

CSC 562: Principles of Operating Systems

Fall 2017

Syllabus

Instructor: Jing Deng

Office: Petty Building 164

Office Hours: Wednesdays 10:00AM-12:00noon

E-mail: jing.deng@uncg.edu (preferred contact)

Prerequisites: Grades of at least C in CSC 261 and CSC 340 (or equivalent if transfer student). Successful completion of CSC 561 will be helpful. The course is designed for junior/senior-level undergraduate computer science majors and graduate students.

Catalog Description: Techniques and strategies used in operating system design and implementation: managing processes, input/output, memory, scheduling, file systems, and protection.

Course Objectives: The aim of this course is to teach the concepts and principles of modern operating systems, and to provide opportunities to relate theoretical principles with operating system implementation. Specifically,

- Learn about processes and process management
- Learn about concurrency and synchronization
- Learn about memory management schemes
- Learn about file system and secondary storage management
- Practice, through programming assignments, the use of implementation of the concepts above
- (Graduate students only) Perform paper reviews in related state-of-the-art technical literature.

Course Outcomes: By the end of the course, students should be able to demonstrate understanding of the basic principles and techniques of operating systems covered in the course, and should be able to implement system programs. Specifically,

- CO1: Demonstrate knowledge in fundamental operating system abstractions such as processes, threads, and process management
- CO2: Demonstrate knowledge in semaphores, monitors, conditions, deadlocks, IPC abstractions, shared memory regions, etc.
- CO3: Demonstrate knowledge in principles of memory management
- CO4: Demonstrate knowledge in file abstraction and storage management
- CO5: Be able to apply knowledge outlines in the above to design and develop system programs based on operating system abstractions and to write correct concurrent programs/software, including basic resource management techniques (scheduling or time management, space management) and considering issues such as performance and fairness objectives, and avoiding deadlocks.