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

SEMESTER LEARNING PLAN

Courses	**CODE**	**Course Family**	**Credit Weight**			**SEMESTER**	**Compilation Date**
PRACTICE. DATA STRUCTURE	5730102205		**T=0**	**P=2**	**ECTS=3.18**	2	July 17, 2024

AUTHORIZATION	**SP Developer**	**Course Cluster Coordinator**	**Study Program Coordinator**
	....	....	Dodik Arwin Dermawan, S.ST., S.T., M.T.

Learning model	**Project Based Learning**
Program Learning Outcomes (PLO)	**PLO study program that is charged to the course**
	Program Objectives (PO)
	PLO-PO Matrix

P.O

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

Short Course Description

Implementation of Data Structure lectures such as pointers, structs, etc. Apart from that, there are also several data structures used in programming, both static and dynamic. And also algorithms in the sorting process and search process. Lectures contain theory, where programming assignments will be given.

References

Main :

1. Ekohariadi, Anita Qoiriah, Pemrograman Dasar Komputer, Unipress, , 2007.
2. Malik, D.S., C Programming: From Problem Analysis to Program Design, Fifth Edition, Course Technology, Cengage Learning, 2011.
3. Malik,D.S., Data Structures Using C , Second Edition, Course Technology, Cengage Learning, 2010.
4. Shaffer, Clifford A. A, Practical Introduction to Data Structures and Algorithm Analysis Edition 3.1 (C Version), Prentice Hall International Inc, 2011.
5. Yatini B, Indra, Erliansyah Nasution, Algortima dan Struktur Data dengan C , Graha Ilmu, 2005.
6. Zakaria, Teddy Marcus, Agus Prijono. Konsep dan Implementasi Struktur Data, Informatika Bandung, 2006.

Supporters:

Supporting lecturer

Dodik Arwin Dermawan, S.ST., S.T., 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	Understand the concepts of arrays, pointers and structures	1.Explain the concept of one and two dimensional arrays 2.Explain the concept of pointers 3.Explain the difference between memory allocation in arrays and pointers 4.Explain the concept of structure 5.Create programs with structure and array declarations on structure data types	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation and practicum 2 X 50			0%
2	Understand the concept of single linked list	1. Explain the declaration of a Single Linked List 2. Explain how to search in a Linked List 3. Explain the operation of inserting nodes in a single Linked List (at the beginning, at the end, in the middle) 4. Explain the operation of deleting nodes in a single Linked List (at the beginning, at the middle, at the end) 5. Implementing a single linked list in a case	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. Create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation and practicum 2 X 50			0%
3	Understand the concept of single linked list	1. Explain the declaration of a Single Linked List 2. Explain how to search in a Linked List 3. Explain the operation of inserting nodes in a single Linked List (at the beginning, at the end, in the middle) 4. Explain the operation of deleting nodes in a single Linked List (at the beginning, at the middle, at the end) 5. Implementing a single linked list in a case	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. Create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation and practicum 2 X 50			0%

4	Understand the concept of double linked lists	1. Explain the declaration of a double Linked List 2. Explain how to search in a double Linked List 3. Explain the operation of inserting nodes in a double Linked List (at the beginning, at the end, in the middle) 4. Explain the operation of deleting nodes in a double Linked List (at the beginning, in the middle, at the end) 5. Implement a double linked list in a case	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation and Practicum 2 X 50			0%
5	Understand the concept of double linked lists	1. Explain the declaration of a double Linked List 2. Explain how to search in a double Linked List 3. Explain the operation of inserting nodes in a double Linked List (at the beginning, at the end, in the middle) 4. Explain the operation of deleting nodes in a double Linked List (at the beginning, in the middle, at the end) 5. Implement a double linked list in a case	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation and Practicum 2 X 50			0%
6	Understand the stack concept	1. Representing a Stack with an array 2. Explaining Stack Operations (Push, Pop, empty, isfull etc.) 3. Representing a Stack with a Single Linked List 4. Representing a Stack with a Double Linked List 5. Implementing a stack in several cases	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation, Assignment and practicum exercises 2 X 50			0%
7	Understand the concept of queue	1. Representing a queue with an array 2. Explaining queue operations (enqueue, dequeue, is empty, isfull etc.) 3. Representing a queue with a Single Linked List 4. Representing a queue with a Double Linked List 5. Implementing queues in several cases	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation, Assignment and practicum exercises 2 X 50			0%
8	U.S.S			2 X 50			0%

9	Understand the concept of recursion functions and their implementation	1. Explain the basic concept of recursion 2. Implement recursion in several cases	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, presentation, Presentation/Assignment and practicum 2 X 50			0%
10	Understand various methods in sequencing and their implementation	1. Explaining the Insertion Method 2. Explaining the Selection Method 3. Explaining the Bubble Method 4. Explaining the Shell Method 5. Explaining the Quick Method 6. Explaining the Merge Method 7. Examples of simple cases that require sorting to solve, create algorithms and flow charts 8. Implement using C language	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation/Assignment and practicum 2 X 50			0%
11	Understand various methods in sequencing and their implementation	1. Explaining the Insertion Method 2. Explaining the Selection Method 3. Explaining the Bubble Method 4. Explaining the Shell Method 5. Explaining the Quick Method 6. Explaining the Merge Method 7. Examples of simple cases that require sorting to solve, create algorithms and flow charts 8. Implement using C language	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation/Assignment and practicum 2 X 50			0%
12	Understand the concept of searching and its implementation	1. Explaining Search using the sequential method 2. Explaining Search using the binary method 3. Comparing the performance of sequential with binary search 4. Implementing the search method for simple cases that require an understanding of searching to solve them	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation/Assignment and practicum 2 X 50			0%

13	Understand the concept of trees and problems that use tree implementations to solve them	1. Explain the concept of a tree 2. Explain the introduction of terms in a tree 3. Explain forming a binary tree 4. Explain visits to a tree in preorder, inorder, or postorder 5. Represent a tree with a linked list 6. Explain the implementation of polish notation using a tree	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation/Assignment and practicum 2 X 50			0%
14	Understand graph concepts and graph implementation in path finding algorithms	1. Explain the concept of graphs 2. Explain types of graphs: directed and undirected graphs 3. Represent graphs with arrays 4. Represent with linked lists 5. Explain the application of graphs in implementing directed and undirected path finding algorithms	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation/Assignment and practicum 2 X 50			0%
15	Understand graph concepts and graph implementation in path finding algorithms	1. Explain the concept of graphs 2. Explain types of graphs: directed and undirected graphs 3. Represent graphs with arrays 4. Represent with linked lists 5. Explain the application of graphs in implementing directed and undirected path finding algorithms	**Criteria:** 1.1. Created a program, but it didn‘t work = 50 2.2. Create a program and run, but it doesn‘t match = 70 3.3. create a program and succeed =90	Approach: Scientific Model: Cooperative Method: Discussion, Presentation/Assignment and practicum 2 X 50			0%
16							0%

Evaluation Percentage Recap: Project Based Learning

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.