

## COURSE SYLLABUS

# Introduction to Script Programming, 7.5 credits

*Skriptprogrammering, 7,5 högskolepoäng*

---

<b>Course Code:</b>	TSPG17	<b>Education Cycle:</b>	First-cycle level
<b>Confirmed by:</b>	Dean Sep 27, 2018	<b>Disciplinary domain:</b>	Technology (95%) and social sciences (5%)
<b>Revised by:</b>	Director of Education Sep 27, 2018	<b>Subject group:</b>	TE9
<b>Valid From:</b>	Aug 1, 2018	<b>Specialised in:</b>	G1N
<b>Version:</b>	3	<b>Main field of study:</b>	Informatics

---

## Intended Learning Outcomes (ILO)

After completing the course, the student shall

Knowledge and understanding

- display knowledge of how a programming language interacts with the operating system and the underlying hardware
- display knowledge about integrated development environments and their usage area
- display knowledge of common terms and concepts in script programming
- display knowledge of common terms and concepts used in object-oriented programming

Skills and abilities

- demonstrate the ability to use common data types and controls structures in a high-level programming language
- demonstrate the ability to create interactive console programs in a high-level programming language
- demonstrate the ability to develop and implement simple programs and algorithms in a high-level programming language
- demonstrate the ability to use object-oriented programming in a high-level programming language
- demonstrate the ability to use script programming to create, read and modify files of different types in a high-level programming language

## Contents

The course starts with the foundations of imperative programming. A programming language on high-level will be used to exemplify these concepts. In the beginning of the course the focus will be on creating simple console programs. The focus will then be to develop and implement algorithms to solve different problems.

The course includes the following elements:

- The foundation of imperative programming
- Integrated Development Environments (IDE)

- Program syntax in a high-level programming language
- The foundations of a programming language
- Creation of algorithms
- Console programs (input and output)
- Usage of different data formats in programming, e.g. XML, JSON, CSV
- Read and write information from and to files
- The foundation of object-oriented programming

### Type of instruction

Instruction consists of lectures and laboratory work.

The teaching is conducted in English.

### Prerequisites

General entry requirements and Mathematics 2a or 2b or 2c. Or: Mathematics B and English A (or the equivalent).

### Examination and grades

The course is graded 5,4,3 or Fail.

Registration of examination:

Name of the Test	Value	Grading
Examination <sup>1</sup>	3 credits	5/4/3/U
Laboratory work	4.5 credits	U/G

<sup>1</sup> Determines the final grade of the course, which is issued only when all course units have been passed.

### Course literature

#### Literature

The literature list for the course will be provided one month before the course starts.

Title: How to Think Like a Computer Scientist: Learning with Python

Author: Allen Downey, Jeffrey Elkner and Chris Meyers

Publisher: Unknown

ISBN-13: 978-0971677500

ISBN-10: 0971677506

(Available free at <http://interactivepython.org/runestone/static/thinkcspy/index.html>)