This course supports the assessment for Operating Systems for Programmers. The course covers 5 competencies and represents 3 competency units.

Introduction

Overview
Operating systems act as an interface between applications and hardware. Although today's systems will often dynamically adjust CPU scheduling and memory usage, programmers must still have a deep understanding of operating systems in order to create applications that maximize performance and minimize the impact on the environment. By minimizing the swapping that may occur with both the CPU, main memory, and secondary storage, many efficiencies can be realized.

This course covers operating systems from the perspective of a programmer, including the placement of the operating system in the layered application development model, as well as an in-depth view to each of the services operating systems provide.

Watch the following video for an introduction to this course:

Note: To download this video, right-click the following link and choose "Save as...": download video.

Watch the following video for information about getting started in this course:

Note: To download this video, right-click the following link and choose "Save as...": download video.

Competencies
This course provides guidance to help you demonstrate the following 5 competencies:

- **Competency 4022.1.1: Introduction to Operating Systems**
  The graduate describes operating systems, their functions, and their structure.
- **Competency 4022.1.2: Process Management**
  The graduate describes processes and threads and their relationship to multithreading and parallel programming.
- **Competency 4022.1.3: Memory Management**
  The graduate explains the different approaches to memory management and how they affect CPU utilization.
- **Competency 4022.1.4: Storage Management**
  The graduate describes different file systems and I/O algorithms.
- **Competency 4022.1.5: Protection and Security**
  The graduate describes mechanisms used by the operating system for protection and security and how they relate to software applications.

Course Instructor Assistance
As you prepare to successfully demonstrate competency in this subject, remember that course instructors stand ready to help you reach your educational goals. As subject matter experts, mentors enjoy and take pride in helping students become reflective learners, problem solvers, and critical thinkers. Course instructors are excited to hear from you and eager to work with you.

Successful students report that working with a course instructor is the key to their success. Course instructors are able to share tips on approaches, tools, and skills that can help you apply the content you're studying. They also provide guidance in assessment preparation strategies and troubleshoot areas of deficiency. Even if things don't work out on your first try, course instructors act as a support system to guide you through the revision process. You should expect to work with course instructors for the duration of your coursework, and you are encouraged to contact them as soon as you begin. Course instructors are fully committed to your success!

Preparing for Success

The information in this section is provided to detail the resources available for you to use as you complete this course.

Learning Resources

The learning resources listed in this section are required to complete the activities in this course. For many resources, WGU has provided automatic access through the course. However, you may need to manually enroll in or independently acquire other resources. Read the full instructions provided to ensure that you have access to all of your resources in a timely manner.

Automatically Enrolled Resources

You can access the learning resources listed in this section by clicking on the links provided throughout the course. You may be prompted to log in to the WGU student portal to access the resources.

Wiley Engage

You will use the following Wiley Engage course:

- Operating Systems for Programmers

The resource will be linked at the activity level.

Other Resources

You will use the following learning resources for this course.

Linux
Some of the exercises in this course will require Linux. If your computer runs MS Windows, you can install a virtual machine to run Linux. The following steps will direct you on how to prepare your Linux lab environment. If you need additional help, module 1 of the Wiley Engage course will give you more detailed directions.

1. Download VMware Player for Windows or VMware Fusion for Mac from MS Imagine tool or from: http://www.vmware.com/download/player/ (under Desktop & End-User Computing section)
2. Download a Linux version VMware Appliance from: http://vmware.com/appliances/ (you can use the Search box to search for Linux or a particular distribution, like Ubuntu)
3. Install VMware Player (if running Windows) or VMware Fusion (if running OS X) on your computer and boot the Linux VMware Appliance
4. Explore the Linux user interface and write a summary of the differences, perceived strengths and weaknesses of the “user experience” from Linux to another operating system you are familiar with (Windows, OS X, etc).

Complete the Preassessment

If you believe you have previous knowledge of some or all topics covered in this course, start by taking the preassessment before you begin and use its results to focus your studies.

- Complete the preassessment located in the Assessment tab.

Course instructors can help you develop a study plan based on your preassessment results.

Pacing Guide

The pacing guide suggests a weekly structure to pace your completion of learning activities. It is provided as a suggestion and does not represent a mandatory schedule. Follow the pacing guide carefully to complete the course in the suggested timeframe.

