	Universitas Negeri Surabaya **Vocational Faculty,** **D4 Informatics Management Study Program**	**Document Code**

SEMESTER LEARNING PLAN

Courses	**CODE**	**Course Family**	**Credit Weight**			**SEMESTER**	**Compilation Date**
Algorithms and Programming	5730102201	Algorithms and Programming	**T=2**	**P=0**	**ECTS=3.18**	1	April 28, 2023

AUTHORIZATION	**SP Developer**	**Course Cluster Coordinator**	**Study Program Coordinator**
	I Gde Agung Sri Sidhimantra S.Kom., M.Kom.	I Gde Agung Sri Sidhimantra S.Kom., M.Kom.	Dodik Arwin Dermawan, S.ST., S.T., M.T.

Learning model	**Case Studies**

Program Learning Outcomes (PLO)

PLO study program which is charged to the course

Program Objectives (PO)

PO - 1	Students are able to understand the types of programming languages and the computer program development cycle.
PO - 2	Students are able to understand algorithms and mathematical modeling
PO - 3	Students are able to carry out program testing and error tracking
PO - 4	Students are able to create simple computer programs

PLO-PO Matrix

P.O
PO-1
PO-2
PO-3
PO-4

PO Matrix at the end of each learning stage (Sub-PO)

P.O	Week															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
PO-1																
PO-2																
PO-3																
PO-4																

Short Course Description	This course teaches basic programming concepts, practical and technical knowledge and experience regarding algorithms, flowcharts and their application in the C++ programming language. The basic materials for making programs are programming basics, introduction to the C++ programming language, C++ control structures, completing conditions, loops, arrays, strings, pointers, functions, abstract data types/structures, and file operations.

References	**Main :**	

	1. Ekohariadi, Qoiriah, A. 2007.Bahasa Pemrograman C. Unipress UNESA. 2. Jeri R. Hanly and Eliiot B. Koffman. 2002.Problem Solving and Program Design in C.Addison Wesley Publishing. 3. Barton, John J., Nackman, Lee R. 1994.Scientific and Engineering C++: an introduction with advanced techniques and examples. Addison Wesley Longman, Inc. 4. The Waite Group's. 1992. C++ Programming, Second Edition. SAMS a division of Prentice Hall Computer Publishing. 5. Kadir, A dan Heriyanto. 2005.Algoritma Pemrograman Menggunakan C++. Yogyakarta: Penerbit Andi. 6. Pranata, A. 2005.Algoritma dan Pemrograman. Yogyakarta: Penerbit Graha Ilmu. 7. Liberty, J., Rao, S., Jones, B. 2008.Sams teach yourself C++ in one hour a day. Sams.	
	Supporters:	
Supporting lecturer	Andi Iwan Nurhidayat, S.Kom., M.T. I Gde Agung Sri Sidhimantra, S.Kom., M.Kom.	

Week-	**Final abilities of each learning stage (Sub-PO)**	**Evaluation**		**Help Learning, Learning methods, Student Assignments, [Estimated time]**		**Learning materials [References]**	**Assessment Weight (%)**
		Indicator	**Criteria & Form**	**Offline (*offline*)**	**Online (*online*)**		
(1)	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**
1	Students are able to apply algorithms and flowcharts in problem solving	1.Explain the basic concepts of algorithms 2.Identify flowchart notations 3.Applying algorithms and flowcharts to solve problems	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
2	Students are able to explain the structure of writing the C programming language	1.Identify types of data types 2.Explain the rules for defining identifiers 3.Identify the difference between variables and constants 4.Identify the types of operators 5.Explain the precedence of arithmetic operators	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Cooperative Method: Discussion, Presentation, Practical 4 X 50			0%
3	Students are able to apply input and output functions in making programs	1.Identify the types of input and output functions 2.Implement input and output functions in the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
4	Students are able to create programs with the branching concept	1.Identify differences in conditions and actions 2.Explain single, compound and multilevel branching 3.Explaining branching using case selection 4.Implement the concept of branching into the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%

