Department of Computer science

College of Science

University of Salahaddin

Subject: Object Oriented Programming

Course Book – For 2nd Year Second Semester

Lecturer's name: Safeen Hasan Rasool

Academic Year: 2022/2023

Course Book

1. Course name	**Object Oriented Programming**
2. Lecturer in charge	**Safeen H. Rasool**
3. Department/ College	**Collage of Science – Computer Department**
4. Contact	**e-mail: saven.rasool@su.edu.krd**
5. Time (in hours) per week	**Theory: 2** **Practical: 2**
6. Office hours	
7. Course code	**CSIT201**
8. Teacher's academic profile	Teacher at Salahaddin university, collage of Science/ computer department since 2011, responsible of too many academic subjects, I obtained BSc in collage of education/computer department in2004 and MSc in same department since 2011. https://academics.su.edu.krd/saven.rasool
9. Keywords	**C++ Programming, Object Oriented Programming, Class**

10. Course overview:

This course provides in-depth coverage of object-oriented programming principles and techniques using C++. Topics include classes, overloading, data abstraction, information hiding, encapsulation, inheritance, polymorphism, file processing, templates, exceptions, container classes, and low-level language features. The course briefly covers the mapping of UML design to C++ implementation and object-oriented considerations for software design and reuse.

11. Course objective:

- Perform object oriented programming to develop solutions to problems demonstrating usage of control structures, modularity, I/O. and other standard language constructs.

- Demonstrate adeptness of object oriented programming in developing solutions to problems demonstrating usage of data abstraction, encapsulation, and inheritance.

- Demonstrate ability to implement one or more patterns involving realization of an abstract interface and utilization of polymorphism in the solution of problems which can take advantage of dynamic dispatching.

- Learn syntax, features of, and how to utilize the Standard Template Library. Learn other features of the C++ language including templates, exceptions, forms of casting, conversions, covering all features of the language.

12. Student's obligation

There will be two main theoretical and practical exams plus a number of quizzes in the lab. The quizzes will be calculated as one exam mark, after studding most Programming and OOP topic skills the student will be required to create an objective OOP project.

13. Forms of teaching

The lectures will be created by Power point presentation viewed to students through data projector and the explanation will be done on the white board, a computer labs. Also required with visual studio package.

14. Assessment scheme

Student marks will be as follow
- Course 1 Theoretical exam will be out of 20
- Course 2 Theoretical exam will be out of 20
- Course 1 Practical exam will be out of 30
- Course 2 Practical exam will be out of 30
- The Lab quizzes calculated out of 10 and each 3 quizzes consider as one exam.

The total over 50 marks then calculated as:

(Average of theoretical course 1 and 2) + (Average of practical course 1, practical course 2, quizzes and project)

The final exam will takes 50% of the marks and it is only theoretical.

15. Student learning outcome:

After completion this course, the student will be able to understand better the object-oriented programming features and will be able to write and test programs that make appropriate use of object-oriented facilities common to many object-oriented languages.

16. Course Reading List and References:

Object-Oriented Programming in C++, Fourth Edition, Robert Lafore, 2002 by Sams Publishing
Available in https://fac.ksu.edu.sa/sites/default/files/ObjectOrientedProgramminginC4thEdition.pdf

A Complete Guide to Programming in C++, Ulla Kirch-Prinz Peter Prinz, JONES AND BARTLETT PUBLISHERS
Available in http://www.lmpt.univ-tours.fr/~volkov/C++.pdf

Week(s)	Basic Tutorial Subject to be covered	Lab
17. The Topics:		**Lecturer's name**
1	**General Review of Objects**	Implementing Examples of class
2-4	**Arrays and String Arrays Fundamentals:** Arrays as Class Member Data, Arrays of object, String, The standard C++ String Class.	Example: Using arrays inside Class, Using dynamic arrays (1 D and 2 D)
5-7	**Operator Overloading:** Overloading unary operations. Overloading binary operators, data conversion, pitfalls of operators overloading and conversion keywords. Explicit and Mutable.	Example: Overloading some C++ operators such as increment (i++) and decrement (i--)
8	**Inheritance: Single Inheritance** (Base Classes and Derived Classes, protected Members, Types of Inheritance, Constructors and destructors functions under inheritance),	Implementing the public inheritance of class
9-10	**Multiple Inheritances** (Multiple inheritances with common base class, Constructors and destructors functions under inheritance, Virtual Base Classes).	Example: Converting classes to a template to accept attributes for different data types (string, int, float)
11-12	**Polymorphism**	Virtual function, function overloading and overriding
13	**Generic Function and Generic Class**	<template>
14	**Templates:** Function Templates (Definition, Instantiation), Class Templates (Definition, Instantiation), virtual function, file stream	Cleaning and Repairing Audio, Using Audition's Built-in Effects, removing music, removing vocabs, changing the frequency of the sound
15	**Final Exam Exam**	

18. Practical Topics (If there is any)

Explained with Theoretical part

19. Examinations:

Write a C++ program to overload + to add two complex number like (5 + 2i and 7 + 6i) and – to subtract two complex number.

Wat is OOP and way we use it?

20. Extra notes:

21. Peer review