SYLLABUS PREDMETA

General information

Course title:	**PYTHON**
ISVU[1] course code:	225888 / MT735
Studies in which the course is taught:	MECHANICAL ENGINEERING
Course Instructor:	Ph.D Adam Stančić, senior lecturer
Course Assistant:	Ms.C Vedran Vyroubal, senior lecturer
ECTS credits:	5.0
Semester of the course execution:	I. (winter sem.)
Academic year:	2022 / 2023
Exam prerequisites:	- - -
Lectures are given in a foreign language:	English
Aims:	The aim of the course is the application of the Python programming language in solving mathematical-logical and technical problems. By algorithmic approach and development of logical thought processes in the process of problem solving and understanding and applying a systematic approach to the problem, the student will be able to solve given problems by applying appropriate algorithms, software and available programming language libraries.

Course

Course structure	Number of contact hours per week:	Number of contact hours per semester:	Student's requirements by type of teaching:
Lectures:	2	30	attendance 80%
Tutorials:	2	30	attendance 80%
Practical (lab) sessions:			
Seminars:			
Field work:			
Other:			
TOTAL:	4	60	

Monitoring of students' work, knowledge evaluation and learning outcomes

Formation of the grade during the implementation of teaching: (Define from minimum 5 to maximum 10 learning outcomes)	LEARNING OUTCOMES (upon completion of the course the student should be able to:)	FACTORS AFFECTING THE GRADE (e.g. term paper, practical work, presentation, ...)	MAXIMUM NUMBER OF POINTS PER FACTOR
	I 1: List the basic characteristics of the Python programming language	Colloquium I	Colloquium I 40 points Colloquium II 40 points Indipendent work 20 points
	I 2: Describe basic mathematical-logical operations on different types of data	Colloquium I	
	I 3: Use data collections and user-defined functions	Colloquium I	
	I 4: Associate the programming code with the available programming language modules	Colloquium II	
	I 5: Manage errors in program code	Colloquium II	
	I 6: Visually and logically valorize the results of data	Colloquium II	

---

[1] ISVU – Information System of Higher Education Institutions in Croatia

	processing		
	I 7: - - -		
	I 8: - - -		
	I 9: - - -		
	I 10: - - -		
Alternative formation of the grade (I 1 – I 10)	**or alternative formation of the grade: I 1 – I 10**		TOTAL: 100 points
Students' competencies	Independent development of applications in the Python programming language in order to solve technical and mathematical problems and the possibility of using different libraries and modules of the programming language.		

Prerequisites for course approval (lecturer's signature):	Attendance at classes and laboratory exercises min. 80%
Prerequisites for taking exams:	Signature + term paper + passed exercises
Grading scale:	(According to the Regulations on student assessment of Karlovac University of Applied Sciences, Article 9, Paragraph 5) 90-100 - excellent (5) (A) 80 to 89.9 - very good (4) (B) 65 to 79.9 - good (3) (C) 60 to 64.9 - sufficient (2) (D) 50 to 59.9 - sufficient (2) (E) 0 to 49.9 – fail (1) (F) Students are graded during class, what forms 70% of final exam. Students who achieve 50% (35 points) and more are allowed to take the final exam. The score on final exam makes 30% of the final grade.

ECTS structure

ECTS credits allocated to the course reflect the total burden to the student during adoption of the course content. Total contact hours, relative gravity of the content, effort required for exam preparation, as well as, every other possible burden are taken in account:

Attendance (active participation)	Term paper	Composition	Presentation	Continuous assessment and evaluation	Practical work
0,5					
Independent work	**Project**	**Written exam**	**Oral exam**	**Other**	
2,0		1,5	1,0		

Review of topics/units per week associated with learning outcomes

Week	Lectures topics/units and learning outcomes:	Tutorials topics/units and learning outcomes:
1.	Basic concepts and characteristics, development environment, versions of prog. language **I 1**	Version selection, installation and startup, operating system settings within the operating system **I 1**
2.	Work within the development environment, code writing, syntax, data types **I 1**	Interactive and script mode, identifiers, reserved expressions, code lines and indentations, data types, constants and variables **I 1**
3.	Operators **I 2**	Defining different types of operators **I 2**
4.	Conditional execution, loops **I 2**	Types of conditions and loops, nested conditions

SYLLABUS PREDMETA

		and loops **I 2**
5.	Numeric values and text handling, display formatting, conversion of numeric and text values **I 2**	Defining the type and conversion of numerical values, basic mathematical functions and constants, display of the desired part of the text, formatting of display **I 2**
6.	Data collections (array, list, tuple, dictionary) **I 3**	Defining, accessing, modifying and deleting data in a field, list, n-torque and dictionary (array, list, tuple, dictionary) **I 3**
7.	Working with data collections (array, list, tuple, dictionary) **I 3**	Methods and operators for working with field, lists, n-tuples and dictionary (array, list, tuple, dictionary) **I 3**
8.	Functions, call by value or reference (Call-by-value, Call-by-reference) **I 3**	Defining and calling a function, calling by value or reference, types of function arguments **I 3**
9.	Working with functions, lambda functions **I 3**	Examples of creating functions, using lambda functions **I 3**
10.	Working with data **I 3**	Reading and writing to a file, renaming and deleting a file, working with directories (folders) **I 3**
11.	Modules **I 4**	Defining, locating and working with modules (os, sys, math, statistics, etc.) **I 4**
12.	Date and time **I 3, I 4**	Working with time, working with dates, measuring time flow, display format, working with date / time modules **I 3, I 4**
13.	Error Management **I 5**	Error types, condition evaluation (assert), error management (try, finally, except, else), arguments **I 5**
14.	Data visualization **I 4, I 6**	Creating charts using Matplotlib library **I 4, I 6**
15.	Presentation of practical works of students **I 1-I 6**	Presentation of practical works of students **I 1 - I 6**

References

REFERENCES (compulsory/additional):

Compulsory:
- Unauthorized lecture tracking scripts and presentations (author: Adam Stančić)

Additional:
- L. Budin, P. Brođanac, Z. Markučić, S. Perić-Rješavanje problema programiranjem u Pythonu, Element d.o.o., 2018
- Z. Kalafatić, A. Pošćić, S. Šegvić, J. Šribar - Python za znatiželjne, Element d.o.o., 2018
- M. Lutz - Learning Python, 5th Edition, O'Reilly Media, 2013
- On-line data sources related to the presented unit

Exams for the academic year: _____2022./____2023.

Exam dates:	According to the schedule of exams for academic year published on the web- site

Contact information

1. Course Instructor/Lecturer:	Ph.D. Adam Stančić, senior lecturer
e-mail:	adam.stancic@vuka.hr
Office hours / Consultations:	Tue, 10:00, Meštrovićeva 10, 1st floor, room no. 109
2. Course Instructor/Lecturer:	Ms.C. Vedran Vyroubal, senior lecturer
e-mail:	vedran.vyroubal@vuka.hr
Office hours / Consultations:	Wed, 10:00, Meštrovićeva 10, 1st floor, room no. 109