

Computer Science Department COMSC-110: Introduction to Programming, Spring 2014, Section 8265



COURSE DESCRIPTION:

Introduction to programming emphasizing modular design and development of programs, coding style, documentation, debugging and testing. All control structures and data types of a commonly used language are covered.

Language: C++ **Prerequisite:** MATH-090 or MATH-090SP or MATH-090E or equivalent. **Recommended:** COMSC-100.

INSTRUCTOR:

[Amr Elkady](#) (aelkady@dvc.edu)

Email: Please *use D2L email* for any communications related to class, assignments, or grades.

Office Hours: TBD

OBJECTIVE:

This is an introductory course to programming *using* the C++ programming language **with particular emphases on practical applications, software engineering, programming best practices, problem analysis, and software design practices**. It is intended for those with little programming background, and those with previous experience will learn to be better programmers as well as learn C++-specific constructs and concepts.

By the end of this course, you should be able to:

- Analyze problem specifications;
- Apply the fundamentals of good programming structure and good programming practices;
- Use variables, branching, looping, libraries, and subprograms;
- Design, code, debug, test, and document programs written to solve diverse problems;
- Use data structures such as records, files, and arrays, and to assess their suitability for various applications;
- Use C++ pointers;
- Use recursion

TOPICS COVERED:

- Review of computer and programming terminology
- Data types, constants, variables expressions
- Input/Output
- C++ functions
- Structure charts, flowcharting, algorithms, pseudocode
- Conditions & logical operators, IF statements, and loops
- Data representation and advanced data types
- Structured types, Arrays
- File basics
- Simple recursive functions.
- Dynamic data structures and singly linked lists
- Collections
- Introduction to selected topics in programming and computer science

TEXT & REFERENCES:

Details about textbook, reference books, web resources, and online references are in the “books” section on the class’s **D2L** website.

FEEDBACK:

Opinions and feedback are always welcomed. You can use feedback discussion group on D2L which gives you the option to provide anonymous feedback. You can also talk to me directly before/after class or by email. You may also be asked to provide feedback during class and in D2L surveys, you are encouraged to participate.

GRADING:

Midterm Exam	15%
Assignments & Projects	20%
Quizzes	35%
Participation (<i>not attendance</i>)	15%
Final Exam	15%
Total	100%

Final letter grades are based on the following overall course percentage:

90 – 100	%	A
80 – 89	%	B
70 – 79	%	C
60 – 69	%	D
00 – 59	%	F

MAKE-UP EXAMS:

Normally there will be no make-up exams. Under exceptional circumstances, a make-up exam may be given if a student notifies me beforehand and presents a valid documented reason for missing the exam. Valid excuses include illness severe enough to require a visit to the doctor.

ASSIGNMENTS:

“**Assignment Guidelines**”, “**C++ Programming Style Guidelines**”, and “**Common Mistakes**” documents, available on D2L, details all information regarding assignments. They are supplemental to this Syllabus; students must read them thoroughly and follow precisely. When assignments are graded, you must read instructor comments and avoid repeating your mistakes. Assignment questions should be posted to D2L discussion. See class policy for in-lab work.

ATTENDANCE & PRATICIPATION:

In case you miss a lecture, you are responsible for all assignments, announcements of exams made in class, and all activities done in class. In case you miss a Quiz, your grade will be 0. You may also be dropped for “inactive enrolment”.

See “**Participation**” document on D2L for full details.

WITHDRAWL FROM CLASS:

During first 2 weeks of classes: If you miss any class during either 1st or 2nd week you will be immediately dropped as NS.

You must remember to sign-in before you leave class.

After the first 2 weeks of classes: If you decided to drop you must drop the class yourself. If your name still shows up on the FINAL Class Roster, school policy requires me to assign you an “F” – Sorry! .. Instructor reserves the right to drop students based on them being inactive for 2 or more weeks.

CLASS POLICY:

- Expect to spend minimum of 15 hours per week on this course reading, attending lectures, doing labs & assignments, and writing your own practice code.
- **Code of Silence:** During lecture, you are expected to keep absolute quiet. If you have questions speak up. Other than that, no side conversations allowed. If you have unfinished chat, step outside. However, during labs and other activities it is ok to talk quietly with your classmates.
- All forms of **recording** (*Audio/Video/Images*) are strictly forbidden in class.
- **No side activities** not directly related to lecture allowed. If you want to check the game scores or your email, there are computer labs in the library and you can go there.
- **You are responsible for everything said in class:** that includes verbal instructions, corrections, and things to do. You must pay attention and take notes.
- Lecture supersedes everything: including books, websites, etc. C++ evolves quickly and there are many wrong and obsolete references out there.
- You **must** take your own lecture notes as part of the learning process. If you don’t, you will find this course extremely difficult. Slides are not shared.
- You are required to keep track of and **promptly follow** (*within the same day*) D2L news announcements and email. You must setup D2L SMS notifications and email forwarding.
- Read chapter on topic before class. Read the chapter again & notes after each lecture.
- **Quizzes: (a) Topic quizzes** will be conducted prior the first lecture of a topic. They cover the reading material of the indicated chapter(s). **(b) Pop quizzes** are not announced.
- We try not to spoon feed. You are expected to be proactive about learning: read, do programming challenges in book, review book questions, take notes in class, practice, and ask questions.
- You must upload the in-lab work you did to the D2L folder. Lab work is pass/fail and maybe rejected if

not showing sufficient quality, quantity, and/or authenticity. Rejected lab work will affect participation grade and will count as a missed/non-participating week. Missing the equivalence of 2 or more weeks may result in being dropped.

- We tolerate learning but not sloppiness. As a result, penalties will be gradually enforced.
- Final and midterm exams are both *open book*.

Reference: [Preparing For computer Science Exams](#).

ACADEMIC INTEGRITY:

The general rule on Cheating & Plagiarism is:

Don’t do it!

Except for the work you hand in as individual contributions, you are strongly encourage to collaborate and help each other. If in doubt if collaboration is legitimate: ask!

Assignments: Students may legitimately receive help from colleagues, friends, YouTube, and others. However, the assignment done and turned in must be primarily the result of student’s own efforts. It is improper to receive from or give to another student one’s program in any form.

Exams & Quizzes: In my experience, most students are honest. However, occasionally I have found that students have copied each other’s test answers. During a test, it is cheating to copy another student’s work, to discuss the test, or to use books or notes that were not explicitly allowed. Shield your test from others’ eyes. I don’t distinguish between “source” and “target” - if you let another student copy your work, you are cheating. I hate the role of policeman, but cheating hurts the learning process of both the cheaters and the rest of the class.

If this should happen, all the parties willingly involved will as a minimum penalty receive 0 for this incident and a final grade which is at least one grade lower than they would otherwise have received. This is the minimum penalty. Grades awarded to work later found plagiarized will be withdrawn and policy will be applied.

Please refer to [The DVC Academic Dishonesty Policy](#) and [Student Code of Conduct Policy](#).

DISRUPTIVE BEHAVIOR:

The classroom is a special environment in which students and faculty come together to promote learning and growth. It is essential to this learning environment that respect for the rights of others seeking to learn and respect for the professionalism of the instructor are maintained. Student conduct which disrupts the learning process shall not be tolerated and may lead to disciplinary action and/or removal from class per [DVC policies](#).

STUDENTS WITH DISABILITIES:

Students with the disabilities may receive assistance through the [Disabled Students Services Center](#).

CHANGES TO THE SYLLABUS:

The contents of this syllabus are subject to change with appropriate notice to the students.