Course Syllabi

Course Title and Code	CS182 Computer Programming III

- **Course Identification and General Information:**

Department	Computer Science	Course Level	4
Contact Hours	4 (2+2)	Credit Hours	3 (2+1)
Web Address			

- **Course Instructor's or Coordinator's Name:** Dr. Walid Karamti

- **Textbook Title, Author, and Year:**

- C ++ How to Program, 8/E edition, Deitel & Deitel. ISBN-978-0-13-266236-9, 2012

- **Other Supplemental Materials:**

- Programming with C++, Schaum's Outlines (J. Hubbard)
- C++ Program Design, Cohoon and Davidson

- **Specific Course Information:**

o **Catalog Description**:
Introduction to Object-Oriented Programming, Program Control Structures, Function, Arrays, Pointers and Strings, Classes Part I: Classes and data abstraction, Classes Part II: Operator Overloading, Inheritance, Virtual Functions and Polymorphism.
o **Pre-requisites:** CS 181
o **Co-Requisites:** NIL
o **Required, Elective, or Selected Elective:** Required

- **Specific Goals for the Course:** Summary of the main learning outcomes for enrolled students.

- Students will enhance their basic programming knowledge.
- Students will understand the Object oriented programming domain.
- Students will get the knowledge how to think globally and act locally through using objects.
- Students will get part of the knowledge of secure programming.
- Use sub-classing to design simple class hierarchies that allow code to be reused for distinct subclasses.
- Student will Compare and contrast (1) the procedural approach and (2) the object-oriented approach.

- Students will discover the different features of Object Oriented programming: Operator Overloading, Inheritance, Virtual Functions and Polymorphism, Templates.

➢ Program Outcomes Addressed by the Course:

This course provides the following outcomes with the following relationship:

Computer Science Program Outcome	Relationship to Course
a) An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline.	Not Related
b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.	High
c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.	High
d) An ability to function effectively on teams to accomplish a common goal.	Not Related
e) An understanding of professional, ethical, legal, security and social issues and responsibilities.	Not Related
f) An ability to communicate effectively with a range of audiences.	Not Related
g) An ability to analyze the local and global impact of computing on individuals, organizations, and society.	Not Related
h) Recognition of the need for and an ability to engage in continuing professional development.	Not Related
i) An ability to use current techniques, skills, and tools necessary for computing practice.	Medium
j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.	Not Related
k) An ability to apply design and development principles in the construction of software systems of varying complexity.	Not Related

➢ Brief List of Topics to be covered:

- C++ Basics,
- Classes
- Objects lifetimes
- Inheritance
- Polymorphism
- Operator overloading
- The standard Template

➢ Outcome Assessment:

1. Direct Assessment

☒ Midterm Written Exam I
☒ Midterm Written Exam II
☒ Final Written Exam
☒ Quizzes
☒ Homework
☒ Integrative Projects
☐ Students' Portfolios
☐ Case Study
☐ Oral Exams
☒ Written Reports
☒ Participation in Lecture
☒ Illustrative Presentations
☒ Use of Computer Facilities by Students
☐ Reading of References Related to Course Topics
☒ Team Work
☒ Practice in the Lab

2. Indirect Assessment

☐ Pre-Course Questionnaire
☒ Post-Course Questionnaire
☐ Group Discussions
☐ Students' Interviews

➢ Course Outline:

Contact Hours	Topics
4	C++ Basics,
4	Classes
2	Objects lifetimes
4	Inheritance
4	Polymorphism
4	Operator overloading
2	The standard Template
Week 8	Midterm exam
30	Lab
60	**Total Contact Hours**

Notes:

- A 3 Credit Theory-Course should utilize **45 Contact Hours** in 15 Weeks.
- A 2 Credit Practical-Course should utilize **30 Contact Hours** in 15 Weeks.
- A 4 Credit Course (3 Theory and 1 Practical) should utilize **75 Contact Hours** in 15 Weeks.
- Each theory-credited **Contact Hour** is equivalent to **50 lecture minutes and 10 minutes of rest**.
- Each practical-credit **Contact Hour** is equivalent to **100 lab minutes and 20 minutes of rest**.