|  |  |
| --- | --- |
| UNITED INTERNATIONAL UNIVERSITY Department of Computer Science and Engineering (CSE) Course Syllabus | |
|  | |
| **Course Title** | Object Oriented Programming |
| **Course Code** | CSE 1115 |
| **Trimester and Year** | Summer 2024 |
| **Pre-requisites** | CSE 1111 Structured Programming Language |
| **Credit Hours** | 3.0 |
| **Section** | E,T |
| **Class Hours** | Sat,Tue 11:11 am- 12:30 pm (E)  Sat,Tue 1:51pm – 3:10 pm (T) |
| **Instructor’s Name** | Md. Mohaiminiul Islam |
| **Email** | mohaiminul@cse.uiu.ac.bd |
| **Office** | 636(C) |
| **Counselling Hours** |  |
| **Text Book** | Java The Complete Reference, Herbert Schildt |
| **Reference** | 1. Java: How to Program, 9th Edition (Deitel) 2. Java Programming By ANM Bazlur Rahman  3. <https://codingbat.com/java> |
| **Course Contents (approved by UGC)** | Object oriented fundamentals, Java Application, Java applets, Methods, Arrays, String & characters, Graphics & java2D, Basic graphical user interface components, Multithreading, Multimedia, Files & streams, JDBC, Servlets, RMI, Networking, Java beans. |
| **Course**  **Outcomes (COs)**   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | CO | Statement | Bloom’s  Domain | Program Outcome | Knowledge Profile | Complex Problem | Engineering Activities | | CO1 | Understand the fundamental concepts and features of Object-Oriented Programming and use these to write programs for solving computational problems. | C | A  Engineering Knowledge | K3 – Engineering fundamentals | P1 – Depth of Knowledge | - | | CO2 | Understand the core concepts of GUI programming, File IO, Collections framework and use these to solve programming problems. | C | B  Problem Analysis | - | | |
| **Teaching Methods** | Lecture, Case Studies, Project Developments. |
| **CO with Assessment Methods** | |  |  |  | | --- | --- | --- | | **CO** | **Assessment Method** | **(%)** | | - | Attendance | 5 | | - | Assignments | 5 | | - | Evaluations (Best n-1 out of n) | 20 | | CO1 | Midterm exam | 30 | | CO1, CO2 | Final exam | 40 | |
| **Mapping of COs and Program outcomes**   |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **COs** | **Program Outcomes (POs)** | | | | | | | | | | | | | **PO**  **a** | **PO**  **b** | **PO**  **c** | **PO**  **d** | **PO**  **e** | **PO**  **f** | **PO**  **g** | **PO**  **h** | **PO**  **i** | **PO**  **j** | **PO**  **k** | **PO**  **l** | | **CO1** | X |  |  |  |  |  |  |  |  |  |  |  | | **CO2** |  | X |  |  |  |  |  |  |  |  |  |  | | |

