

COURSE NAME: CIS142 Programming Fundamentals

Credit Value: 6
Total Course Hours: 84
Prerequisite Course(s): None
Corequisite Course(s): None

COURSE DESCRIPTION

This course covers the fundamental principles of computer programming, with an emphasis on problem solving strategies using structured programming techniques. The C programming language, which is widely used and forms the syntactical basis for object-oriented languages such as C++, C#, Objective-C, and Java, is used to introduce problem analysis, algorithm design, and program implementation. Students work in a Linux environment.

PLAR INFORMATION

This course is eligible for Prior Learning Assessment and Recognition. Students are advised to discuss options with their program coordinator.

COURSE LEARNING OUTCOMES

Upon completion of this course, the student will have reliably demonstrated the ability to:

- | | |
|--|--|
| 1.0 Prepare programming plans using logical components to solve practical problems. <ul style="list-style-type: none">1.1 Read and Create Flowcharts1.2 Read and Create Pseudocode1.3 Explain the purposes of procedural programming features to inform businesspersons. | 5.0 Interpret and implement properly different types of loops in code <ul style="list-style-type: none">5.1 Increment Operators (++, --, +=, -=)5.2 The "while" loop5.3 The "do while" loop5.4 The "for" loop5.5 Break and Continue statements |
| 2.0 Follow the execution of a coded program to validate its correctness. <ul style="list-style-type: none">2.1 Properly create, place, and interpret comments in code2.2 Read through a program to find both syntax and logical errors | 6.0 Properly structure a program using functions <ul style="list-style-type: none">6.1 Interpret and implement a function's prototype, definition, and a call to a function6.2 Interpret and implement a function that returns no arguments, one argument, or multiple arguments.6.3 Interpret and implement a function that returns a result6.4 Interpret and implement a function that does not return a result |
| 3.0 Identify, declare, initialize and use variables of different data types in code <ul style="list-style-type: none">3.1 Characters3.2 Integers3.3 Floating-point3.4 Strings/Character Arrays (**Not technically a data type in C) | 7.0 Create, initialize, and search one- and two-dimensional arrays |
| 4.0 Interpret and implement properly different types of conditions in code <ul style="list-style-type: none">4.1 Simple "if" statements4.2 Nested "if" statements4.3 Switch/Case Statements4.4 Boolean Algebra using && and Operators | 8.0 Properly interpret and implement pointers in code <ul style="list-style-type: none">8.1 Declare and initialize pointers8.2 Pointers and functions8.3 Passing arrays to functions |
| | 9.0 Properly interpret and implement structures |

in code

- 9.1 Create structures in code
- 9.2 Create an arrays of structures
- 9.3 Passing Structures by value and by reference

10.0 Dynamically allocate memory in code

GENERAL EDUCATION

This is not a General Education course.

PROGRAM OUTCOMES

This course contributes to the following Ministry of Colleges and Universities approved program learning outcomes (PLO):

Computer Programmer Analyst - Advanced Diploma

1. Identify, analyze, design, develop, implement, verify and document the requirements for a computing environment.
2. Diagnose, troubleshoot, document and monitor technical problems using appropriate methodologies and tools.
4. Analyze, develop and maintain robust computing system solutions through validation testing and industry best practices.
5. Communicate and collaborate with team members and stakeholders to ensure effective working relationships.
10. Gather, analyze and define software system specifications based on functional and non-functional requirements.
11. Design, develop, document, implement, maintain and test software systems by using industry standard software development methodologies based on defined specifications and existing technologies/frameworks.
12. Select and apply object-oriented and other design concepts and principles, as well as business requirements, to the software development process.

ESSENTIAL EMPLOYABILITY SKILLS OUTCOMES

This course contributes to the following Ministry of Colleges and Universities approved essential employability skills (EES) outcomes:

1. Communicate clearly, concisely, and correctly in the written, spoken, and visual form that fulfils the purpose and meets the needs of the audience.
3. Execute mathematical operations accurately
4. Apply a systematic approach to solve problems
5. Use a variety of thinking skills to anticipate and solve problems
7. Analyse, evaluate, and apply relevant information from a variety of sources.
10. Manage the use of time and other resources to complete projects.
11. Take responsibility for one's own actions, decisions, and consequences.

EXTERNAL COURSE ACCREDITATIONS AND CONDITIONS

COURSE EVALUATION

Labs and Assignments - 45%

Test/Quizzes - 30%

Final Assessment - 25%

PROGRAM SPECIFIC GRADING

Per College Grading System

GRADING SYSTEM

A+:	90-100%	B+:	77-79%	C+:	65-69%	D:	50-54%	S - Satisfactory
A:	85-89%	B:	73-76%	C:	60-64%	F:	0-49%	I - Incomplete
A-:	80-84%	B-:	70-72%	D+:	55-59%			F- Repeat Course, included in GPA
								FS- Failure Supplemental
								FR- Repeat course, excluded from GPA

*For a complete chart of grades and descriptions, please see the Grading Policy.

LEARNING RESOURCES

Other Resources:

REQUIRED

C Programming for the Absolute Beginner (3rd edition) by Davenport, Keith, and Michael Vine. C, published by Cengage Learning

Resources listed on the course outline support the achievement of learning outcomes, and may be used throughout the course to varying degrees depending on the instructor's teaching methodology and the nature of the resource.

LEARNING ACTIVITIES

In-class lecture, in-class exercises, and hands-on activity

DELIVERY MODE

This course may be delivered, in whole or in part, in a number of modalities, including in class, online, hybrid, in a synchronous or asynchronous manner or a combination thereof, as per accreditation and/or regulatory standards where appropriate.

ACADEMIC POLICIES

Canadore College is committed to the highest standards of academic integrity, and expects students to adhere to these standards as part of the learning process in all environments. The College's Academic Integrity policy seeks to ensure that all students understand their rights and responsibilities in upholding academic integrity and that students receive an accurate and fair assessment of their work. Please review the Academic Integrity policy (A-18) and other academic policies found on our website:
<https://www.canadorecollege.ca/about/policies>.

COLLEGE POLICIES

- Protecting human rights in support of a respectful college community

For college policies please see: <http://www.canadorecollege.ca/about-us/college-policies>.

STUDENT SUCCESS SERVICES - Your Success Matters!

Student Success Services provides student-focused services to facilitate students' success in their studies. Staff provide support by reducing and/or removing educational-related barriers through individualized accommodations and supports to students with disabilities.

Please visit our webpage to learn more: <https://www.canadorecollege.ca/support/student-success-services> or look for our events on social media.

To connect with Student Success Services email studentsuccessnow@canadorecollege.ca or call 705.474.7600 ext 5205.

FIRST PEOPLES' CENTRE:

A culturally safe environment offering CONFIDENTIAL student focused services, drop in or make an appointment to access:

- One on one counselling
- Elder in residence program
- Peer tutoring
- Peer mentorship
- Lunch & learn workshops on study skills, self-care, life skills
- Learning Resource Centre

Drop by our offices at C254 College Drive, E101 Commerce Court or call 705 474 7600 Ext. 5961 College Drive / 5647 Commerce Court.

<https://www.canadorecollege.ca/experience/indigenous-student-experience>

WAIVER OF RESPONSIBILITY

Every attempt is made to ensure the accuracy of this information as of the date of publication. The college reserves the right to modify, change, add, or delete content.

HISTORICAL COURSE OUTLINES

Students use course outlines to support their learning. Students are responsible for retaining course outlines for future use in applications for transfer of credit to other educational institutions.