Index

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EXPERIMENT NAME** | Perform  Date | Submission  Date | Page  No. | Sign |
| 1. Design suitable data structures and implement pass-I of a two-pass assembler for pseudo-machine in Java using object-oriented feature. Implementation should consist of a few instructions from each category and few assembler directives. |  |  |  |  |
| 1. Implement Pass-II of two pass assembler for pseudo-machine in Java using object-oriented features. The output of assignment-1 (intermediate file and symbol table) should be input for this assignment. |  |  |  |  |
| 1. Design suitable data structures and implement pass-I of a two-pass macro-processor using OOP features in Java. |  |  |  |  |
| 1. Design suitable data structures and implement Pass-I and Pass-II of a two-pass macro-processor. The output of Pass-I (MNT, MDT and intermediate code file without any macro definitions) should be input for Pass-II. |  |  |  |  |
| 1. Write a program to create Dynamic Link Layer for any mathematical operations and write an application program to write it. (Java Native Interface/Use VB or VC++) |  |  |  |  |
| 1. Write a program to solve Classical Problems of Synchronization using Mutex and Semaphore. |  |  |  |  |
| 1. Write a Java program (using OOP features) to implement following scheduling algorithms: FCFS, SJF (Preemptive), Priority (Non-Preemptive) and Round Robin (Preemptive). |  |  |  |  |
| 1. Write a Java program to implement Banker’s Algorithm. |  |  |  |  |
| 1. Write a Program to simulate Memory placement strategies – best fit, first fit, next fit and worst fit. |  |  |  |  |
| 1. Write a java program to implement Page Replacement Policies LRU & OPT. |  |  |  |  |