

Operating Systems

Level: 3

Career Cluster: Digital Technology

Program: Information Technology

Pathway(s): Networking Systems and Security

Course Catalog Description

Course that develops students' understanding of how operating systems manage hardware, software, and user interaction. Students apply system configuration, resource management, security practices, troubleshooting, and virtualization concepts to maintain efficient and secure computing environments while demonstrating professional communication and problem-solving skills.

Summary

Course that provides students with comprehensive knowledge of how operating systems function as the interface between hardware, software, and users. Students explore core system components including process management, memory management, file systems, device control, and user interface environments while developing skills in navigating both graphical and command-line tools. Through hands-on learning, students apply operating system tools to manage users, organize files, monitor system performance, and maintain secure and efficient computing environments. The course emphasizes system security practices, troubleshooting methodologies, and preventive maintenance to diagnose and resolve technical issues effectively. Students also examine modern computing environments by exploring virtualization, cloud-based systems, and emerging technologies that support scalability, remote access, and system recovery. Throughout the course, learners develop professional skills through collaborative problem-solving, technical documentation, and real-world support scenarios, culminating in a final project that demonstrates their ability to install, configure, secure, and evaluate an operating system solution.

Notes

The course reflects a significant shift from a narrowly focused, security-heavy curriculum to a more comprehensive and structured information technology course that emphasizes both foundational knowledge and applied skills. The original course concentrated primarily on advanced security concepts such as malware, rootkits, kernel debugging, and protection mechanisms within operating systems environments. In contrast, the revised course expands the scope to include core operating system functions such as process, memory, file, and device management, as well as user interface navigation and system utilities. Additionally, the new version introduces stronger emphasis on practical system management skills, including file organization, user permissions, system monitoring, and troubleshooting using both graphical and command-line tools. The revised course also incorporates modern concepts such as virtualization, cloud-based environments, and scalability, which were not present in the earlier version.

Student Learning Outcomes

1. Career Exploration and Postsecondary Opportunities

Objective: *Students will analyze career pathways in information technology, exploring industry roles, workplace expectations, required education, certifications, professional organizations, and postsecondary opportunities available in Colorado in order to make informed academic and career decisions.*

- 1) Identify employment opportunities in various fields with a focus in the area of information technology
 - a. Recognize the work typically performed, tools and technology used, and nature of work environments
 - b. Identify potential certifications within the careers
 - c. Find membership organizations associated with the careers
 - d. Understand the necessary education associated within the careers
- 2) Define professionalism within the context of information technology
 - a. Criticism and evaluation
 - b. Presentation
 - c. Assessment
- 3) Recognize laws, regulations, and ethics significant to the fields of information technology
- 4) Identify postsecondary opportunities offered in the state of Colorado
 - a. Technical college, community college, 4-year institutions
 - b. Certificates, associates, bachelor, and advanced degree opportunities

2. Foundations of Operating Systems

Objective: *Students will explain the purpose, structure, and core functions of operating systems and distinguish among common operating system types and interfaces.*

- 5) Define the role of an operating system as the interface between hardware, software, and users.
- 6) Describe major operating system functions, including process management, memory management, file management, device management, and user interface support.
- 7) Compare desktop, server, mobile, embedded, and network operating systems to determine appropriate use cases.
- 8) Identify common command-line and graphical user interface tools used to navigate and manage operating systems.
- 9) Summarize how the kernel, shell, drivers, and utilities contribute to system performance and usability.

3. Managing Users, Files, and System Resources

Objective: *Students will apply operating system tools and procedures to organize data, manage users, configure settings, and monitor system resources.*

- 10) Demonstrate how to create, organize, rename, move, and secure files and folders using operating system tools.

- 11) Apply file permissions and user account settings to support responsible access and basic security.
- 12) Use system utilities to monitor CPU, memory, storage, and running processes.
- 13) Interpret system information and logs to explain performance issues or configuration changes.
- 14) Differentiate among local storage, network storage, and cloud-based storage within an operating system environment.

4. Operating System Security and Protection

Objective: *Students will analyze common operating system security risks and implement foundational practices that protect devices, data, and user accounts.*

- 15) Identify common threats to operating systems, including malware, weak passwords, unpatched software, and unauthorized access.
- 16) Explain the purpose of updates, authentication, authorization, encryption, backups, and access controls.
- 17) Apply basic hardening strategies such as updating systems, managing passwords, enabling firewalls, and configuring security settings.
- 18) Analyze user behaviors and system configurations to determine potential security vulnerabilities.
- 19) Evaluate the ethical and legal responsibilities associated with privacy, data protection, and acceptable use.

