NCEAC.FORM.001-D

# COURSE DESCRIPTION FORM

**INSTITUTION** National University of Computer and Emerging Sciences

# PROGRAM (S) TO BE

**EVALUATED**

BS Computer Science

# A. Course Description

(Fill out the following table for each course in your computer science curriculum. A filled-out form should not be more than 2-3 pages.)

|  |  |
| --- | --- |
| **Course Code** | CS2006 |
| **Course Title** | Operating Systems |
| **Credit Hours** | 3+1 |
| **Prerequisites by Course(s) and Topics** | ITC & Data Structures |
| **Assessment Instruments with Weights** (homework, quizzes, midterms, final, programming assignments, lab work, etc.) | Midterms 30%  Assignments & class activities 10% Projects 10%  Final Exam 50% |
| **Course Coordinator** | Dr. Ghufran Ahmed |
| **URL (if any)** | https://classroom.google.com/u/0/c/Mzg4NzU1NTM4ODcy |
| **Current Catalog** | Introduction to operating systems, Operating system structures and design, Process |
| **Description** | Concepts, Inter-process Communication, Process scheduling, FCFS, SJF, SRTF, |
|  | Priority, RR, multiprocessor, real-time, thread scheduling and threads security by scope of threads. Memory management |
|  | techniques segmentation, fragmentation, paging, structure of page table, Virtual |
|  | memory, COW, page replacement algorithms, FCFC, optimal, LRU, second |
|  | chance, Frame allocation, thrashing, Kernel memory, buddy, slab. Process |
|  | Synchronization, Peterson solution, test and set instruction, mutex lock, |
|  | semaphore, classical problems, bounded buffer, reader writer, dinning |
|  | philosopher. Deadlock detection, prevention, avoidance method, banker’s |
|  | algorithm resource request algorithm and protection and security of resources and processes. Disk scheduling FCFS, SSTF, SCAN, |
|  | CSCAN, LOOK, CLOOK, protection and security introduction. |

NCEAC.FORM.001-D

|  |  |
| --- | --- |
| **Textbook** (or **Laboratory Manual** for Laboratory Courses) | * Operating system Concepts by Silberschatz 10th Edition. * Linux Fundamentals by Paul Cobbaut * Shell Scripting by Steve Parker * System Software (An Introduction to System Programming) 3rd Edition by Leland L. Beck |
| **Reference Material** | * Modern Operating Systems by Andrew S. Tanenbaum * Operating System Internal Designs & Principles by William Stallings (latest Edition) * How Linux Works by Brian Ward 2nd Edition. * System Programming with C and Unix by Adam Hoover |

NCEAC.FORM.001-D

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Goals** |  | **A. Course Learning Outcomes (CLOs)** | | | | | | | | | | | |  |
|  | **CLO** | | **Course Learning Outcome (CLO)** | | **Domain** | **Taxonomy Level** | **PLO** | **Tools** | | |  |  |
| 01 | | Understand the  characteristics of different structures of the Operating Systems and identify the core  functions of the Operating Systems. | | Cognitive | 3 | 2 | **CA, M1, F** | | |
| 02 | | Analyze and evaluate the algorithms of the core functions of the Operating Systems and explain the major performance issues about the core functions. | | Cognitive | 4 | 3 | **CA, M2, F** | | |
| 03 | | Demonstrate the knowledge in applying system software and tools available in modern  operating systems. | | Cognitive | 5 | 5 | **A, P, F** | | |
| *Tool: A = Assignment, Class Activities = CA, P=Project, M = Midterm, F=Final,* | | | | | | | | | |
|  | | | | | | | | | | | |
|  |  | **B. Program Learning Outcomes** | | | | | | |  |  | |
|  | For each attribute below, indicate whether this attribute is covered in this course  or not. Leave the cell blank if the enablement is little or non-existent. | | | | | | |  |
| 02 | | | **Problem Analysis:** Identify, formulate, research literature, and  analyze complex computing problems, reaching substantiated conclusions using first principles of mathematics, natural sciences, and computing sciences. | | | | | |
| 03 | | | **Design/Develop Solutions:** Design solutions for complex  computing problems and design systems, components, and processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations. | | | | | |
| 05 | | | **Modern Tool Usage:** Create, select, and apply appropriate  techniques, resources and modern computing tools, including prediction and modelling for complex computing problems. | | | | | |
| **C. Relation between CLOs and PLOs**  (CLO: Course Learning Outcome, PLOs: Program Learning Outcomes) | | | | | | | | |

