

CUNY John Jay College of Criminal Justice
MATH AND COMPUTER SCIENCE

CSCI 271 Introduction to Computer Science
Credits: 3

COURSE SYLLABUS CSCI 271 – section ##

Class Location:	Meeting Days/Time:	
Instructor Name:	Email:	Phone:
Office Hours:	Office Location:	

TEXT & REFERENCE MATERIAL

C++ How to Program (10th Edition)
By Paul Deitel & Harvey Deitel
ISBN-13: 978-0-13-444823-7

C++ Reference: <http://en.cppreference.com/w/cpp>
C++ FAQ: <https://isocpp.org/faq>

CATALOG DESCRIPTION

Computer problem solving and programming in a high level language such as C++ are introduced. Algorithmic problem solving and basic programming techniques are emphasized. Problems are solved using methods such as top-down design and stepwise iterative refinement. Programming topics include basic data types, operators and expressions, control structures, functions, arrays and pointers. Students are introduced to a modern program development environment in the computer lab and are expected to complete several programming projects.

COURSE REQUIREMENTS

Prerequisites: ENG 101, and MAT 105 or the equivalent.

This course satisfies the Flexible Core: Scientific World area of the Gen Ed Program. This course is also a prerequisite for CSCI 272 and the first Core Computer Science course in the Computer Science and Information Security major.

COURSE OBJECTIVES

Students will be able to:

- CO1. Identify the major logical units of a computer, understand the difference between machine, assembly and high-level computer languages and explain the process that source code goes through to become an executable program. (Ch. 1)
- CO2. Demonstrate the ability to design programs to take input from a keyboard and output text to a computer terminal window. (Ch. 2)
- CO3. Understand operator precedence and demonstrate the ability to use operators to perform arithmetic calculations and assign values to variables. (Ch. 2, 4)
- CO4. Demonstrate the ability to build simple conditions, as well as more complex conditions using boolean logical operators, in selection control structures, executing the appropriate code according to the result of the condition evaluation. (Ch. 4, 5)
- CO5. Explain the difference between sequence, selection and repetition control structures and demonstrate the ability to use each to implement algorithms. (Ch. 4, 5)
- CO6. Understand the rationale behind dividing program tasks into separate functions, the operation of the function call stack and the difference between passing arguments by value and by reference, and demonstrate the ability to construct and use functions. (Ch. 6)
- CO7. Demonstrate the ability to declare and use single and multi-dimensional arrays, within a function or passed between functions. (Ch. 7, 8)
- CO8. Explain what a pointer is, how pointers relate to arrays, and demonstrate the use of pointers and pointer operators. (Ch. 8)

Module	Chapter	Topics	Assignments
1	1	Intro to Computers and Programming Languages (CO1) <ul style="list-style-type: none"> • computers: hardware and software (1.2, 1.3) • data hierarchy (1.4) • machine, assembly and high-level languages (1.5 – 1.7) • Object and Software Development Technology (1.8, 1.13) • C++ development environment and debugging (1.9, 1.10) • Operating Systems, Internet (1.11, 1.12) 	HW 1
2	2	C++ Programming Constructs and Concepts: Variables, Assignments (CO2, CO3) <ul style="list-style-type: none"> • basic output (2.2 – 2.3) • variable declaration and basic input (2.4 – 2.5) • arithmetic operators and operator precedence (2.6) • equality and relational operators (AND, OR, XOR, NOT) (2.7) 	HW 2

Module	Chapter	Topics	Assignments
3	4	C++ Programming Constructs and Concepts: Sequential Execution, Loops, Control Structures, Assignments (CO3, CO4, CO5) <ul style="list-style-type: none"> • describing algorithms (4.2 – 4.3) • sequence, selection and repetition control structures (4.4) • if / if ... else statements (4.5 – 4.6) • while loops (4.8) • counter-controlled repetition vs. sentinel-controlled repetition (4.9 – 4.10) • nested control structures (4.11) • assignment operators and increment/decrement operators (4.12 – 4.13) 	HW 3
4		Midterm Exam 1	
5	5	C++ Programming Constructs and Concepts: Loops, Control Structures, Basic Boolean Logic/Operators (CO4, CO5) <ul style="list-style-type: none"> • counter-controlled repetition, for loops (5.2 – 5.6) • do ... while loops (5.8) • multiple selection with switch statements (5.9) • break and continue statements (5.10) • boolean logical operators (AND / OR / NOT) (5.11) 	HW 4 Project 1
6	6	C++ Programming Constructs and Concepts: Functions Part 1 (CO6) <ul style="list-style-type: none"> • defining functions and function prototypes (6.2, 6.4) • C++ standard library libraries, including math (6.3) • argument coercion and promotion rules (6.5) • random number generation (6.7, 6.9) • user-defined enumeration type (6.8) • scope rules (6.10) • function call stack (6.11) • inline functions (6.12) • pass by value vs. pass by reference (6.13) • default arguments (6.14) • unary scope resolution operator (6.15) • function overloading (6.16) • introduction to recursion (6.18 – 6.20) 	HW 5
7		Midterm Exam 2	

Module	Chapter	Topics	Assignments
8	7, 8	C++ Programming Constructs and Concepts: Arrays (CO6, CO7) <ul style="list-style-type: none"> • declaring and using one-dimensional arrays (7.2 – 7.5, 8.5) • searching and sorting arrays (7.7) • multi-dimensional arrays (7.8) 	HW 7 Project 2
9	8	C++ Programming Constructs and Concepts: Pointers (CO6, CO7, CO8) <ul style="list-style-type: none"> • pointer declaration and initialization (8.2) • address and dereference operators (8.3) • using pointers with functions (8.4) • using const with pointers (8.5) • pointer expressions and arithmetic (8.6, 8.7) • pointers and arrays (8.9) 	HW 8
10		Final Exam (Departmental)	

GRADING

Homework	10%
Projects	20%
Midterm Exams	40%
Final Exam	30%

STUDENT INTEGRITY

Statement of the College Policy on Plagiarism

Plagiarism is the presentation of someone else's ideas, words, or artistic, scientific, or technical work as one's own creation. Using the ideas or work of another is permissible only when the original author is identified. Paraphrasing and summarizing, as well as direct quotations require citations to the original source.

Plagiarism may be intentional or unintentional. Lack of dishonest intent does not necessarily absolve a student of responsibility for plagiarism.

It is the student's responsibility to recognize the difference between statements that are common knowledge (which do not require documentation) and restatements of the ideas of others.

Paraphrase, summary, and direct quotation are acceptable forms of restatement, as long as the source is cited.

Students who are unsure how and when to provide documentation are advised to consult with their instructors. The Library has free guides designed to help students with problems of documentation. (*John Jay College of Criminal Justice Undergraduate Bulletin*, <http://www.jjay.cuny.edu/academics/654.php>, see Chapter IV Academic Standards)