5. Troubleshooting and System Maintenance

Objective: *Students will use systematic troubleshooting methods to diagnose, resolve, and document operating system problems.*

- 20) Apply a troubleshooting process to investigate startup, performance, file access, and software compatibility issues.
- 21) Use command-line and graphical tools to test, diagnose, and resolve operating system errors.
- 22) Distinguish between hardware, software, network, and user-related causes of system problems.
- 23) Document troubleshooting steps, solutions, and outcomes using clear technical language.
- 24) Recommend preventive maintenance practices that improve reliability and extend system life.

6. Virtualization and Modern Operating Environments

Objective: *Students will evaluate how virtualization and modern operating environments support testing, deployment, and efficient computing.*

- 25) Describe the purpose of virtual machines, containers, and cloud-based operating environments.
- 26) Compare physical and virtual systems in terms of flexibility, efficiency, cost, and security.

- 27) Demonstrate basic setup or use of a virtualized environment for testing or learning tasks.
- 28) Analyze scenarios to determine when virtualization is an effective solution.
- 29) Explain how modern operating environments support remote work, scalability, and system recovery.

7. Work-Based Learning and Professional Practice

Objective: *Students will apply operating systems knowledge in authentic workplace-style tasks while demonstrating professionalism, communication, and documentation skills.*

- 30) Complete help desk, technician, or system support tasks that reflect workplace expectations.
- 31) Interpret technical requests and prioritize tasks based on user needs, urgency, and available information.
- 32) Produce professional documentation such as service tickets, setup checklists, troubleshooting logs, or user guides.
- 33) Collaborate with peers to solve technical problems and communicate recommendations to technical and non-technical audiences.
- 34) Evaluate personal performance, time management, and quality of work using instructor or industry-based criteria.

8. Leadership and Collaboration in Information Technology

Objective: *Students will develop leadership behaviors that support collaboration, ethical decision-making, and shared responsibility in technical environments.*

- 35) Demonstrate active listening, respectful communication, and constructive feedback during technical discussions and team tasks.
- 36) Organize team responsibilities and contribute to project planning, goal setting, and task completion.
- 37) Analyze how leadership, accountability, and ethical behavior influence team success in information technology settings.
- 38) Defend technical decisions using evidence, criteria, and appropriate terminology.
- 39) Reflect on personal leadership strengths and identify strategies for growth in collaborative environments.

9. Final Assessment Project

Objective: *Students will create and present a culminating project that demonstrates their ability to apply, analyze, evaluate, and communicate operating systems knowledge.*

- 40) Design and complete a final project in which an operating system is installed, configured, secured, maintained, or evaluated to meet a defined scenario or user need.
- 41) Apply course concepts to justify system choices, settings, organization methods, security practices, and troubleshooting decisions.
- 42) Analyze evidence from testing, performance checks, or user requirements to improve the final solution.

- 43) Present the project process, technical decisions, and results using professional written, visual, and oral communication.
- 44) Evaluate the completed project against a rubric that measures technical accuracy, problem solving, professionalism, and reflection.

10. Career and Technical Student Organizations

Objective: *Using the CTSO Resource Document, chapter activities will provide some of the best opportunities CTSO members will have to learn by doing. A successful program of work creates a positive learning atmosphere in the classroom. CTSO members learn how to accept responsibility, work as a team, manage a budget, and handle success and failure. Per the CTE Admin Handbook, CTSOs are embedded into curriculum.*

Professional Development

Prepare each member for entry into the workforce and provide a foundation for success in a career. Becoming a professional does not stop with acquiring a skill, but involves an increased awareness of the meaning of good citizenship and the importance of labor and management in the world of work.

Community Service

Promote and improve goodwill and understanding among all segments of the community through services donated by CTSO chapters, and to instill in its members a lifetime commitment to community service.

Business and Industry Connections

Increase awareness of quality job practices and attitudes, and increase the opportunities for employer engagement and eventual employment.

Financial Leadership

Plan and participate in activities to allow all members to carry out the chapter's projects. Educate members on their own personal finance for current and future success.

