|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | **UNIVERSITAS NEGERI PADANG**  **FACULTY OF ENGINEERING**  **ELECTRONIC DEPARTMENT**  **INFORMATIC EDUCATION STUDY PROGRAM** | | | | | | | | | | | **Document Code** | |
| **SEMESTER LEARNING PLAN (SLP)** | | | | | | | | | | | | | | |
| **COURSES** | | | | | | **CODE** | **Course Group** | | **Credit Point(s)** | | **SEMESTER** | **Date Of Creation** | | |
| **Data Structure Practicum** | | | | | | TIK1. 61.2303 | Study Program Compulsory Courses | | 2 SKS (PRACTICE) | | 2 | July 2017 | | |
| **AUTHORIZATION** | | | | | | **Lecturer** | | | **Course Coordinator** | | **Coordinator of Study Program** | | | |
| **Fadhli Ranuaharja, S.Pd., M.Pd.T.** | | |  | | **Ahmaddul Hadi, S.Pd, M.Kom**  **NIP. 197612092005011003** | | | |
| **Learning Outcomes (LO)** | | | **PLO** | | |  | | | | | | | | |
| PLO-S1 | Believe in God Almighty and be able to show a religious attitude | | | | | | | | | | |
| PLO-S9 | Demonstrate an attitude of responsibility for work in their field of expertise independently | | | | | | | | | | |
| PLO-PP3 | Mastering the use of Information Technology products in learning in the field of Information Engineering | | | | | | | | | | |
| PLO-KU1 | Able to apply logical, critical, systematic, and innovative thinking in the context of developing or implementing science and technology pay attention to and apply humanities values ​​by their field of expertise | | | | | | | | | | |
| PLO-KU5 | Able to make decisions appropriately in the context of problem-solving in their area of ​​expertise, based on the results of information and data analysis. | | | | | | | | | | |
| PLO-KK15 | Ability to create multimedia products and graphic products with various design methods and technologies. | | | | | | | | | | |
| **CO** | | |  | | | | | | | | |
| CO-1 | Students are able to understand, describe, and implement the basic concepts of Algorithms, Data Structures, and the use of IDEs. | | | | | | | | | | |
| CO-2 | Students are able to understand, describe, and implement Arrays, Pointers, & Structures in C programming using the IDE. | | | | | | | | | | |
| CO-3 | Students are able to understand, describe, and implement Single Linked List in C programming using IDE. | | | | | | | | | | |
| CO-4 | Students are able to understand, describe, and implement Double Linked List in C programming using IDE. | | | | | | | | | | |
| CO-5 | Students are able to understand, describe, and implement Circular Linked List in C programming using IDE. | | | | | | | | | | |
| CO-6 | Students are able to understand, describe, and implement Stack in C programming using an IDE. | | | | | | | | | | |
| CO-7 | Students are able to understand, describe, and implement Queue in C programming using IDE. | | | | | | | | | | |
| CO-8 | Students are able to understand, describe, and implement Sorting: Bubble Sort, Insertion Sort, Selection Sort, in C programming using IDE. | | | | | | | | | | |
| CO-9 | Students are able to understand, describe, and implement Sorting: Shell Sort, Quick Sort, Merge Sort in C programming using IDE. | | | | | | | | | | |
| CO-10 | Students are able to understand, describe, and implement Searching: Linear Search & Binary Search in C programming using an IDE. | | | | | | | | | | |
| CO-11 | Students are able to understand, describe, and implement Tree: Binary Tree, Binary Search Tree, AVL Tree in C programming using IDE. | | | | | | | | | | |
| CO-12 | Students are able to understand, describe, and implement Graphs: BFS Algorithm & DFS Algorithm in C programming using IDE. | | | | | | | | | | |
|
| **Course Description** | | | This course studies and implements basic algorithm and programming concepts using the C language. The course material consists of an introduction to Algorithms, Data Structures, and the use of IDEs, Arrays, Pointers, & Structures, Single Linked List, Double Linked List, Circular Linked List, Stack , Queue, Sorting: Bubble Sort, Insertion Sort, Selection Sort, Shell Sort, Quick Sort, Merge Sort, Searching: Linear Search & Binary Search, Tree: Binary Tree, Binary Search Tree, AVL Tree, Graphs: BFS Algorithm & DFS Algorithm in programming to solve a particular problem. | | | | | | | | | | | |
