

Utblickar och sammenfattning

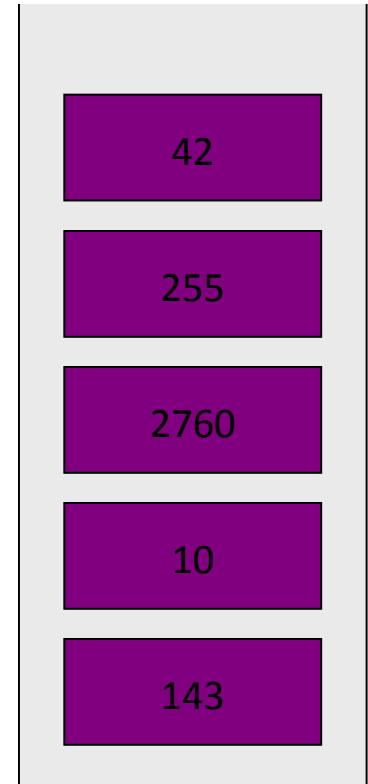
TDP007 Konstruktion av datorspråk
Föreläsning 8

Översikt

- Kompilering av kod i Java
- Olika språks popularitet
- Sammanfattning av kursen
- Planering av examination

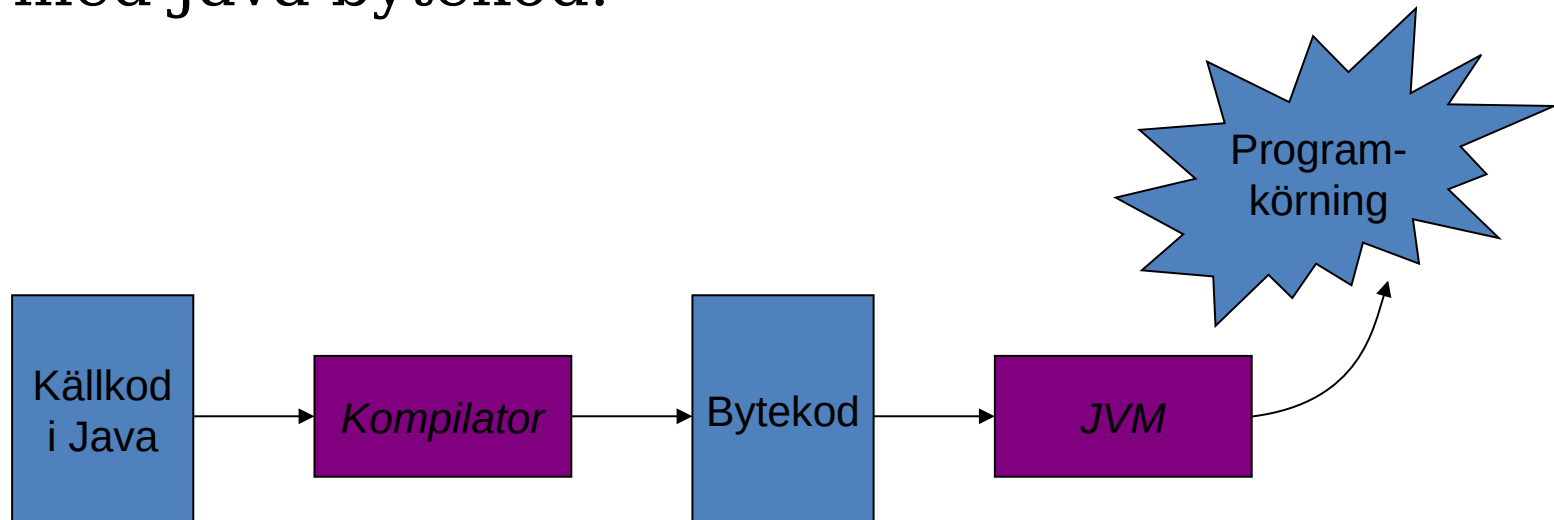
Stackmaskin

- En teoretisk beräkningsmodell där minnet i princip endast består av en stack
- Ett program som simulerar en sådan maskin