Public Relations

Make the general public aware of the good work that students in career and technical education are doing to better themselves and their community, state, nation, and world.

Social Activities

To increase cooperation in the school and community through activities that allow CTSO members to get to know each other in something other than a business or classroom setting.

Academic Standards Alignment

Colorado CTE courses are required to integrate academic and essential skills into course content. The following academic standards have been identified as aligned to this course and should be integrated throughout instruction.

[CO CTE Academic Standards Resources](#) (Rubrics, integration strategies, etc)
[CDE Colorado Academic Standards Online](#) (Online standards lookup tool)

Note: Mathematics and Science standards will be added in the future. Visit the [Colorado Academic Standards](#) page for the most current version.

Reading, Writing and Communicating Crosswalk

Course: Operating Systems

Colorado Academic Standards – Reading, Writing, and Communicating

Standard	Prepared Graduate Competency	Description	Course Evidence / Alignment
Standard 1: Oral Expression and Listening	Prepared Graduate 1 – Collaboration	Collaborate effectively as group members or leaders	Students collaborate during labs and problem-solving activities focused on operating system configuration, security analysis, and virtualization.
Standard 1: Oral Expression and Listening	Prepared Graduate 2 – Oral Presentations	Deliver effective oral presentations	Students explain operating system functionality, security threats, protection mechanisms, and troubleshooting strategies in discussions and demonstrations.
Standard 2: Reading for All Purposes	Prepared Graduate 4 – Informational Texts	Read a wide range of informational texts	Students read and interpret operating system documentation, security advisories, configuration guides, and industry

			resources.
Standard 2: Reading for All Purposes	Prepared Graduate 5 – Language in Context	Understand how language functions in different contexts	Students use precise operating system and cybersecurity terminology and adapt explanations for technical and non-technical audiences.
Standard 3: Writing and Composition	Prepared Graduate 7 – Informational Writing	Craft informational/explanatory texts	Students produce written artifacts such as configuration notes, security analyses, lab reports, and explanations of OS-level protection techniques.
Standard 3: Writing and Composition	Prepared Graduate 9 – Writing Process	Demonstrate mastery of the writing process	Students revise and refine technical documentation to ensure clarity, coherence, and professional accuracy.
Standard 4: Research Inquiry and Design	Prepared Graduate 10 – Research and Ethics	Gather and ethically use information	Students research operating system security threats, tools, and standards; evaluate sources; and apply information ethically and responsibly.

Free Resource Links

1. Career Exploration and Postsecondary Opportunities

- CompTIA IT Careers (IT careers, certifications, pathways)
<https://www.comptia.org/content/it-careers-path-roadmap>
- O NET Online (job roles, skills, work environments)
<https://www.onetonline.org>
- CareerOneStop Technology Careers (career exploration, job outlook, education)
<https://www.careeronestop.org/ExploreCareers/explore-careers.aspx>
- Colorado Community Colleges (postsecondary programs, certificates, degrees)
<https://cccs.edu>

2. Foundations of Operating Systems

- How Computers Work Operating Systems (OS basics, components, functions)
<https://www.howtogeek.com/operating-systems-explained>
- Microsoft Learn Windows Fundamentals (OS features, system tools, navigation)
<https://learn.microsoft.com/windows>
- Linux Journey (Linux basics, command line, OS concepts)
<https://linuxjourney.com>
- Computer Hope Operating Systems Guide (OS types, functions, commands)
<https://www.computerhope.com/jargon/o/os.htm>

3. Managing Users Files and System Resources

- Microsoft Support File Management (files, folders, permissions)
<https://support.microsoft.com/windows>
- Windows Task Manager Guide (CPU, memory, process monitoring)
<https://support.microsoft.com/windows-task-manager>
- Apple macOS User Guide (system settings, file organization)
<https://support.apple.com/macOS>
- Google Drive Help (cloud storage, file management, sharing)
<https://support.google.com/drive>

4. Operating System Security and Protection

- Cybersecurity and Infrastructure Security Agency (cyber hygiene, protection practices)
<https://www.cisa.gov>
- Stay Safe Online (security awareness, safe computing practices)
<https://staysafeonline.org>
- Microsoft Security Basics (updates, authentication, encryption)
<https://learn.microsoft.com/security>
- Kaspersky Cybersecurity Fundamentals (threats, protection strategies, malware)
<https://www.kaspersky.com/resource-center>