Week 1

- Preparing for success
    - Wiley Engage: Operating Systems for Programmers Note: You will use the link above for all activities in this pacing guide
    - Download the free VMware Player tool from:http://www.vmware.com/download/player/ (under Desktop & End-User Computing section)
    - o Download a Linux version VMware Appliance from:http://vmware.com/appliances/ (you can use the Search box to search for Linux or a particularly distribution, like Ubuntu)
    - o Install VMware Player on your computer and boot the Linux VMware Appliance
    - o Explore the Linux user interface and write a summary of the differences, perceived strengths and weaknesses of the “user experience” from Linux to another operating system you are familiar with (Windows, OS X, etc).
- Introduction to Operating Systems
    - Complete all activities in Module 1: Introduction to Operating Systems of the Wiley Engage course.

Week 2

- Process and Threads
    - Complete all activities from Topic 1: Processes and Threads in Module 2: Process Management of the Wiley Engage course.
- CPU Scheduling
    - Complete all activities from Topic 2: CPU Scheduling in Module 2: Process Management of the Wiley Engage course.

Week 3

- Memory Management
    - Complete all activities from Topic 1: Memory Management in Module 3: Memory Management of the Wiley Engage course.
- Virtual Memory
    - Complete all activities from Topic 2: Virtual Memory in Module 3: Memory Management of the Wiley Engage course.

Week 4

- Mass Storage
    - Complete all activities from Topic 1: Mass Storage in Module 4: Storage Management of the Wiley Engage course.
- File Systems
    - Complete all activities from Topic 2: File Systems in Module 4: Storage Management of the Wiley Engage course.
- I/O Systems
    - Complete all activities from Topic 3: I/O Systems in Module 4: Storage Management of the Wiley Engage course.

Week 5

- Protection and Security
    - Complete all activities in Module 5: Protection and Security of the Wiley Engage course.

Week 6

- Final Steps
    - Complete the PAG7 pre-assessment scoring 90% or better
    - Complete the ACO1 objective assessment

Note: This pacing guide does not replace the course. Please continue to refer to the course for a comprehensive list of the resources and activities.

Introduction to Operating Systems

Operating systems are a fundamental part of any computing system from smart phones to

servers. Because of their central role in the layered development approach, operating systems are involved in almost every process that involves input, output, processing, and storage.

Introduction to Operating Systems

An operating system is the software that controls computer hardware resources, including memory allocation, storage allocation, processor scheduling and input/output, as well as providing common services for other computer programs. Computer hardware in and of itself is not generally very useful, unless there is a way to input data, perform tasks or calculations, and store or output the results of those tasks or calculations in a meaningful fashion. This topic covers the basic function, structure, and design of an operating system.

This topic addresses the following competency:

- **Competency 4022.1.1: Introduction to Operating Systems**
  The graduate describes Operating Systems, their functions, and their structure.

This topic highlights the following objectives:

- Explain the role of the operating system.
- Classify the main services that operating systems provide.
- Describe system calls.
- Classify system services.
- Describe the operating-system design process.
- Identify important considerations of operating system design to enhance maintenance and portability.
- Explain the role of the bootstrap procedure.

Complete: Wiley Engage Module 1

As you complete the listed material below, keep in mind the following key points:

- Operating systems are not only present in personal computers; mainframes, servers, and smartphones are some of the other devices that run operating systems.
- The bootstrap procedure is present in all devices but stored and executed differently in some of them.
- Operating systems provide multiple services. Can you name a few?

Complete all activities in Module 1: Introduction to Operating Systems of the Wiley Engage course.

Process Management

A running program on a computer is called a process. Early computers were capable of running a single program or process at a time, with that program utilizing all available system resources. Operating systems in modern computers can manage the sharing of system resources, so that multiple processes can run concurrently. This topic expounds on what exactly a process is and how it works in a modern, multiplexed environment.

Process and Threads

A process is a unit of work in a modern, time-sharing system, also often referred to as a job. A process includes all of the components of a program in execution; including program counters, process stack and data section. In early computer systems, a process had a one-to-one relationship with a thread of instructions being executed. Modern operating systems allow a process to have multiple threads of execution.