Java Virtual Machine (JVM)

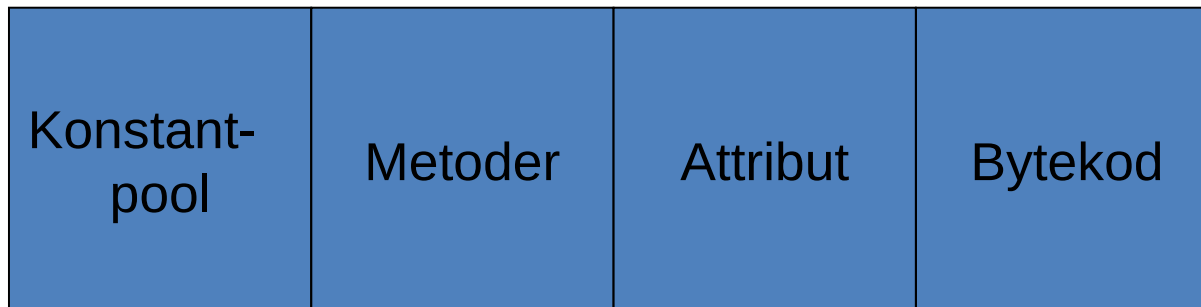
- JVM är ett slags utökad stackmaskin som kan sägas interpretera program representerade med Java bytekod.



Exempelkod i Java

```
public class Example {  
  
    public static void main(String[] args) {  
  
        outer:  
        for (int i=2; i<1000; i++) {  
            for (int j=2; j<i; j++) {  
                if (i % j == 0)  
                    continue outer;  
            }  
            System.out.println(i);  
        }  
    }  
}
```

Vad finns i bytekodfilen?



```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

Variabler



```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

2

Variabler


```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

Variabler

1 \leftarrow 2

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

2

Variabler

1 ← 2

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

1000
2

Variabler

1 ← 2

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

1000
2

Variabler

1 \leftarrow 2



```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

2

Variabler

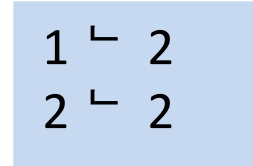
1 ← 2

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack



Variabler



```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

2

Variabler

1 \leftarrow 2

2 \leftarrow 2

```

00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return

```

Stack

2
2

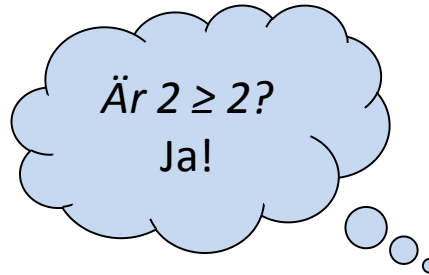
Variabler

1 \leftarrow 2
2 \leftarrow 2


```

00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return

```



Stack

```

2
2

```

Variabler

```

1 ← 2
2 ← 2

```

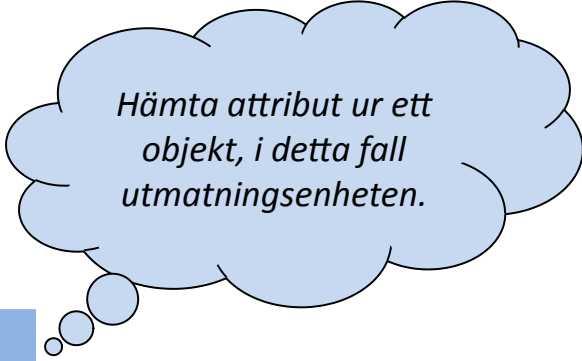
```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

out

Variabler

1 \leftarrow 2
2 \leftarrow 2



Hämta attribut ur ett
objekt, i detta fall
utmatningsenheten.

```

00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return