| **Course Matter** | | | 1. Introduction: Algorithms, Data Structures, and IDEs. 2. Arrays, Pointers, & Structures 3. Single Linked List 4. Circular Linked List and Double Linked List 5. Stack 2 meetings 6. Queue 7. Sorting: Bubble Sort, Insertion Sort, Selection Sort 8. Sorting: Shell Sort, Quick Sort, Merge Sort 9. Searching: Linear Search & Binary Search 10. Tree: Binary Tree, Binary Search Tree, AVL Tree 11. Graphs: BFS Algorithm & DFS Algorithm | | | | | | | | | | | |
| **References** | | | **Main:** | | |  | | | | | | | | |
| 1. Kernighan, Brian W, & Ritchie, Dennis M. 1988. The Ansi C Programming Language Second Edition, Prentice-Hall. 2. Cipta Ramadhani. 2015. Dasar Algoritma & Struktur Data. Yogyakarta: ANDI. | | | | | | | | | | | |
| **Supporting:** | | |  | | | | | | | | |
| 1. Jobsheet for the practical data structure of the informatics engineering education program | | | | | | | | | | | |
| **Learning Media** | | | **Software:** | | | | | | **Hardware :** | | | | | |
| DEV C ++ | | | | | | LCD & Projector | | | | | |
| **Lecturer** | | | **Fadhli Ranuaharja, S.Pd., M.Pd.T.** | | | | | | | | | | | |
| **Prerequisites** | | |  | | | | | | | | | | | |
| **Weeks-** | **Sub-CO**  **(Expected Final Ability in each learning stage)** | | | | **Assessment Indicator** | | **Assessment Criteria** | **Learning Method, Students’ Learning Experience**  **[Time Allocation]** | | **Learning Material [Topic from Reference]** | | | | **Score (%)** |
| **(1)** | **(2)** | | | | **(3)** | | **(4)** | **(5)** | | **(6)** | | | | **(7)** |
| 1 | Students are able to understand, describe, and implement the basic concepts of Algorithms, Data Structures, and the use of IDEs. | | | | 1. The accuracy in describing the algorithmic concept of the data structure and the use of the IDE 2. The accuracy of implementing the algorithmic concept of the data structure and using the IDE | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task 1**: create material on the basic concepts of algorithms, data structures and the use of IDEs  Practical report  **[BT + BM: (1) x (2x70 ”)]** | | *Introduction*: Algorithms, Data Structures, and IDEs. | | | | **5%** |
| 2 | Students are able to understand, describe, and implement Arrays, Pointers, & Structures in C programming using an IDE. | | | | 1. Accurately describing arrays, pointers, and structures. 2. The precision of implementing arrays, pointers, and structures. | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-2**: create programs by implementing arrays, pointers, and structures in C programming  Practical report  **[BT + BM: (1) x (2x70 ”)]** | | *Arrays, Pointers, & Structures* | | | | **5%** |
| 3 | Students are able to understand, describe, and implement Single Linked List in C programming using IDE. | | | | 1. Accuracy in describing *Single Linked List in C programming using IDE.* 2. Accuracy in implementing *Single Linked List in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-3**: create a program by implementing Single Linked List in C programming using IDE.  Practical report  **[BT + BM: (1) x (2x70 ”)]** | | *Single Linked List* | | | | **10%** |
| 4 | Students are able to understand, describe, and implement Double Linked List in C programming using IDE. | | | | 1. Accuracy in describing *Single Linked List in C programming using IDE.* 2. Accuracy in implementing *Single Linked List in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task 4 -**: create a program by implementing Double Linked List in C programming using IDE.  Practical report  **[BT + BM: (1) x (2x70 ”)]** | | *Double Linked List* | | | | **10%** |
| 5 | Students are able to understand, describe, and implement Circular Linked List in C programming using IDE. | | | | 1. Accuracy in describing *Single Linked List in C programming using IDE.* 2. Accuracy in implementing *Single Linked List in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | Lectures:  Practicum  **[TM: 1x (2x100 ")]**  • **Task-5**: create a program by implementing a Circular Linked List in C programming using the IDE.  Practical report  **[BT + BM: (1) x (2x70 ”)]** | | *Circular Linked List* | | | | **10%** |
