**College of Engineering and Architecture**

**COURSE SYLLABUS**

**CSCI136: Computer Science II**

**Department**: Electrical Engineering and Computer Science

**Course Number:** CSCI136

**Course Title:** Computer Science II **Total Credits:** 3

**1.0 General Course Information**

**1.1 Current Catalog Description**

Course exposes students to the software development life cycle with a focus on the concepts and use of the object-oriented paradigm in problem analysis, solution design, software development and implementation. This course is designed to enhance the student’s ability to engineer software that is efficient, maintainable and cost efficient over its entire life cycle. Data abstraction is discussed in depth and students gain experience in the use of classes, object and member functions. Students gain an understanding of the development of reusable abstract data types. Software reuse is emphasized and object-oriented concepts are used throughout the course. O-notation and the complexity of algorithms are discussed at relevant points in the course.

**1.2 Prerequisites**

CSCI100 – Introduction to Systems and Computer Science, CSCI135 – Computer Science I with a final grade of “C” or better

**1.3 Course Staff**

**Instructor**: Harry N. Keeling, Ph.D.

**Office:** 1016 LK Downing Hall

**Phone**: (202) 806-4830

**Email:** [hkeeling@howard.edu](mailto:hkeeling@howard.edu)

**Office Hours:** Tues and Thurs. 11:30 – 2:00 PM

Wed: 4:00 – 5:00 PM

Any other time - by appointment

**Course URL**: http://hkeeling.net

**Grading**

Assignments 45%

Attendance/Quizzes 15%

Exams (3) 40%

**2.0 Learning Resources**

**2.1 Required Text**

Howard University CSCI 136 Fall 2016: Programming in C++ and Data Structures with ZzyBook

**2.2 Department Resources**

Laboratories: (obtain account logins from Mr. Guy Langari)

**3.0 Aims, Objectives, Program Outcomes**

**3.1 Course Aims**

Course exposes students to the software development life cycle with a focus on the concepts and use of the object-oriented paradigm in problem analysis, solution design, software development and implementation.  This course is designed to enhance the student’s ability to engineer software that is efficient, maintainable and cost efficient over its entire life cycle. Data abstraction is discussed in depth and students gain experience in the use of classes, object and member functions.  Students gain an understanding of the development of reusable abstract data types.  Software reuse is emphasized and object-oriented concepts are used throughout the course. O-notation and the complexity of algorithms are discussed at relevant points in the course. Prereq: Computer Science I (with a grade of ‘C’ or better).

**3.2 Course Outcomes**

By the end of this course, you should be able to perform the appropriate tasks that illustrate the following course outcomes:

1. Understanding basic software engineering principles

a. Identify all steps involved and activities within each step of the software life cycle

1. Understand abstract data types

a. Identify various abstract data types

b. Understand and identify data at application, logical, and implementation levels

1. Understand sorted and unsorted lists, stacks, queues, and linked lists

a. Implement each using C++ programming using arrays and/or dynamic memory

1. Understand recursions
   1. Identify output values for recursive algorithms

**3.3 Program Outcomes Used**

## 

|  |  |
| --- | --- |
| A. - An ability to apply knowledge of computing and mathematics appropriate to discipline  B. - An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.  C. - An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.  D. - An ability to function effectively on teams to accomplish a common goal.  E. - An understanding of professional, ethical, legal, security and social issues and responsibilities.  F. - An ability to communicate effectively with a range of audiences.  G. - An ability to analyze the local and global impact of computing on individuals, organizations, and society.  H. - Recognition of the need for and an ability to engage in continuing professional development.  I. - An ability to use current techniques, skills, and tools necessary for computing practice.  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.  K. - An ability to apply design and development principles in the construction of software systems of varying complexity. | 3a, 4a  1a, 2a, 2b, 3a, 4a  1a, 2a, 2b, 3a, 4a |