This topic addresses the following competency:

- **Competency 4022.1.2: Process Management**
  The graduate describes processes and threads and their relationship to multi-threading and parallel programming.

This topic highlights the following objectives:

- Describe a process.
- Describe the process control block and its function.
- Explain the concept of a thread.
- Explain process scheduling.
- Explain how the operating system uses the scheduler to execute tasks.
- Describe the process of context switching.
- Identify tools and techniques for debugging processes and kernel failures.
- Explain the system generation procedure.

Complete: Wiley Engage Module 2

As you complete the listed material below, keep in mind the following key points:

- Processes can communicate with each other using different methods. The most common methods are shared memory or message passing.
- Orderly execution of processes in a multithreading system is very complex and must be carefully controlled.
- There are multiple algorithms to schedule CPU utilization; each of them has their own challenges.

Complete all activities from **Topic 1: Processes and Threads** in Module 2: Process Management of the Wiley Engage course.

CPU Scheduling

This topic introduces CPU scheduling in a multiprogramming environment. Multiprogramming allows for greater processor utilization, but with that comes the challenge of managing CPU allocation. There are multiple strategies for dealing with this complexity, including various CPU-scheduling algorithms. This topic will cover the logic behind choosing a CPU-scheduling algorithm and examine how operating systems implement them.

This topic addresses the following competency:

- **Competency 4022.1.2: Process Management**

The graduate describes processes and threads and their relationship to multi-threading and parallel programming.

This topic highlights the following objectives:

- Describe the circumstances under which the CPU-scheduling decisions take place.
- Describe scheduling criteria.
- Differentiate between scheduling algorithms.
- Explore scheduling issues.
- Explain approaches to multiple-processor scheduling.
- Analyze load-balancing in a multiple-processors environment.

Complete: Wiley Engage Module 2

As you complete the listed material below, keep in mind the following key points:

- Processes can communicate with each other using different methods. The most common methods are shared memory or message passing
- Orderly execution of processes in a multithreading system is very complex and must be carefully controlled.
- There are multiple algorithms to schedule CPU utilization; each of them has their own challenges.

Complete all activities from **Topic 2: CPU Scheduling** in Module 2: Process Management of the Wiley Engage course.

Memory Management

In previous competencies, we have learned about how the operating system manages the sharing of computer-system resources to maximum the efficient utilization of those resources. This section covers the concept of memory management, various approaches to memory management, and their strengths and weaknesses.

Memory Management

Memory is a key component in a computer-system, directly interacting with the CPU for retrieval and storage of instructions. In previous modules, we learned that, in a modern, multiprogramming environment, the OS manages access to CPU resources from multiple programs via scheduling. Similarly, the OS manages how these programs interact with memory, organizing and allocating memory to maximize performance and system utilization.

This topic addresses the following competency:

- **Competency 4022.1.3: Memory Management**
  The graduate explains the different approaches to memory management and how they affect CPU utilization.

This topic highlights the following objectives:

- Describe the various ways to manage memory.
- Explain the concept of swapping as it relates to main memory to backing store.
- Explain the concept of swapping in a mobile system and its limitations.
- Discuss the basic concept of contiguous memory allocation and its role in memory.
- Explain principles and limitations of memory allocation.
- Discuss the concepts of fragmentation as it relates to memory allocation.
- Explain segmentation and segmentation hardware.
- Explain the principles and limitations of the paging and hardware Translation Lookaside Buffer (TLB).
- Explain principles of memory protection.
- Identify key concepts of shared pages.
- Identify some of the key techniques for structuring paging tables.

Complete: Wiley Engage Module 3

As you complete the listed material below, keep in mind the following key points:

- There are multiple ways to manage memory and each of them has its advantages and disadvantages depending on the platform.
- Memory management in a mobile environment is particularly challenging.
- Same as with storage, memory fragmentation can cause performance lose.

Complete all activities from **Topic 1: Memory Management** in Module 3: Memory Management of the Wiley Engage course.

Virtual Memory

In a multiprogramming environment, an OS uses various strategies to keep multiple processes in memory simultaneously. Virtual memory is a technique by which a process can be executed even if it is not fully in memory, which allows the abstraction of physical memory into a larger array of storage and allows programs to be larger than actual, physical memory.

