Alexander College

YOUR ROAD TO UNIVERSITY STARTS HERE!

CURRICULUM GUIDE: OFFICIAL COURSE OUTLINE

<table>
<tr><td>Course Code</td><td>CPSC 295</td><td>Course Title</td><td colspan="4">Introduction to Computer Systems</td></tr>
<tr><td>Credit Value</td><td>3</td><td>Department</td><td colspan="4">Mathematics and Science</td></tr>
<tr><td rowspan="2">No. of weeks</td><td rowspan="2">14</td><td rowspan="2">Hrs. per week</td><td><i>Lecture</i></td><td><i>Tutorial</i></td><td><i>Laboratory</i></td><td><i>Total</i></td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td>Course Description</td><td colspan="6">This course introduces students to topics necessary to understand the digital systems that form the basis of computer systems. Topics include software architecture, operating systems, I/O architectures, the relationships between application software, operating systems, and computing hardware, threads and synchronization, deadlock avoidance, principles and operation of disks and networks.

Preclusion: Students with credit for CPSC 250 may not take CPSC 295 for further credit.</td></tr>
<tr><td>Prerequisite(s)</td><td colspan="6">ENGL 099, MATH 115 or CPSC115, CPSC 111, CPSC 112</td></tr>
<tr><td rowspan="3">Initial Articulation Targets</td><td><i>UBC</i></td><td><i>SFU</i></td><td><i>UVic</i></td><td><i>UNBC</i></td><td colspan="2"><i>TRU</i></td></tr>
<tr><td>CPSC 2nd (3)</td><td>CMPT 295 (3)</td><td>CSC 2XX (1.5)</td><td>CPSC 1XX (3), waive CPSC 231 (4)</td><td colspan="2">COMP 2130 (3)</td></tr>
<tr><td colspan="6">For updated information on the transferability of this course, please consult the BC Transfer Guide, www.bctransferguide.ca</td></tr>
<tr><td>Learning Outcomes</td><td colspan="6">Upon successful completion of this course, the student will be able to:
• Explain the common bit-level representations of numeric values (unsigned, two’s complement, floating point) and the mathematical properties of their arithmetic and bit level operations.
• Explain the components and encoding of instructions and the roles they play in a computer system.
• Write simple programs in C and assembly language and understand the relationships between the two languages.
• Explain how a computer executes a program written in assembly language.
• Explain basic digital systems; be able to implement Boolean functions with gates; be able to explain digital switching and functional units.
• Explain how a CPU is organized and the differences between instructions excecuted in single cycle and in multi cycle (pipelining).
• Explain the different types of memory commonly used in a computer system
• Explain threads and processes and synchronization using semaphores.
• Explain the sources of conflict that can arise when multiple threads of execution share resources and the basics of how to mediate those conflicts.</td></tr>
</table>

<table>
<tr><td></td><td colspan="3">• Use tools that aid program development, including compilers, code analyzers, debuggers, consistency checkers, and profilers to create reliable and efficient programs.
• Analyze the consequences of imperfect system usage, such as poor memory and CPU performance, crashes, and security vulnerabilities.</td></tr>
<tr><td>Content</td><td colspan="3"><b>Core</b> topics – all of the following will be covered:
• Representation of Data
• Representation of Instructions
• RISC-V instruction set
• Basic digital systems
• CPU organization
• Memory organization
• Threads and synchronization

Additional topics may also be covered, at the discretion of the instructor.</td></tr>
<tr><td>Methods of Instruction</td><td colspan="3">Lecture, discussions, quizzes, in-class activities, assignments</td></tr>
<tr><td>Required Textbook(s)</td><td colspan="3">The following textbook(s) is/are required, or approved equivalent(s).

Patterson, David A. and John L. Hennessy. Computer Organization and Design RISC-V Edition: The Hardware Software Interface. Morgan Kaufmann, 2017. ISBN-13: 978-0128122754</td></tr>
<tr><td>Required Equipment and Technology</td><td colspan="3">Students are required to have a computer with internet access.
The following resources are provided by the College:
• Office 365
• Student email</td></tr>
<tr><td>Homework Hours</td><td colspan="3">At minimum, students can expect one hour of homework for every hour of instructional time.</td></tr>
<tr><td rowspan="2">Evaluation</td><td><i>Component</i></td><td colspan="2"><i>% Value</i></td></tr>
<tr><td>Final examination
Midterm examinations
In-Class Activities
Assignments
Quizzes
Project</td><td colspan="2">25 - 35%
20- 30%
10 - 20%
10 - 20%
10 - 20%
15 - 25%</td></tr>
<tr><td>Completion Requirements</td><td colspan="3">The minimum grade to pass this course is D (50%). Unless otherwise stated, a minimum grade of C- (55%) is required for this course to fulfil a prerequisite.</td></tr>
<tr><td>Course Designer(s)</td><td>Radwa Hammad, M.Sc., Department of Math and Science, Alexander College

Kelly Cheung, Ph.D., Department of Math and Science, Alexander College</td><td>Consultant(s), <i>if applicable</i></td><td>John Edgar, M.Sc. School of Computing Science, Simon Fraser University</td></tr>
<tr><td>Dean’s Approval</td><td>Barbara Moon, Ph.D., Dean of Arts and Sciences, Alexander College</td><td>Dean’s Approval Date</td><td>February 17, 2021</td></tr>
</table>

Curriculum Committee Approval Date	February 17, 2021	First Term Offered	
Last Review Date	February 17, 2021	Next Review Date	February 17, 2026
Revision History			