union**
- Nav operatora **goto**
- Nav daudzkārtējās mantošanas
- Nav **inline** metodes
- Nav šablonu (**template**)
- Nav operāciju pārdefinēšanas