



## COURSE DESCRIPTION CARD - SYLLABUS

Course name

Program languages [S1IBio1E>JEP]

### Course

Field of study

Biomedical Engineering

Year/Semester

1/2

Area of study (specialization)

–

Profile of study

general academic

Level of study

first-cycle

Course offered in

English

Form of study

full-time

Requirements

compulsory

### Number of hours

Lecture

15

Laboratory classes

30

Other

0

Tutorials

0

Projects/seminars

0

### Number of credit points

4,00

### Coordinators

dr inż. Mateusz Wróbel

### Lecturers

### Prerequisites

Basic knowledge of logic and computer science

### Course objective

Transfer of knowledge allowing procedural and object-oriented programming

### Course-related learning outcomes

Knowledge:

The student recognizes and knows the features of procedural, object-oriented and visual programming

The student knows the basic structures of selected programming languages

The student knows the concepts of classes, structures, objects, inheritance, polymorphism, encapsulation

Skills:

The student can create dedicated software

Social competences:

The student understands the role of computerization in the modern economy. Is able to participate

creatively

## Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lecture: Test, 20 closed questions, passing the subject -50% of the maximum points

Laboratory: short tests, passing the subject- 50% of the maximum points

## Programme content

Lecture: General principles of program construction. Compilers and interpreters. Programming in low and high level languages, overview and division of languages. Visual programming languages. Structural programming. Basics of programming in C / C ++. Variables, data types, pointers, operators, loops, conditional instruction, input and output functions. Object oriented programming. The concepts of encapsulation, classes, objects, inheritance, polymorphism. Basics of object-oriented programming in C ++. References, operator overloading, streams, exceptions, namespaces. Basics of Python language. Lab: Structured programming in C / C ++, examples: data input and output, simple calculations, use of conditional instruction, selection instruction, loops, writing and reading a text, binary file, creating functions. Object-oriented programming in C ++, examples: creating classes, single-inheritance, operator overloading, using the STL library. Basics of Python language.

## Course topics

none

## Teaching methods

Lecture: multimedia presentation with theory and examples.

Laboratory classes: practical exercises, problem solving

## Bibliography

Basic:

1. Liberty J., Rao S., Jones B, L, - C++ dla każdego, Helion, Gliwice 2011
2. Wróblewski P., Algorytmy, struktury danych i techniki programowania, Helion, Gliwice 2009
3. Sarbicki S., Python. Kurs dla nauczycieli i studentów, Helion, Gliwice 2019.

Additional:

Sedgewick R., Algorytmy w C++, READ ME, Łódź 1999

Kliszewski M., Inżynieria oprogramowania obiektowego, WKT, Warszawa 1994

## Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	47	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	53	2,00