**4.1 Learning**

**4.1 Major Topics Covered in the Course**

|  |  |  |  |
| --- | --- | --- | --- |
| **CONCEPT** | **FAMILIARITY** | **COMPETENCE** | **MASTERY** |
| Abstract data types |  | √ |  |
| Encapsulation & data hiding |  | √ |  |
| Polymorphism | √ |  |  |
| Object-oriented program design |  | √ |  |
| Dynamic binding | √ |  |  |
| Basic data types |  | √ |  |
| Internal data representation | √ |  |  |
| Enumerated data types |  | √ |  |
| Pointers |  | √ |  |
| Dynamic storage |  | √ |  |
| Structs |  | √ |  |
| Searching & sorting with arrays and records |  | √ |  |
| Base classes |  | √ |  |
| Derived classes |  | √ |  |
| Constructors |  | √ |  |
| Destructors |  | √ |  |
| Analysis of algorithm | √ |  |  |
| Sorting and searching | √ |  |  |
| Stacks and queue |  | √ |  |
| Simulation | √ |  |  |
| Abstract operators |  | √ |  |
| Overloaded functions |  | √ |  |
| Overloaded operators |  | √ |  |
| Assignment and initialization |  | √ |  |
| Safe arrays | √ |  |  |
| String class |  | √ |  |
| Linked lists |  | √ |  |
| Recursion | √ |  |  |
| Trees | √ |  |  |
| Multiple inheritance | √ |  |  |
| Templates | √ |  |  |

**5.0 Assessment**

**5.1 Course Grading**

**Grading Policy:** 90 – 100 …………………………………A  
80 – 89.9 …………………………....……B  
70 – 79.9 …………………………………C  
60 – 69.9 ….………………………...……D  
Below 60 .………………………..….……F

1. **Howard University Statement on ADA Policies and Procedures**

Howard University is committed to providing an educational environment that is accessible to all students. In accordance with this policy, students in need of accommodations due to a disability should contact the Dean of Student Services for verification and determination of reasonable accommodations as soon as possible. Note: Accommodations are not retroactive. The Office of Student Services is located in Suite 725 of the Howard Center and may be reached at (202) 238-2420.

1. **Howard University Statement on Interpersonal Violence**  
   Howard University takes sexual assault, dating violence, domestic violence, stalking and sexual harassment seriously. If a student reveals that he or she needs assistance with any of these issues, all Responsible Employees, which includes faculty, are required to share this information with the University Title IX Office (806-2550) or a student can be referred for confidential services to the Interpersonal Violence Prevention Program (IVPP) (238-2382) or University Counseling Services (806-6870). For more information about these services, please go to [www.CampusSafetyFirst.howard.edu](http://www.CampusSafetyFirst.howard.edu).

**Syllabus Addendum for all CEA Courses**

The following understandings, expectations, and requirements shall apply to all classes that are offered by the *College of Engineering and Architecture*, effective fall 2016. These are the expectations that the students can have from their professors, and the expectations that the professors will have from each student taking her/his class. These are intended promote the success of our students here at Howard, and after graduation.

***Students expect that their professors will:***

1. Care about the success of each student, and promote mutual respect.
2. Come to every class, and on time.
3. Keep abreast of the technical field she/he is teaching.
4. Explain how the subject being taught is broadly connected to the field and possibly to other courses.
5. Keep abreast of, and adapt to, evolving teaching approaches.
6. Have office hours for each class, and be present at these times.
7. Coordinate any travel with his/her Department Chairs.

*Students who feel that a professor fails in the above expectations may confidentially express his concerns in the Comments box in the Office of Student Services in the L. K. Downing Building, Room 1114, or email Dr. Rhoulac Smith, Director of Student Services, at* [*trhoulac@howard.edu*](mailto:trhoulac@howard.edu)*. Your communication must be respectful, professional, and truthful.*

***Professors expect that their Students will:***

*(Failing to comply will result in appropriate penalties. In certain cases these penalties are expressly defined below.)*

1. Take a professional approach to all class activities and interactions. Show that your take the class seriously and come to class prepared.
2. ***Absenteeism***: Come to every class.
   * 2% penalty from total class grade for every unexcused absence
3. ***Lateness***: Not come to class after it begins.
   * 1% penalty from total class grade for every instance of infraction
4. ***Leaving Early***: Not leave class before it ends.
   * 1% penalty from total class grade for every instance of infraction
5. ***Disruption***: Avoid entering and leaving the classroom during instruction.
6. ***Eating***: No eating in class.
   * 1% penalty from total class grade for every instance of infraction.
7. ***Electronics***: No use of laptop, cell phone, iPad, headphones or other electronic equipment that are not explicitly requested to be used by the Professor.
   * 1% penalty from total class grade for every unexcused absence
8. ***Cheating***: Any form of cheating, including plagiarism, in an exam or assignment shall automatically result in a zero-grade for all involved, for that exam or assignment.
9. ***Communication***: Communication, oral or written, with the Professor, including email, in all matters concerning the course, shall be done professionally; that is, as it would be done with a potential employer. (e.g., respectful, include your full name, clearly articulate the objective of the communication concisely)