# Kishwaukee College Schedule
# CIS 150 - 5H01
# C++ Programming I - Honors

## Tentative Weekly Schedule

Please note that this schedule and the topics covered are likely to change. Changes will be announced in class. If you are not able to attend class, it is your responsibility to find out what was covered. A more detailed schedule is provided on the course website. Assignment descriptions and due dates will also be posted on the course web site.

| Week | Date | Topics |
|---|---|---|
| 1 | 1/18 | Overview of course and introduction to programming (Chapter 1)<br>• **School closed for MLK birthday observance on 1/16/17**<br>• syllabus<br>• C++ compilers, MSDNAA downloads<br>• intro to zybooks.com<br>• writing a simple program, using Visual Studio<br>• program structure, basic input and output, comments, errors<br>• basic programming concepts |
| 2 | 1/23,<br>1/25 | Variables, expressions, and assignment statements (Chapters 1 and 2)<br>• identifiers, variables, and constants<br>• assignment statements and arithmetic expressions<br>• data types in C++<br>• the binary number system<br>• output formatting<br>• **Chapter 1 challenge activities due**<br>• **Chapter 2 challenge activities due**<br>• **In-class lab: Input, output, expressions, calculations** |
| 3 | 1/30, 2/1 | More variables and basics (Chapter 3)<br>• characters and strings<br>• overflow<br>• number types and unsigned numbers<br>• type conversions<br>• math functions<br>• random numbers<br>• debugging<br>• style guidelines<br>• **Chapter 3 challenge activities due**<br>• **Program due: Input, output, calculations**<br>• **Honors project: Initial proposal due** |

| | | |
|---|---|---|
| 4 | 2/6, 2/8 | Selection (Chapter 4)<br>• the Boolean (bool) data type<br>• logical operators<br>• relational operators<br>• using "if" and "if/else" selection statements<br>• using the "switch" selection statement<br>• the conditional (?) operator<br>• **Chapter 4 challenge activities due**<br>• **In-class lab: Selection, calculation, output formatting** |
| 5 | 2/13,<br>2/15 | Repetition (Chapter 5)<br>• using the "while" statement<br>• using the "do/while" statement<br>• using the "for" statement<br>• nested loops<br>• increment and decrement operators<br>• the "break" and "continue" statements<br>• loop counters and sentinel values<br>• accumulators<br>• **Chapter 5 challenge activities due**<br>• **In-class lab: Repetition, input validation**<br>• **Program due: Selection, output formatting**<br>• **Honors project: User/project requirements due** |
| 6 | 2/20,<br>2/22 | Functions (Chapter 6)<br>• breaking a program into simpler, modular pieces<br>• creating and using simple functions<br>• declaring and defining functions<br>• calling functions<br>• passing values to functions<br>• returning values from functions<br>• how functions work<br>• **In-class lab: Functions, input validation**<br>• **Program due: Repetition** |
| 7 | 2/27, 3/1 | Functions continued (Chapter 6)<br>• common errors in functions<br>• passing references to functions<br>• variable scope and lifetime in functions<br>• default parameter values<br>• overloading functions<br>• unit testing for functions<br>• **Chapter 6 challenge activities due**<br>• **Program due: Functions, input validation** |

| | | |
|---|---|---|
| 8 | 3/6, 3/8 | Application of concepts so far and Midterm exam<br>• In-class demonstration of concepts covered so far<br>• **Midterm exam #1**: input, output, variables, calculations, selection, repetition<br>• **Program due: Functions**<br>• **Honors project: Basic design document due** |
| | 3/13,<br>3/15 | **School closed 3/13/17 - 3/19/17 for Spring Break** |
| 9 | 3/20,<br>3/22 | File I/O (Chapter 7), Arrays (Chapter 8)<br>• declaring arrays<br>• initializing arrays<br>• array bounds<br>• accessing array values<br>• processing arrays<br>• **In-class lab: Sequential (text) file input/output** |
| 10 | 3/27,<br>3/29 | Arrays (Chapter 8)<br>• creating and using arrays of strings<br>• passing arrays to functions<br>• **Chapter 8 challenge activities due**<br>• **In-class lab: Creating and using arrays**<br>• **In-class lab: Pointers and arrays**<br>• **Program due: Sequential (text) file I/O**<br>• **Honors project: Detailed design document due** |
| 11 | 4/3, 4/5 | Searching and sorting arrays (notes on course website)<br>String and character operations (Chapter 9)<br>• char data type operations<br>• C++ string access and modification operations<br>• C-style strings (char arrays) and associated operations<br>• **Chapter 9 challenge activities due**<br>• **In-class lab: Sorting**<br>• **Program due: Arrays** |
| 12 | 4/10,<br>4/12 | Pointers and reference variables (chapter 10)<br>• declaring pointer variables<br>• initializing pointer variables<br>• the address-of operator (&)<br>• using pointer variables, de-referencing (*)<br>• dynamic memory allocation<br>• releasing dynamic memory<br>• working with pointers<br>• using reference variables instead of pointers<br>• types of memory: heap vs. stack |

| | | |
|---|---|---|
| | | • memory leaks<br>• functions: passing by value vs. passing by reference<br>• functions: passing pointers<br>• functions: passing reference variables<br>• **Chapter 10 challenge activities due**<br>• **Midterm exam #2**: functions, arrays, sequential (text) file I/O<br>• **School closed on 4/13/17 for faculty development**<br>• **School closed on 4/14/17 for Good Friday** |
| 13 | 4/17,<br>4/19 | Enumerations and structured data (Chapter 11)<br>• defining and accessing structures<br>• passing structures to functions<br>• arrays and pointers to structures<br>• **Chapter 11 challenge activities due**<br>• **Program due: Pointers, arrays, sorting**<br>• **Honors project: Pleliminary implementation due** |
| 14 | 4/24,<br>4/26 | Advanced file operations (notes on course website)<br>• reading and writing binary file data<br>• implementing random access files using C++<br>• **In-class lab: Random access (binary) file input/output** |
| 15 | 5/1, 5/3 | Introduction to classes (Chapter 12)<br>• introduction to objects<br>• introduction to classes<br>• defining class members<br>• defining access: private and public<br>• the difference between a class and a struct<br>• constructors and member initialization<br>• overloading<br>• destructors<br>• accessors<br>• mutators<br>• **Chapter 12 challenge activities due**<br>• **In-class lab: Classes and objects**<br>• **Program due: Random access (binary) file I/O, structs** |
| 16 | 5/8, 5/10 | Exceptions (Chapter 13)<br>• exception basics<br>• using exceptions with functions<br>• multi-file programs<br>• separating header and implementation files for classes<br>• preprocessor directives: include, define<br>• namespaces<br>• **Program due: Classes, objects**<br>• **Honors project: Final implementation due** |

| | | |
|---|---|---|
| | | • **Honors project: Reflection report due** |
| 17 | 5/15 | **Final exam: 10:00 A.M. - 11:50 A.M., Rm. A-1374**<br>comprehensive with emphasis on classes, objects, random access (binary) file I/O |