**Lecture Outline**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Topics/Assignments** | **COs** | **Reading Material** | **Lecture Outcomes/Activities** |
| 1. 1 | Introduction, Review of Programming, programming language, Motivation to use OOP | - | Slide | 1. What Programming is? 2. Describe different types of programming. 3. Differentiate between Programming and Programming Language. |
| 1. 2 | Java basics (Why Java, Application Class, Main method, identifier, data type, operator), From C to Java | CO1 |  | 1. What is Application class 2. Describe rules of java identifier. 3. Develop basic Hello World program. |
| 1. 3 | Java Basics (control statement, array),  Some Concepts: Scope of variable, ref variable, pass by value/reference, garbage collection | CO1 | Ch 3, 4, 5, 6 | 1. Describe data type, operators, control statement. 2. Define what array is and why we use array. 3. Develop simple program using different types of data, operator and control statement. 4. Differentiate between normal and reference variable. 5. Explain Scope of a variable. 6. Describe the effect of pass-by-value and pass-by-reference |
| 1. 4 | Class and Object (Constructor, Initialization block, this keyword, default value, member of class, create object and access member) | CO1 | Ch 6, 7 | 1. Describe what class and object are. 2. Describe who the members of a class are. 3. Able to create class and object and access members. |
| 1. 5 | OOP Feature: Encapsulation (getter/setter), Method overloading (constructor overloading) | CO1 | Ch 7 | 1. Explain what encapsulation and overloading are and where to use these features. 2. Describe importance of encapsulation and overloading. 3. Able to develop code using encapsulation and overloading. |
| 1. 6 | **Assessment (CT1).**  Package, access modifier. | CO1 | Ch 7, 9 | 1. Describe what is accessible from a specific point in regards to access modifier & package. 2. Describe how to use package and what the benefit of library is. |
| 1. 7 | OOP Feature: Inheritance, this and super keyword, Object Class. | CO1 | Ch 8 | 1. Explain what inheritance is. 2. Describe what get inherited to child class and what can’t be inherited. 3. Get familiar with Object class and some of its method. |
| 1. 9 | OOP Feature: Method Overriding, override equals() and toString() method. | CO1 | Ch 8 | 1. Explain what method overriding is and where to use this feature. 2. Describe importance of method overriding 3. Able to develop code using overriding. |
| 1. 10 | Static & Final keyword, SubClass Polymorphism, Benefit of Polymorphism | CO1 | Ch 7 | 1. Describe what is static and final variable and method. 2. Explain the benefits |
| 1. 11 | **Assessment (CT2).** Abstraction, Abstract Class, abstract method | CO1 | Ch 8 | 1. Explain what abstraction is & how to achieve abstraction. |
| 1. 12 | Review |  |  |  |
|  | Review |  |  |  |
|  | **MIDTERM EXAM** |  |  |  |
| 1. 13 | Interface- variables, methods, abstract class vs. interface | CO1 | Ch 8 | 1. Explain what interface is & how to declare an interface. 2. How can we use interface to achieve inheritance relationship |
| 1. 14 | Exception – try/catch/finally, nested try/catch, throw vs. throws, method stack | CO1 | Ch 10 | 1. Explain what Exception is. 2. Explain how to handle exception using try/catch block. 3. Explain how to throw an exception. |
| 1. 15 | Checked/unchecked exception. User Defined Exception | CO1 | Ch 10 | 1. Differentiate between checked and unchecked exception. 2. Can create and use user defined exception. |
| 1. 16 | **Assessment (CT3).** Nested Class- anonymous class, inner class, accessing variable and method of nested class. | CO1 | Ch 7 | 1. Explain and Apply Nested Classes: Local, Inner and Anonymous class concept. |
| 1. 17 | GUI Basic – Components, Container, Layout | CO2 | Ch 31-33 | 1. Explain different components of GUI. 2. Create GUI application using different Layout and components. |
| 1. 18 | GUI Event Handling- source, listener, event object. Steps to handle event. Handle multiple events | CO2 | Ch 24-26 | 1. Explain and apply the event handling process. 2. Develop GUI application involving multiple event handling. |
| 1. 19 | IO- Streams, Buffering, File read | CO2 | Ch 20 | 1. Explain the IO model, buffering. 2. Able to develop application involve reading from file. |
| 1. 20 | **Assessment (CT4).**  File write | CO2 | Ch 20 | 1. Able to develop application involve writing to file. |
| 1. 21 | Collections- framework, list, ArrayList | CO2 | Ch 18 | 1. Explain the components of Collection framework. 2. Able to use the already defined Collection classes. 3. Able to create ArrayList of objects. |
| 1. 22 | Comparable, Comparator, ArrayList sorting | CO2 | Ch 18 | 1. Able to use Comparable, Comparator to compare the items in a Collection. 2. Able to sort an ArrayList of objects. |
| 1. 23 | Set – HashSet, Map – HashMap | CO2 | Ch 18 | 1. Able to create HashSet and HashMap of objects |
| 1. 24 | Review |  |  |  |

\*\* Class Assessment schedules can be changed later with convenience of everyone

**Appendix 1: Assessment Methods**

|  |  |
| --- | --- |
| **Assessment Types** | **Marks** |
| Attendance | 5% |
| Assignments | 5% |
| Class Tests | 20% |
| Mid Term | 30% |
| Final Exam | 40% |

**Appendix 2: Grading Policy**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Letter Grade** | **Marks %** | **Grade Point** | **Letter Grade** | **Marks%** | **Grade Point** |
| A (Plain) | 90-100 | 4.00 | C+ (Plus) | 70-73 | 2.33 |
| A- (Minus) | 86-89 | 3.67 | C (Plain) | 66-69 | 2.00 |
| B+ (Plus) | 82-85 | 3.33 | C- (Minus) | 62-65 | 1.67 |
| B (Plain) | 78-81 | 3.00 | D+ (Plus) | 58-61 | 1.33 |
| B- (Minus) | 74-77 | 2.67 | D (Plain) | 55-57 | 1.00 |
|  |  |  | F (Fail) | <55 | 0.00 |

**Appendix-3: Program outcomes**

|  |  |
| --- | --- |
| **POs** | **Program Outcomes** |
| **PO1** | An ability to apply knowledge of mathematics, science, and engineering |
| **PO2** | An ability to identify, formulate, and solve engineering problems |
| **PO3** | An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability |
| **PO4** | An ability to design and conduct experiments, as well as to analyze and interpret data |
| **PO5** | An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice |
| **PO6** | The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context |
| **PO7** | A knowledge of contemporary issues |
| **PO8** | An understanding of professional and ethical responsibility |
| **PO9** | An ability to function on multidisciplinary teams |
| **PO10** | An ability to communicate effectively |
| **PO11** | Project Management and Finance |
| **PO12** | A recognition of the need for, and an ability to engage in life-long learning |