Computer Programming in Phyton (D013519)

Course size   *(nominal values; actual values may depend on programme)*

Credits 3.0   **Study time 90 h**

Course offerings and teaching methods in academic year 2024-2025

A (semester 2)	Dutch	Gent	lecture

Lecturers in academic year 2024-2025

Vermeirssen, Vanessa	WE14	lecturer-in-charge

Offered in the following programmes in 2024-2025	crdts	offering
Bachelor of Science in Biomedical Sciences	3	A
Linking Course Master of Science in Biomedical Sciences	3	A

Teaching languages

Dutch

Keywords

Programming, problem solving, programming language Python, Jupyter Notebooks

Position of the course

In recent years the medical world has been confronted with a data tsunami, from clinical research to patient files. In addition, the learning environment, in society in general and more specifically within higher education, is changing drastically with the widespread use of digital technology. The acquisition of new competences have thus become mandatory.

Specifically, biomedical researchers and people from the medical field often have to deal with time-consuming and repetitive tasks, which can be automated by computers through programming. Hence, it is of utmost importance that students biomedical sciences acquire basic principles of prorgamming. Moreover, they acquire a (self)critical, analytical and synthetic thinking attitude.

Python is a general and simple programming language, with a readable syntax, which is used in many societal and scientific contexts, including the biomedical sciences.

The aim of this course is to teach the students the basic skills of programming in Python so they can apply these during their education and later on in the workfield.

Contents

To acquire the necessary knowledge and competences the course is divided in a theoretical part and a practical part, spread over lectures, practicals and self-study.

Programming is the process of writing the source code of computer programs. This requires knowledge of the syntax and the semantics of a programming language and the skills to write programs in that language. In the first place it also learns the skill of thinking in problem solving through programming,

The theoretical part focusses on programming in Python for biomedical sciences.

• Why do you need programming skills in biomedical sciences?

• What is programming and how do I learn to think as a programmer?

• What is Python? Why programming in Python?

• Writing and running Python programs.

• Elements in Python:

- basic components: instructions, variables, data types and operators
- control structures: conditional tasks, control loops and functions
- data structures: strings, lists, tuples, dictionaries, sets, files and modules

The practical part contains the writing and interpreting of Python source code for simple problems in the biomedical field and out there.

- Use of Jupyter Notebooks, a web application to share documents and source code for data scientists.
- Writing computer programs in Python.
- Exercises via the e-learning environment Dodona.
- Initiation in processing biomedical datasets in Python.
- Translating a (biomedical) problem to a computer program.

Initial competences

The basic ICT knowledge and skills as acquired in contemporary secondary school (ages 12 to 18).

Final competences

1 Applying the basic principles of programming in Python
2 Writing autonomously and accurately Python programs
3 Detecting and correcting errors in Python code
4 Applying interactively Python code in Jupyter Notebooks
5 Applying Python programming in the online learning platform Dodona, even under mild time pressure
6 Translating a biomedical research question and/or time consuming task from natural language to Python code

Conditions for credit contract

Access to this course unit via a credit contract is unrestricted: the student takes into consideration the conditions mentioned in 'Starting Competences'

Conditions for exam contract

This course unit cannot be taken via an exam contract

Teaching methods

Seminar, Lecture, Independent work

Extra information on the teaching methods

- Lectures
- Self-study and self-evaluation with digital aids
- Practical exercises in PC-class / on laptop
- Use of electronic learning platform Ufora
- Use of Dodona (e-learning platform for programming)
- Use of Jupyter Notebooks (web application for data scientists)

Study material

Type: Other

Name: Freely available course material on the e-learning platforms Dodona and Ufora
Indicative price: Free or paid by faculty
Optional: no

References

Thematic and complementary information is given on Ufora and Dodona e-learning platforms

Course content-related study coaching

Course content-related study coaching
- Possibility to active questioning during and immediately after contact hours (colleges and practicals)
- Trough the Ufora and Dodona e-learning platforms

Assessment moments

end-of-term and continuous assessment

Examination methods in case of periodic assessment during the first examination period

Skills test, Written assessment with multiple-choice questions

Examination methods in case of periodic assessment during the second examination period

Skills test, Written assessment with multiple-choice questions

Examination methods in case of permanent assessment

Skills test

Possibilities of retake in case of permanent assessment

examination during the second examination period is possible in modified form

Extra information on the examination methods

Via online programmes the skill of programming in Python is evaluated:
• MCQ with standard setting in Curios (closed book)
• Programming skill exercises via open-ended questions in Dodona (open book)
The use of any generative AI at evaluations is explicity forbidden, since in this basic
course programming the student needs to learn the basic skills.

Calculation of the examination mark

Non-periodic evaluation / permanent evaluation:
• Skills test programming in Python, 4 of 20 points
Periodic evaluation (examination):
• MCE and skills test exam programming in Python, 16 of the 20 points
DO NOTE:
• Unjustified absence in the permanent evaluation in the online Ufora system will
  give rise to a mark of 0/4.
• Both parts of the exam are equally weighted and students need to pass both
  MCE and Dodona programming skills test individually to pass for the course.