# The Hashemite University
# Faculty of Engineering
# Computer Engineering Department
## Operating SystemsCourse Syllabus
## (110408455)
## Spring 2016

| | | |
|---|---|---|
| **Instructor:** | Dr. Dheya Mustafa | **Assessment and Course Grade:**<br>• Lab 30%<br>• Project 30%<br>• Final Exam 40% |
| **Office No.:** | E3048 | |
| **Email:** | dheya@hu.edu.jo | |
| **Office Hours:** | 9:00 –11:00 Su,Tu, Thur, or by appt | |

## Course Description

Operating system course is one of the fundamental courses. The course familiarizes the student with basic knowledge of computer operating systems. It is designed to give students in Computer Science and Computer Engineering insight into the design, implementation and use of computer operating systems. We consider the impact of OS design decisions on application programs, and explore the wide range of possible system designs for different goals. The course focuses on some issues and problem in which the operating system provides a solution for.

There is a significant emphasis on C programming (class examples, lab sessions, projects and may quizzes). This will improve the student's programming skills and will support understanding the theoretical concepts discussed in the class.

## Textbooks

1. **Operating Systems Concepts**, by Silberschatz, Galvin, and Gagne, 8th Edition, 2009, John Wiley.

## Course Objectives

Successful completion of this course should lead to the following learning outcomes:

- The ability to state clearly what is an operating system and the software components that made up any operating system.
- The ability to state the different tasks in which the operating system is responsible for.
- The ability to understand the operating system as a running environment and relates the concept of processes and threads to this task of operating system.
- The ability to understand how the operating system performs the scheduling between the jobs.
- The ability to analyze the synchronization problem and the methods in which the operating system handles this problem.
- The ability to define the system deadlock state conditions and solutions.
- The ability to understand the operating system memory management techniques.
- The ability to understand operating system file structure and mass storage managements.
- The ability to deal with open source operating systems.
- Theability to write C codes to mimic the behaviour of real operating services such as: scheduler, dispatcher, synchronization and memory managment.

## Course Plan

| Topic | Chapter No. |
|---|---|
| **Introduction**<br>• What Operating Systems Do, Computer-System Organization, Computer-System Architecture, Operating-System Structure, Operating-System Operations | **1** |
| **Operating System Structures**<br>• Operating-System Services, User Operating-System Interface, System Calls, Types of System Calls, System Programs, Design and Implementation of Operating-System, Operating-System Structure | **2** |
| **Processes**<br>• Process Concept, Process Scheduling, Operations on Processes, Inter-process Communication, Communication in Client–Server Systems | **3** |
| **Threads**<br>• Overview, Multithreading Models, Thread Libraries, Threading Issues, Threads in Windows and Linux | **4** |
| **CPU Scheduling**<br>• Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Thread Scheduling, Multiple-Processor Scheduling, Algorithm Evaluation | **5** |
| **Process Synchronization**<br>• Background, The Critical-Section Problem, Peterson’s Solution, Synchronization Hardware, Semaphores, Monitors | **6** |
| **Deadlocks**<br>• System Model, Deadlock Characterization, Methods for Handling Deadlocks<br>• Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock | **7** |
| **Main Memory Management**<br>• Background, Swapping, Contiguous Memory Allocation, Paging, Segmentation | **8** |
| **Virtual memory Management**<br>• Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory | **9** |
| **File System** | **10** |
| **Secondary storage** | **12** |
| **I/O Systems** | **13** |
| **Distributed operating systems** | **16** |

## Lab Work

1. attendance is mandatory
2. strong programming skills is required

## Projects

1. paper review: chapter 21,22,23,appendix A, B,C. : each student must write a samary for selected topic and present it in class

2. OS programming: strong Java programming skills is required.