**CSC 201: C Programming Language**

**Instructor**: Dr. Soha S.Zaghloul,

Building 6, Third Floor, Office 124 (6T124)

**Schedule**: Sun, Tue, and Thu from 11 to 11:50

**Office Hours**: Mon, Tue and Thu from 9 to 11 and 12 to 1

Sun from 12 to 1

**Blog:** www.csc201.wordpress.com

**Course Description:**

CSC 201 provides an introduction to the C programming language for those with **no prior programming experience**. The course aims to teach the syntax and use of major constructs of the C language. The course coversC built-in data types, arithmetic, assignment, relational, and logical operators. In addition, it includes the selection and repetition constructs. Functions, arrays, pointers, character strings, and files are also discussed. After successfully completing this course, the student will be able to write computer programs using C language.

**Course Objectives:**

This course aims at gaining experience in the computer programming using the C language. The course enables the students to analyze and apply the concepts of programming in the C language. Upon completion of the course, the student should be able to:

* Understand introductory computer science terms.
* Understand the program specification.
* Translate the program specification into the C language
* Compile and execute a C program in a programming environment.
* Debug C programs.
* Perform arithmetic, relational and logical operations.
* Manipulate characters and strings.
* Apply various selection constructs.
* Implement repetition controls.
* Use functions.
* Manipulate arrays using pointers.
* Manipulate files

**Textbook:**

Paul Deitel and Harvey Deitel.“C: How to Program”, Prentice Hall, 7th Edition, 2012.

**Grades Distribution:**

Homeworks and Quizzes 10

Labs and Projects 20

Mid 1 15

Mid 2 15

Final 40

**Prospective Course Plan (*for the current semester)***

|  |  |  |  |
| --- | --- | --- | --- |
| **W** | **L** | **Topics** | **Reference in Deitel** |
| 1 | 1 | Course Orientation |  |
|  | 2 | Introduction   * Computer Organization * Compilers * C Standard Library | Section 1.3  Section 1.6  Section 1.9 |
|  | 3 | C Keywords  Writing a sample program  Writing comments in a program  The Memory Concept  Data Types in C  Strings & characters fundamentals  Constants & Variables | Fig. 2.15  Fig. 2.1 to 2.5  Section 9.6  Section 7.5 |
| 2 | 1 | The scanf statement   * Reading formatted input with scanf   The printf statement   * Formatting output with printf * Printing characters and strings * Other conversion specifiers * Printing with field widths and precision * Using flags in the printf statement * Printing literals and escape sequences | Section 9.11  Sections 9.3 to 9.5  Section 9.7 (defer p)  Section 9.8  Section 9.9  Section 9.10 |
|  | 2 | Operators in C language   * Arithmetic operators * Assignment operator * Increment & Decrement operators * Relational operators * Confusing Equality (==) & Assignment operator (=) * Logical operators * Rules of precedence | Fiig. 2.9  Section 3.11  Section 3.12  Fig. 2.12 & 2.13  Section 4.11  Section 4.10  Figs 2.10, 2.11, 2.14, 4.16 |
|  |  | **Quiz #1: All Above** |  |
|  | 3 | Program Development in C   * Algorithms, flowcharts &pseudocode * The if selection statement * The if…else selection statement * Nested if statements | Sections 3.1 to 3.4  Section 3.5  Section 3.6  Section 3.6 |
| 3 | 1 | The selection structure (cnt’d)   * The switch statement * The break statement | Section 4.7  Section 4.9 |
|  |  | **Quiz #2: Selection Structure** |  |
|  | 2 | Loops:  The counter-controlled repetition   * The for statement | Sections 4.3 to 4.6 |
|  | 3 | Loops:  The sentinel-controlled operation   * The while repetition statement * The continue statement | Section 3.7  Section 4.9 |
| 4 | 1 | Loops:  The sentinel-controlled operation (cnt’d)   * The do…while repetition statement | Section 4.8 |
|  | **2** | **The National Day** |  |
|  |  | **Quiz #3: Loops** |  |
|  | 3 | Functions   * What are functions? * Function prototype * Function definition * Function calling (execution) | Section 5.6, Fig. 5.3  Section 5.5  Section 5.7, Fig. 5.3 |
| **5** | **1** | **Eid Al-Adha** | |
|  | **2** |
|  | **3** |
| **6** | **1** |
|  | **2** |
|  | **3** |
| 7 | 1 | Functions (cnt’d)   * Call-by-value * Scope Rules | Section 5.9 |
|  | 2 | Functions (cnt’d)   * C built-in functions * String manipulation functions | Fig. 5.2, 5.6, 8.1, 8.5, 8.12  Fig. 8.17, 8.20, 8.22, 8.36 |
|  |  | **Quiz #4: Functions** |  |
|  | 3 | Arrays   * What is an array? * Declaration (definition) of arrays * Examples | Section 6.2  Section 6.3  Section 6.4 |
| 8 | 1 | Arrays with functions   * Passing arrays to functions * Searching arrays * Sorting arrays | Section 6.5  Section 6.8  Section 6.6 |
|  | 2 | Arrays (cnt’d)   * Multiple-subscripted arrays * The sizeof operator | Section 6.9  Section 7.7 |
|  |  | **Quiz #5: Arrays** |  |
|  | 3 | Revision (if time allows) |  |
| 9 | 1 | **Mid-Term 1 Exams** | |
|  | 2 |
|  | 3 |
| 10 | 1 |
|  | 2 |
|  | 3 |
| 11 | 1 |
|  | 2 |
|  | 3 | Pointers:   * Definition & initialization * Pointer operators * Pointer expressions and arithmetic * The pointer conversion specifier | Section 7.2  Section 7.3  Section 7.8  Section 9.7 |
| 12 | 1 | Pointers with functions   * Call-by-reference * Example: Bubble sort using call-by-reference | Sections 5.9 and 7.4  Section 7.6 |
|  | 2 | Relationship between pointers and arrays  Arrays of pointers | Section 7.9  Section 7.10 |
|  | 3 | Revision (if time allows) |  |
|  |  | **Quiz #6: Pointers** |  |
| 13 | 1 | **Mid-Term 2 Exams** | |
|  | 2 |
|  | 3 |
| 14 | 1 |
|  | 2 |
|  | 3 |
| 15 | 1 |
|  | 2 |
|  | 3 | Files:   * Data Hierarchy * Files and streams | Section 11.2  Section 11.3 |
| 16 | 1 | Files (cont’d):   * Creating a sequential-access file * Reading data from a sequential-access file | Section 11.4  Section 11.5 |
|  | 2 | Files (cont’d):   * Random-Access files * Creating a random-access file * Writing data randomly into a random-access file * Reading data from a random-access file | Section 11.6  Section 11.7  Section 11.8  Section 11.9 |
|  | 3 | Revision (if time allows) |  |
| 17 | 1 | **Practical Exams Week** | |
|  | 2 |
|  | 3 |
| 18 | 1 | **Final Exams** | |
|  | 2 |
|  | 3 |
| 19 | 1 |
|  | 2 |
|  | 3 |
| 20 | 1 |
|  | 2 |
|  | 3 |

**Important Policies:**

**1)** Cheating is prohibited. If the student caught cheating, then the grade on that assignment for the person providing answers and the one taking the answers will be **0%.**

**2)** Late assignments will not be accepted.