Course Specifications
Valid as from the academic year 2019-2020

Object Oriented Programming (E620500)

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

<table>
<thead>
<tr>
<th>Credits</th>
<th>Study time 90 h</th>
<th>Contact hrs 36.0 h</th>
</tr>
</thead>
</table>

Course offerings and teaching methods in academic year 2019-2020

A (semester 1) Dutch
lecture 12.0 h
seminar 24.0 h

Lecturers in academic year 2019-2020

Ongenae, Veerle TW05 lecturer-in-charge

Offered in the following programmes in 2019-2020 crds offering
Bachelor of Science in Engineering Technology (main subject Machine and Production Automation) 3 A
Linking Course Master of Science in Electrical Engineering Technology (main subject Automation) 3 A
Linking Course Master of Science in Electromechanical Engineering Technology 3 A

Teaching languages
Dutch

Keywords
Software, programming, OOP, C#, .NET, software layers, informatics

Position of the course
The main goal of this course is learning to program in an object oriented way. The program language is C#. Moreover, the course has a general educating value: it provides insight in abstract structures and processes, the development of analytics, it helps with the modular thinking, ... This theoretical knowledge and the new competences are useful for other domains.
The course is the base for other software oriented courses.

Contents
- .NET Framework & Visual Studio
- Memory
  - Stack & Heap
  - Value types vs. reference types
- Basic principles on object oriented programming:
  - classes
  - objects
  - properties, methods and events
  - constructors and finalizers
  - overloading
  - Inheritance
  - overriding
  - polymorphism
  - dynamic binding
  - interfaces
- Software layers (GUI, BLL, DAL)
- Exception handling
- IO

Initial competences
The course unit informatics has to be followed or the intended competencies has to be achieved in one way or another.

(Approved)
Final competences
1. Knowing the basic concepts on object oriented programming in C#.NET and be able to use them.
2. To be able to analyse and structure a problem and be able to translate everything into an (layered) OO software concept/design.
3. Creating, testing and debugging a C#.NET program based on a (layered) OO software concept/design.

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

Conditions for exam contract
This course unit cannot be taken via an exam contract.

Teaching methods
Lecture, seminar

Extra information on the teaching methods
The theoretical concepts are explained step by step with examples during the lectures. In the seminars, the student works autonomous on a computer.

Learning materials and price
- Presentations (Dutch), free pdf version on the electronic learning platform
- Software Visual Studio

References

Course content-related study coaching
- Interactive support via the electronic learning environment and mail
- Personal feedback by appointment

Evaluation methods
end-of-term evaluation and continuous assessment

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

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

Examination methods in case of permanent evaluation
Skills test

Possibilities of retake in case of permanent evaluation
examination during the second examination period is possible

Extra information on the examination methods

Periodic evaluation
The examination is practical, based on exercises with maybe some theoretical questions.

Permanent evaluation
During the seminars, there is one or more skill test.

Calculation of the examination mark
Final score (/20) = C1 x P1 + C2 x P2
Here are Cx the weighting coefficients and Px the points (/20):
P1: points theory (periodic evaluation)
P2: points seminar (permanent evaluation)
C1 = 50%, C2 = 50%
During the second exam chance the points of the NPE disappear and only the points obtained on the exam count.

(Approved)