5	Students are able to create programs with the branching concept	1. Identify differences in conditions and actions 2. Explain single, compound and multilevel branching 3. Explaining branching using case selection 4. Implement the concept of branching into the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
6	Students are able to create programs with the concept of repetition	1. Identify types of repetition 2. Explain the loop structure 3. Apply the concept of repetition to the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
7	Students are able to create programs with the concept of repetition	1. Identify types of repetition 2. Explain the loop structure 3. Apply the concept of repetition to the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
8	Subsummative Exam / Midterm Exam	Subsummative Exam / Midterm Exam	**Criteria:** Subsummative Exam / Midterm Exam	Subsummative Exam / Midterm Exam 4 X 50			0%
9	Students are able to create programs using array concepts	1. Explain the definition of an array 2. Identify types of arrays 3. Explains how to declare each array 4. Implementing arrays in programs	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
10	Students are able to create programs with string concepts	1. Explains the definition of a string 2. Explains how to declare string variables 3. Explains how to enter and display the contents of a string variable 4. Explains how to access string elements 5. Implementing strings in programs	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
11	Students are able to use functions in making programs	1. Explain the basic concept of function 2. Explains how to declare a function 3. Explains how to call a function 4. Implement functions in programs	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%

12	Students are able to use recursive functions in making programs	1. Explain the definition of a recursive function 2. Explains how to declare a recursive function 3. Explains how to call a recursive function 4. Identify the similarities and differences between iterative and recursive functions 5. Identify the advantages and disadvantages of recursive functions 6. Implementing recursive functions in programs	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
13	Students are able to use pointers in making programs	1. Explain the basic concept of pointers 2. Explain how to use pointers 3. Applypointer in the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
14	Students are able to create programs with the concept of structure	1. Explain the basic concepts of structure 2. Explains how to declare structure variables 3. Implementing structure variables in the program	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
15	Students are able to create programs for file operations	1. Identify the differences between text files and binary files 2. Identify types of file operations in text files and binary files 3. Implementing file operations in program creation	**Criteria:** Assessment rubric (attached)	Approach: Scientific Model: Problem-based learning Method: Discussion, Presentation, Practical 4 X 50			0%
16	Summative Exam / Final Semester Exam	Summative Exam / Final Semester Exam	**Criteria:** Summative Exam / Final Semester Exam	Summative Exam / Final Exam 4 X 50 Semester			0%

Evaluation Percentage Recap: Case Study

No	Evaluation	Percentage
		0%

Notes

1. **Learning Outcomes of Study Program Graduates (PLO - Study Program)** are the abilities possessed by each Study Program graduate which are the internalization of attitudes, mastery of knowledge and skills according to the level of their study program obtained through the learning process.
2. **The PLO imposed on courses** are several learning outcomes of study program graduates (CPL-Study Program) which are used for the formation/development of a course consisting of aspects of attitude, general skills, special skills and knowledge.
3. **Program Objectives (PO)** are abilities that are specifically described from the PLO assigned to a course, and are specific to the study material or learning materials for that course.
4. **Subject Sub-PO (Sub-PO)** is a capability that is specifically described from the PO that can be measured or observed and is the final ability that is planned at each learning stage, and is specific to the learning material of the course.
5. **Indicators for assessing** ability in the process and student learning outcomes are specific and measurable statements that identify the ability or performance of student learning outcomes accompanied by evidence.
6. **Assessment Criteria** are benchmarks used as a measure or measure of learning achievement in assessments based on predetermined indicators. Assessment criteria are guidelines for assessors so that assessments are consistent and unbiased. Criteria can be quantitative or qualitative.
7. **Forms of assessment:** test and non-test.
8. **Forms of learning:** Lecture, Response, Tutorial, Seminar or equivalent, Practicum, Studio Practice, Workshop Practice, Field Practice, Research, Community Service and/or other equivalent forms of learning.
9. **Learning Methods:** Small Group Discussion, Role-Play & Simulation, Discovery Learning, Self-Directed Learning, Cooperative Learning, Collaborative Learning, Contextual Learning, Project Based Learning, and other equivalent methods.
10. **Learning materials** are details or descriptions of study materials which can be presented in the form of several main points and sub-topics.
11. **The assessment weight** is the percentage of assessment of each sub-PO achievement whose size is proportional to the level of difficulty of achieving that sub-PO, and the total is 100%.
12. TM=Face to face, PT=Structured assignments, BM=Independent study.