This topic addresses the following competency:

- **Competency 4022.1.3: Memory Management**
  The graduate explains the different approaches to memory management and how they affect CPU utilization.

This topic highlights the following objectives:

- Describe the various ways to manage memory.
- Explain the concept of swapping as it relates to main memory to backing store.
- Explain the concept of swapping in a mobile system and its limitations.
- Discuss the basic concept of contiguous memory allocation and its role in memory.
- Explain principles and limitations of memory allocation.
- Discuss the concepts of fragmentation as it relates to memory allocation.
- Explain segmentation and segmentation hardware.
- Explain the principles and limitations of the paging and hardware translation lookaside

buffer (TLB).
- Explain principles of memory protection.
- Identify key concepts of shared pages.
- Identify some of the key techniques for structuring paging tables.

Complete: Wiley Engage Module 3

As you complete the listed material below, keep in mind the following key points:

- There are multiple ways to manage memory and each of them has its advantages and disadvantages depending on the platform.
- Virtual memory allows the OS to have more files open than the physical memory amount allows for.
- Swapping algorithms are used to move data between virtual and physical memory.
- As a programmer, you need to consider that almost all applications (except those which implement DMA) do not erase from memory. Once a variable completes its lifecycle, it is marked for deletion and will be deleted the next time the OS executed the garbage collection mechanism.

Complete all activities from **Topic 2: Virtual Memory** in Module 3: Memory Management of the Wiley Engage course.

Storage Management

Previous chapters discussed memory in a computer system and it's interaction with the CPU for retrieval and storage of instructions. This section covers secondary storage, its physical structure, secondary-storage devices, the concept of mass-storage devices and their performance, as well as disk scheduling algorithms.

Mass Storage

Mass storage is differentiated from memory in that mass storage is persistent and can be used for short-term or long-term storage. Mass storage is represented by physical storage in a computer system, most often in the form of a hard disk. Modern computers can utilize various forms of mass-storage devices, from traditional physical hard disk drives to solid-state-drives (SSD) to magnetic tapes. It is important to understand physical-storage device structure, performance, and how the OS manages these devices.

This topic addresses the following competency:

- **Competency 4022.1.4: Storage Management**
  The graduate describes different file systems and I/O algorithms.

This topic highlights the following objectives:

- Describe the physical structure of storage devices.
- Identify the key concepts of magnetic disk performance.
- Identify uses for tertiary storage.
- Identify different disk scheduling algorithms.

- Describe disk management.
- Describe the use of swap space.

Complete: Wiley Engage Module 4

As you complete the listed material below, keep in mind the following key points:

- There are many measurements associated with the performance of storage devices; usually the RPMs is the advertised. Transfer rate, latency, and seek time also determine performance.
- USB drives are a type of removable storage.
- Swap space can either be done as a raw partition on the disk drive or through the file system. Raw is almost always faster.

Complete all activities from **Topic 1: Mass Storage** in Module 4: Storage Management of the Wiley Engage course.

File Systems

This topic covers the various aspects of file systems and the major directory structures, discussing the semantics of sharing files among multiple processes, users and computers. It also discusses ways to handle file protection and controlling who may access files and how they may be accessed.

This topic addresses the following competency:

- **Competency 4022.1.4: Storage Management**
  The graduate describes different file systems and I/O algorithms.

This topic highlights the following objectives:

- Explain the functions of a file system.
- Describe the system interfaces to a file system.
- Understand the tradeoffs in creating a file system.
- Understand file system protection challenges.
- Understand the basics of network file systems.

Complete: Wiley Engage Module 4

Before you complete the listed material below, do the following:

- There are many file systems utilized in the market. Make a list of the file systems used by your personal devices (Computers, tablets, smartphones, SD card, USB cards) and try to understand why that is (or if it is not) the most appropriate for that device.
- Highlight the file systems that repeat in the list made in the previous step.

Complete all activities from **Topic 2: File Systems** in Module 4: Storage Management of the Wiley Engage course.

I/O Systems

This chapter discusses the structure of an OS's I/O subsystems and discussed the principles and complexities of I/O hardware, including an explanation of the performance aspects of I/O hardware and software.

