|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sl. No | Syllabus | Curriculum | Deployment Strategy and  Tool | Cross-cutting issues  integrated | PO, PSO and CO | Attainment Verification |
| 1. | UNIX SYSTEM PROGRAMMING | 1. This course will prepare students to develop software in and for Linux/UNIX environments. Topics to be covered include basic operating system concepts, effective command line usage, shell programming, the C language, programming development tools, system programming, network programming (client-server model and sockets), and GUI programming.  2.  The course is primarily about system programming. In particular, it covers the following parts of the  kernel API: general I/O structure, device and terminal control, the le system interface, process and thread  management, signals and inter-process communication methods. It also covers a bit about event driven  programming and the curses library.  This course devotes a small amount of time to shells and UNIX tools, and significant emphasis on system  programming, | 1. Chalk and   Talk method   1. PPT | * Business   Ethics   * Human   values | PO1:Engineering Knowledge  PO2:Problem Analysis  PO3:Design/Development Of Solutions |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | PSO1:Professional Skills  PSO2:Problem Solving Skill |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | **CO1** Analyze the difference between various standards and compare them with Unix standard.  **CO2** Implement program routine using various APIs to perform specific function.  **CO3** Illustrate the Unix environment for programming and Model system calls.  **CO4** Demonstrate applications using various IPC mechanisms. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

