Course Specifications
Valid as from the academic year 2019-2020

Course
Programming II (I700203)

Valid as from the academic year 2019-2020

Course offerings and teaching methods in academic year 2019-2020

A (semester 2) Dutch lecture 12.0 h
seminar: practical PC room classes 20.0 h

Lecturers in academic year 2019-2020
Verwaeren, Jan LA26 lecturer-in-charge

Offered in the following programmes in 2019-2020
Bachelor of Science in Bioscience Engineering Technology
3 A

Teaching languages
Dutch

Keywords
programming, Python

Position of the course
Today, collecting, storing and processing of information is largely automated and
computerized in applied biosciences. The course Programming II aims, in the first
place, at providing the student some programming skills needed to hook onto this
modern evolution. Specifically, this course builds upon the introductory programming
course Programming I and elaborates on a number of programming concepts and pays
attention to a number of common applications and problems where programming skills
provide a significant added value.

Contents
The content of this course builds primarily upon the course Programming I and
considers a number of more advanced scientific programming concepts:
• Numpy for working with arrays
• Matplotlib for visualizing data
• (general) file input/output
• (general) exception handling and debugging
The basic concepts of Programming I and the concepts mentioned above will be
applied to a number of domains. For each of these domains, a theoretical and practical
introduction will be given whereupon students solve real-life problems in the domain of
applied biosciences. The list below is a guidance to the above mentioned domains:
• Numerical methods in a mathematical-scientific context (i.e. numpy and scipy)
• Data visualization (i.e. matplotlib)
• Image Processing (i.e. scikit-image)
• The use of single-board computers (such as Raspberry Pi) for information collection
  and processing with sensors
• Prediction problems

Initial competences
This course builds upon certain final competences acquired by the course
Programming I.

Final competences
1 The student can distinguish the different components of a problem by analytical
reasoning, detect parts of a solution strategy that can be automated and implement
this solution strategy in Python.
2 Knowing and being able to use Python as a scientific programming environment

(Approved)
The student should, for a given problem, be able to select and apply a suited Python module.

The student has a deeper understanding of various domains within the field of computer science (such as digital image processing) that have applications in the field of applied biosciences.

Access to this course unit via a credit contract is determined after successful competences assessment.

This course unit cannot be taken via an exam contract.

Lecture, seminar: practical PC room classes

During the lectures new theoretical principles are introduced and illustrated by means of examples.

During the seminars, the students can, under the guidance of the lecturer and/or an educational supervisor, practice and apply the above concepts in a number of case studies from research areas.

Lecture slides are available via the electronic platform Minerva.

Python practical course notes.

For advanced and applied topics, public access course material is used.


There is a permanent tutorial service on Friday. Students can use this tutorial service all year long, but will be requested to notify this at least one day in advance by email.

Evaluation methods

End-of-term evaluation

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

Open book examination

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

Open book examination

Examination methods in case of permanent evaluation

Possibilities of retake in case of permanent evaluation

Not applicable

Extra information on the examination methods

The periodical evaluation is an open book exam, consisting of PC exercises. During this exam, the students use a PC and the students are assessed on the basis of the submitted source code.

Calculation of the examination mark

The periodical evaluation is an open book exam, consisting of PC exercises. During this exam, the students use a PC and the students are assessed on the basis of the submitted source code.