Goals and content

ID1018

July 12, 2014

Goals and content

Learning goals

Having finished the course, the student is able to:

- Analyze, explain, modify, and extend a program consisting of fundamental program constructs.
- Describe how numerical and character based data is represented, and use primitive data types, character strings, and built-in data structures.
- Design, implement, test, and debug a program that uses basic computation, simple in- and ouput, common conditional and iterative structures, and function definitions.
- Discuss the importance of algorithms, identify the necessary properties of good algorithms, and create algorithms to solve simple problems.
- Develop code that handles exceptional states in the execution.
- Discuss programs with objects and understand the concepts encapsulation, inheritance, polymorphism, and interface.
- Design, implement, test, and debug simple programs in an object-oriented programming language.

Contents of the course

A Java program

- A problem and an algorithm
- The structure of a Java program
- Compiling and executing a Java program

- Designing and commenting Java code
- Basic output
- The Java language and the standard library

Data storage

- Variables
- Primitive data types in Java
- Type conversions when storing data
- Character strings

Standard input

- An input stream
- Entering characters
- Entering character strings
- Entering numbers

Operations on primitive values

- Operations on integers
- Operations on floating point numbers
- Operations on a variable
- Comparisons of primitive values
- Operations on Boolean values

Logic

- Conditional actions
- Iterations

Vectors

- One-dimensional vectors
- Multi-dimensional vectors

Methods

- Creating and using methods
- Managing methods
- Vectors as parameters and return values
- Recursive methods

A class library

- Distributed code
- The standard library in Java
- Exceptions of a method

Algorithms

- A problem and an algorithm
- Selection algorithms
- Search algorithms
- The complexity of an algorithm
- The correctness of an algorithm

Objects

- Define, create and use objects
- Objects that manages strings
- Typical services for an object
- Object resources and class resources
- Inheritance and class resources
- Managing objects
- Algorithms related to objects

Input and output

- Streams
- Standard input and standard output
- Managing files
- Textfiles
- Binary files
- Files with objects
- Files with direct access

Creating new object types

- Defining a new type of object
- Implementing the definition class
- Testing the definition class
- A description of a definition class

Inheritance

- A subclass
- Superclass references and subclass references
- Polymorphism and dynamic binding

Class hierarchies

- A class hierarchy
- Managing a class hierarchy
- The root class in the class hierarchy of Java
- Type independent programming
- Type independent data structures

Interfaces

- Defining and implementing an interface
- Interfaces with constants
- Hierarchies of interfaces
- Type independent programming with interfaces
- Interfaces vs multiple inheritance
- Interfaces and inner classes