```

Stack

2
out

Variables

1 ← 2
2 ← 2

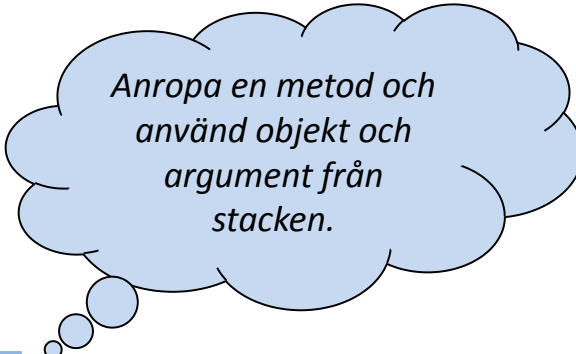
```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

2
out

Variabler

1 \leftarrow 2
2 \leftarrow 2



Anropa en metod och
använd objekt och
argument från
stacken.

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack



Variabler

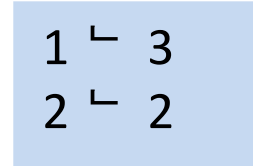
1	←	3
2	←	2

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack



Variabler



```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

3

Variabler

1 \leftarrow 3
2 \leftarrow 2

```
00: iconst_2
01: istore_1
02: iload_1
03: sipush 03e8
06: if_cmpge 2c
09: iconst_2
0a: istore_2
0b: iload_2
0c: iload_1
0d: if_cmpge 1f
10: iload_1
11: iload_2
12: irem
13: ifne 19
16: goto 26
19: iinc 2, 1
1c: goto 0b
1f: getstatic #2
22: iload_1
23: invokevirtual #3
26: iinc 1, 1
29: goto 02
2c: return
```

Stack

```
1000
3
```

Variabler

```
1 ← 3
2 ← 2
```

Och så vidare...

Kursens mål

- förklara och använda reguljära uttryck *seminarie 1-2
seminarie 2*
- använda verktyg för hantering av uppmärkningspråk (t.ex. XML) *seminarie 3*
- använda och modifiera en tolk för ett enklare programspråk *seminarie 3-4
samt opposition
och
utvecklarblogg*
- redogöra för och tillämpa grundläggande principer för design av programspråk

Olika typer av datorspråk

Dataspråk

Programspråk

reguljära uttryck *XML*

Domänspecifika språk

Generella språk

*Ruby-specifika
tekniker för DSL*

parsning

Traditionell beräkning

Alternativ beräkning

*icke-deterministisk
programmering*

*constraint-
nätverk*

Vilka språk är mest populära?

TIOBE Programming Community Index

- <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>

Varför finns det så många språk?

- Språk utvecklas hela tiden
- Speciella tillämpningar
- Tycke och smak

Varför blir ett språk populärt?

- Uttryckskraft
- Lätt att lära sig
- Standarder
- Öppen källkod
- Bra verktyg
- Stöd från företag och organisationer

Varför ska man studera språk?

- Bättre på att förstå hela språket
- Kan välja olika sätt att uttrycka sig
- Bättre på att använda hjälpverktyg
- Återskapa funktionalitet i språk som saknar den
- Återanvända språktekniker för andra ändamål

Planering av slutexamination 2017

- Dugga 2
 - Onsdag 02/3 kl 09.00-11.00
- Datortenta (Istället för dugga 1 och 2)
 - Fredag 24/3 kl 14.00-18.00
 - Senaste anmälan tisdag 14/3
 - *Kom i tid!*
 - *Ta med LiU-id!*
- Hemtenta (omexamination för inlämningar/seminarier)
 - Jag återkommer till berörda efter sista seminariet

Tack för ert deltagande i denna kurs

- Jag finns tillgänglig för diskussion av datorspråk (TDP019) eller för förberedelser inför sem4 (TDP007)

www.liu.se