This topic addresses the following competency:

- **Competency 4022.1.4: Storage Management**
  The graduate describes different file systems and I/O algorithms.

This topic highlights the following objectives:

- Explain the functions of a file system.
- Describe the system interfaces to a file system.
- Understand the tradeoffs in creating a file system.
- Understand file system protection challenges.
- Understand the basics of network file systems.

Complete: Wiley Engage Module 4

As you complete the listed material below, keep in mind the following key points:

- I/O systems are designed in layers.
- There are several I/O ports, each with its specific purpose.

Complete all activities from **Topic 3: I/O Systems** in Module 4: Storage Management of the Wiley Engage course.

Protection and Security

For an operating system to effectively manage system resources and to allocate resources between different processes, it must be able to ensure that only processes that have gained proper authorization access those resources. Protection in this context refers to OS mechanisms for controlling the access of programs, processes, and users to resources as defined by the computer system. For a computer system, this is an internal problem.

Security in the context of computer systems refers to mechanisms to protect against unauthorized access, malicious destruction or alteration, and accidental introduction of inconsistency. An OS uses various strategies and technologies to detect and guard against such attacks. Security, unlike protection, must also consider the external environment in which the system operates.

Protection and Security

The more complex and sophisticated a computer environment, the greater the need for protection and security. Complex multitasking and multi-user environments demand an OS that can adequately manage and control resource allocation, as well as an OS that can define and maintain access authorization and data consistency.

This topic addresses the following competency:

- **Competency 4022.1.5: Protection and Security**
  The graduate describes mechanisms used by the operating system for protection and security and how they relate to software applications.

This topic highlights the following objectives:

- Describe the goals of protection.
- Distinguish between a mechanism and a policy.
- Describe the principles of protection.
- Identify the role of access control in the implementation of protection goals.
- Describe basic security terminology.
- Describe the different types of attacks.
- Identify the different methods of attack.
- Identify the different levels of security measures.
- Describe the different types of program threats.
- Describe the different types of system and network threats.
- Describe the basic cryptography terms and concepts.
- Explain the purpose of digital signatures and hash functions that provide data integrity.
- Describe the purpose of a virtual private network (VPN).

Complete: Wiley Engage Module 5

As you complete the listed material below, keep in mind the following key points:

- The main difference between a policy and a mechanism is that the policy determines what to do (or not do) and the mechanism determines how to enforce it.
- There are many types of attacks. The most common are: breach of confidentiality, breach of integrity, breach of availability, theft of service, and denial-of-service (DoS).
- Since the most common type of hacking is social engineering, security policies are not complete without a human factor.
- Restricting physical access to assets to authorized users only must be included in any company’s security policy.

Complete all activities in Module 5: Protection and Security of the Wiley Engage course.

Final Steps

Congratulations on completing the activities in this course! This course has prepared you to complete the assessment associated with this course. If you have not already been directed to complete it, schedule and complete the assessment now.

Complete the Preassessment

The preassessment can help you determine your level of preparation. It is highly recommended that you pass the preassessment before attempting the final exam.

- Complete the preassessment located in the Assessment tab.

Course instructors can help you develop a review plan based on your preassessment results.

First Attempt Checklist

One of the many things that makes WGU unique is its competency-based education model. If you know the material, all you have to do is prove it by passing the exam. If you can do this, you can accelerate the receipt of your degree.

To make sure you have the best chance possible to pass the exam on your first attempt, the following steps should be completed successfully before you take it:

1. Complete all five modules in the Wiley Engage Operating Systems for Programmers course including required chapter readings and exercises.
2. Complete all five post quizzes in the Wiley Engage Operating Systems for Programmers course.
3. Complete all required lab assignments in the Wiley Engage course utilizing VMWare and Linux.
4. Take the preassessment for the course, striving for 85% or higher. Use the coaching report to determine where you still need to review and reach out to the course instructors with any questions you have.

If you have completed the steps above and you feel comfortable with all of the concepts presented, you are most likely ready to attempt the exam.

If you fail your first attempt, you will be required to contact the course instructor to see what went wrong and how you can prepare to ensure a successful second attempt. After determining you are ready, your course instructor will approve your request once to make another exam attempt.