

# Mål och innehåll

## Kursens lärandemål

Efter avslutad kurs ska studenten kunna:

**A**nysera, förklara, modifiera och utöka ett program bestående av fundamentala programkonstruktioner. **B**eskriva hur numeriska och teckenbaserade data representeras samt använda primitiva datatyper, teckensträngar och inbyggda datastrukturer.

**D**esigna, implementera, testa och felsöka ett program som använder grundläggande beräkning, enkel inmatning och utmatning, vanliga villkors- och iterationsstrukturer och funktionsdefinitioner.

**D**iskutera vikten av algoritmer, identifiera de nödvändiga egenskaperna hos goda algoritmer och skapa algoritmer för att lösa enkla problem.

**U**tveckla kod som hanterar undantagstillstånd under exekveringen.

**D**iskutera program med objekt och förstå begreppen inkapsling, arv, polymorfism och gränssnitt.

**D**esigna, implementera, testa och felsöka enkla program i ett objektorienterat programmeringsspråk.

## Kursens innehåll

### *Ett Javaprogram*

- Ett problem och en algoritm
- Ett Javaprograms struktur
- Kompilera och exekvera ett Javaprogram
- Utforma och kommentera Javakod
- Grundläggande utmatning
- Javaspråket och standardbiblioteket

### *Datalagring*

- Variabler
- Primitiva datatyper i Java
- Typomvandlingar vid lagringen
- Teckensträngar

### *Standardinmatning*

- En inmatningsström
- Mata in tecken
- Mata in teckensträngar
- Mata in tal

### *Operationer med primitiva värden*

- Operationer med heltal
- Operationer med flyttal
- Operationer på en variabel
- Jämförelse av primitiva värden
- Operationer med booleska värden

### *Logik*

- Villkorliga handlingar
- Upprepningar

### *Vektorer*

- Endimensionella vektorer
- Flerdimensionella vektorer

### *Metoder*

- Skapa och använda metoder
- Hantera metoder
- Vektorer som parametrar och returvärden
- Rekursiva metoder

### *Ett klassbibliotek*

- Distribuerad kod
- Javas standardbibliotek
- En methods undantag

### *Algoritmer*

- Ett problem och en algoritm
- Urvalsalgoritmer
- Sorteringsalgoritmer
- Sökningsalgoritmer
- En algoritms komplexitet
- En algoritms riktighet

### *Objekt*

- Definiera, skapa och använda objekt
- Objekt som hanterar strängar
- Typiska tjänster för ett objekt
- Objektresurser och klassresurser
- Arv och klasshierarkin
- Hantering av objekt

## Algoritmer i samband med objekt

### *Undantagssituationer*

- Hantera en undantagssituation
- Hierarkin med undantagsklasser
- Olika mönster för undantagshantering

### *Inmatning och utmatning*

- Strömmar
- Standardinmatning och standardutmatning
- Hantera filer
- Textfiler
- Binära filer
- Filer med objekt
- Filer med direkt åtkomst

### *Skapa nya objekttyper*

- Sammanfatta datatyper
- Ett objekts data
- Ett objekts operationer
- Inkapsling och gränssnitt
- Instansresurser och klassresurser
- Ett objekts referenser
- Nästlade och inre klasser

### *Utveckla nya objekttyper*

- Definiera en ny typ av objekt
- Implementera definitionsklassen
- Testa definitionsklassen
- Beskrivning av en definitionsklass

### *Arv*

- En subklass
- Superklassreferenser och subklassreferenser
- Polymorfism och dynamisk bindning

### *Klasshierarkier*

- En klasshierarki
- Hantera en klasshierarki
- Rotklassen i Javas klasshierarki
- Typoberoende programmering
- Typoberoende datastrukturer

## *Gränssnitt*

Definiera och implementera ett gränssnitt

Gränssnitt med konstanter

Hierarkier av gränssnitt

Typoberoende programmering med gränssnitt

Gränssnitt gentemot multipelt arv

Gränssnitt och inre klasser