| 6 | Students are able to understand, describe, and implement Stack in C programming using an IDE. | | | | 1. Accuracy in describing *Stack in C programming using IDE.* 2. Accuracy in implementing *Stack in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-6**: create a program by implementing Stack in C programming using the IDE.  Practical report  **[BT + BM: (1) x (2x70 ”)]** | | *Stack* | | | | **10%** |
| 7 | **QUIZ # 1** | | | | | | | | | | | | | |
| 8 | **Mid Semester Exam (UTS) - Mid Test** | | | | | | | | | | | | | |
| 9 | Students are able to understand, describe, and implement Queue in C programming using IDE. | | | | 1. Accuracy in describing *Queue in C programming uses IDE.* 2. Accuracy in implementing *Queue in C programming uses IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-7**: create a program by implementing Queue in C programming using IDE.  Practical report   * **[BT + BM: (1) x (2x70 ”)]** | | *Queue* | | | | **10%** |
| 10 | Students are able to understand, describe, and implement Sorting: Bubble Sort, Insertion Sort, Selection Sort in C programming using IDE. | | | | 1. Accuracy in describing *Sorting: Bubble Sort, Insertion Sort, Selection Sort in C programming using IDE.* 2. Accuracy in implementing *Sorting: Bubble Sort, Insertion Sort, Selection Sort in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-8**: create a program by implementing Sorting: Bubble Sort, Insertion Sort, Selection Sort in C programming using the IDE.  Practical report   * **[BT + BM: (1) x (2x70 ”)]** | | *Sorting: Bubble Sort, Insertion Sort, Selection Sort* | | | | **10%** |
| 11 | Students are able to understand, describe, and implement Sorting: Shell Sort, Quick Sort, Merge Sort in C programming using IDE. | | | | 1. Accuracy in describing *Sorting: Shell Sort, Quick Sort, Merge Sort in C programming using IDE.* 2. Accuracy in implementing *Sorting: Shell Sort, Quick Sort, Merge Sort in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-9**: create programs by implementing Sorting: Shell Sort, Quick Sort, Merge Sort in C programming using the IDE.  Practical report   * **[BT + BM: (1) x (2x70 ”)]** | | *Sorting: Shell Sort, Quick Sort, Merge Sort* | | | | **10%** |
| 12 | Students are able to understand, describe, and implement Searching: Linear Search & Binary Search in C programming using an IDE. | | | | 1. Accuracy in describing *Searching: Linear Search & Binary Search in C programming using IDE.* 2. Accuracy in implementing *Searching: Linear Search & Binary Search in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-10** : create a program by implementing Searching: Linear Search & Binary Search in C programming using the IDE.  Practical report   * **[BT + BM: (1) x (2x70 ”)]** | | *Searching: Linear Search & Binary Search* | | | | **10%** |
| 13 | Students are able to understand, describe, and implement Tree: Binary Tree, Binary Search Tree, AVL Tree in C programming using IDE. | | | | 1. Accuracy in describing *Tree: Binary Tree, Binary Search Tree, AVL Tree in C programming using IDE.* 2. Accuracy in implementing *Tree: Binary Tree, Binary Search Tree, AVL Tree in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-11**: create a program by implementing Tree: Binary Tree, Binary Search Tree, AVL Tree in C programming using IDE.  Practical report   * **[BT + BM: (1) x (2x70 ”)]** | | *Tree: Binary Tree, Binary Search Tree, AVL Tree* | | | | **10%** |
| 14 | Students are able to understand, describe, and implement Graphs: BFS Algorithm & DFS Algorithm in C programming using IDE. | | | | 1. Accuracy in describing *Graphs: BFS Algorithm & DFS Algorithm in C programming using IDE.* 2. Accuracy in implementing *Graphs: BFS Algorithm & DFS Algorithm in C programming using IDE.* | | **Criteria:**   1. Practicum 2. Systematics 3. Presentation 4. Discussion Material 5. Task Report | * Lectures:   Practicum  **[TM: 1x (2x100 ")]**  • **Task-12**: create a program by implementing Graphs: BFS Algorithm & DFS Algorithm in C programming using IDE.  Practical report   * **[BT + BM: (1) x (2x70 ”)]** | | *Graphs: BFS Algorithm & DFS Algorithm* | | | | **10%** |
| 15 | **QUIZ # 2** | | | | | | | | | | | | | |
| 16 | **Final Semester Examination (UAS) - Final Test** | | | | | | | | | | | | | |