5. Troubleshooting and System Maintenance

- Microsoft Troubleshooting Windows (diagnostics, system repair)
<https://support.microsoft.com/windows/troubleshoot>
- Lifewire Computer Troubleshooting Guide (diagnosing problems, fixes)
<https://www.lifewire.com/computer-troubleshooting-4165523>
- ITF+ Troubleshooting Steps Guide (troubleshooting methodology, documentation)
<https://www.comptia.org/resources>
- HowToGeek PC Maintenance (preventive maintenance, optimization)
<https://www.howtogeek.com/pc-maintenance>

6. Virtualization and Modern Operating Environments

- VMware Learning Zone (virtual machines, virtualization concepts)
<https://www.vmware.com/learning>
- Oracle VirtualBox Documentation (virtual environment setup, configuration)
<https://www.virtualbox.org/wiki/Documentation>
- Microsoft Azure Fundamentals (cloud computing, virtual systems)
<https://learn.microsoft.com/azure>
- IBM Cloud Computing Basics (cloud environments, scalability, virtualization)
<https://www.ibm.com/cloud/learn>

7. Work Based Learning and Professional Practice

- LinkedIn Learning IT Skills (technical training, professional growth)
<https://www.linkedin.com/learning>
- Indeed Career Guide (workplace skills, communication, professionalism)
<https://www.indeed.com/career-advice>
- Google Career Certificates (IT support, workplace readiness)
<https://grow.google/certificates>

8. Leadership and Collaboration in Information Technology

- MindTools Leadership Resources (teamwork, communication, leadership skills)
<https://www.mindtools.com>
- Harvard Business Review Leadership Articles (decision making, leadership strategies)
<https://hbr.org>
- Toastmasters Communication Resources (public speaking, presentations)
<https://www.toastmasters.org>
- Trello Project Management (team collaboration, task organization)
<https://trello.com>

9. Final Assessment Project

- GitHub (version control, project documentation, collaboration)
<https://github.com>
- Microsoft Learn Labs (hands on projects, system configuration)
<https://learn.microsoft.com/training>

- Codecademy Computer Science Projects (practice, system applications)
<https://www.codecademy.com>
- CodeHS Project Library (IT projects, real world applications)
<https://codehs.com>



Technology Student Association – National Competitive Events

Software Development

Objective: *Students will design, develop, document, test, and present a software application that responds to the annual TSA Software Development challenge.*

- Analyze the annual TSA Software Development challenge and identify the intended users, purpose, and functional requirements.
- Plan the software application using appropriate design tools such as pseudocode, flowcharts, diagrams, or wireframes.
- Develop a functioning software product using programming practices appropriate to the selected platform.
- Create documentation that explains the software purpose, development process, features, testing, and revisions.
- Apply quality assurance practices through testing, debugging, and refinement.
- Prepare a presentation or demonstration that explains the software solution and development decisions.
- Evaluate the final software using TSA criteria for functionality, design quality, documentation, usability, and presentation.

Resources

- Technology Student Association – Software Development competition guidelines
<https://tsaweb.org/competitions-programs/tsa/themes-problems>
- GitHub Docs – Version control and project collaboration
<https://docs.github.com/>
- Visual Studio Code Documentation
<https://code.visualstudio.com/docs>
- VirtualBox – Virtual machine software
<https://www.virtualbox.org/>



Technology Student Association – National Competitive Events

Coding

Objective: Students will demonstrate coding knowledge and develop a software solution to an on-site programming challenge aligned with TSA Coding competition requirements.

- Study coding concepts such as syntax, data structures, control flow, algorithms, functions, and object-oriented programming.
- Prepare for an individual coding-concepts test used to determine semifinalist placement.
- Analyze an on-site coding problem, evaluation criteria, required tools, and programming constraints.
- Design and develop a software solution within the allotted competition time.
- Use an approved programming language, operating system, or API specified for the challenge.
- Test and debug the program to improve accuracy, functionality, and efficiency.
- Present the completed solution for objective testing and judge evaluation.

Resources

- Technology Student Association – Coding competition guidelines
<https://tsaweb.org/competitions-programs/tsa/themes-problems>
- Python Documentation
<https://docs.python.org/3/>
- Linux Documentation Project
<https://tldp.org/>
- Microsoft Learn – Windows command line documentation
<https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/windows-commands>