NCEAC.FORM.001-D

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | | **PLOs** | | | | | | | | | | | | |  |  |
| **1** | **2** | **3** | **4** | | **5** | **6** | **7** | | **8** | **9** | | **10** |
| **CLOs** | 1 |  |  |  |  | |  |  |  | |  |  | |  |
| 2 |  |  |  |  | |  |  |  | |  |  | |  |
| 3 |  |  |  |  | |  |  |  | |  |  | |  |
| 4 |  |  |  |  | |  |  |  | |  |  | |  |
|  | 5 |  |  |  |  | |  |  |  | |  |  | |  |
|  | 6 |  |  |  |  | |  |  |  | |  |  | |  |
|  | 7 |  |  |  |  | |  |  |  | |  |  | |  |
|  | | | | | | | | | | | | | | | | |
| **Topics Covered in the Course, with Number of Lectures on Each Topic** (assume 15-week instruction and one- hour lectures) |  | **1. Topics to be covered:** | | | | | | | | | | | | | |  | | | |
| List of Topics | | | | | | | No. of Weeks | | Contact Hours | | CLO | | |
| **Introduction to Operating system:** basic OS definition, computer organization, I/O, DMA, mass storage, protection, UMA and NUMA architecture, symmetric & asymmetric clustering, security, computing platforms. | | | | | | | **1.5** | | **4** | | **1** | | |
| **Operating system structure:** basic concept CLI, GUI, scripts, API, system programming &  goals, OS design principles. | | | | | | | **1.5** | | **4.5** | | **1,3** | | |
| **Process Concept:** basic concept, scheduler types, Queues, process  creation, interprocess communication methods. | | | | | | | **1** | | **4** | | **1,2,3** | | |
| **Process scheduling Algorithm:** pre-emptive & non –preemptive, FCFS, SJF, SRTF, Priority, RR,  multiprocessor, real –time scheduling. | | | | | | | **1.5** | | **4** | | **1,2,3** | | |

NCEAC.FORM.001-D

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Multi-threaded Programming:** basic control blocks, thread models, thread concepts, process vs. threads, data and task parallelism, Amdahl’s law, pthread APIs, OpenMP, threads security by scope of threads | **1.5** | **4.5** | **2,3,** | |  |
| **Memory Management:**  basic memory definition, dynamic allocation, problems of dynamic allocation, swapping, fragmentation, segmentation, paging, structure of page tables,  System architecture | **2** | **6** | **1,2** | |
| **Virtual Memory:** basic VM concept, demand paging, COW, page replacement algos, FIFO, optimal, LRU, second chance, frame allocation, thrashing, kernel  memory, buddy, slab allocation. | **2** | **6** | **1,2** | |
| **Process Synchronization:** concurrency, race condition, critical section, Peterson solution, test and set instruction, mutex, semaphore. Classical problems such as bounder buffer, reader  writer, dinning philosopher. | **2** | **6** | **2,3,** | |
| **Deadlock:** basic concept**,** detection, prevention, avoidance**,**  banker’s algorithm. protection and security of resources and processes. | **1** | **3** | **1,2** | |
| **Security:**  security threats and attacks, fundamentals of encryption, authentication, and hashing, various countermeasures to security attacks | **1** | **3** | **1,2** | |
| Total | **15** | **45** |  | |
| **Laboratory Projects/Experiments Done in the Course** |  | **Lab 1:** Introduction to Linux & Basic Linux Commands | | | |  | |
| **Lab 2:** Basic Linux Commands | | | |
| **Lab 3:** Shell Programming/Scripting | | | |
| **Lab 4:** Shell Programming/Scripting | | | |
| **Lab 5:** System Call related to Process Management | | | |
| **Lab 6:** Inter- Process Communication | | | |
| **Lab 7:** Kernel Configuration | | | |
| **Lab 8:** Mid Exam | | | |
| **Lab 9:** Multithread Programming in Pthreads | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **Lab 10:** Creating a module in Kernel | | | |  |
| **Lab 11:** Multithread Programming in OpenMP (shared memory) | | | |
| **Lab 12:** Semaphores in Linux | | | |
| **Lab 13:** Signals in Linux | | | |
| **Lab 14: Revisions** | | | |
| **Lab 15:** Final Lab Exam | | | |
| **Programming Assignments Done in the Course** | Programming assignment is given to students. | | | | | |
| **Class Time Spent on**  (In credit hours) | **Theory** | | **Problem Analysis** | **Solution Design** | **Social and Ethical Issues** | |
| 20 | | 30 | 40 | 10 | |
| **Oral and Written Communications** | Every student is required to submit a project along with its report of not more than 8 pages. | | | | | |

## Instructor Name: ANAUM HAMID

**Instructor Signature**

## Date 31st, January 2022