মেট্রোপলিটন ইউনিভার্সিটি
Metropolitan University

Department of Computer Science and Engineering

Course Code: CSE222	Credits: 1.5
Course Name: Object Oriented Programming Lab	Faculty: FRS

Lab 02 – Basics of Array and Inheritance

Task 1: Find the maximum and minimum elements in an array.

Sample Input: [5, 3, 9, 2, 1, 7]

Output: Maximum: 9

Minimum: 1

Task 2: Reverse the elements of an array. Store it in a new one and print the new Array. The size of the array is not specified, means it could be 5 or 7 or any integer value.

Sample Input: [1, 2, 3, 4, 5]

Output: [5, 4, 3, 2, 1]

Sample Input: [5, 3, 9, 2, 1, 7]

Output: [7, 1, 2, 9, 3, 5]

Task 3: Remove duplicates from a sorted array.

Sample Input: [1, 1, 2, 2, 3, 4, 5, 5]

Output: [1, 2, 3, 4, 5]

Task 4: Given an array containing integers from 1 to n with one element missing, find the missing number.

Sample Input: [1, 2, 4, 5, 6]

Output: Missing Number: 3

Task 5: Given an array of integers and a target sum, find two numbers in the array that add up to the target sum.

Sample Input: [2, 7, 11, 15], 9

Output: 2, 7

Task 6: Given an array of integers, move all the zeroes to the end of the array while maintaining the relative order of non-zero elements.

Sample Input: [0, 1, 0, 3, 12]

Output: [1, 3, 12, 0, 0]

Task 7: Complete the **Animal**, **Dog**, **Cat** classes. So that it matches the output.

```
public class AnimalInheritance {
    public static void main(String[] args) {
        Animal animal = new Animal("Generic Animal");
        Dog dog = new Dog("Buddy");
        Cat cat = new Cat("Whiskers");

        animal.speak();
        dog.speak();
        cat.speak();
    }
}
```

Output:

An animal makes a sound.

Buddy barks.

Whiskers meows.

Task 8: Complete the **Shape**, **Circle**, **Rectangle** classes. So that it matches the output.

```
public class ShapeHierarchy {
    public static void main(String[] args) {
        Circle circle = new Circle(5.0);
        Rectangle rectangle = new Rectangle(4.0, 6.0);

        circle.displayType();
        rectangle.displayType();
    }
}
```

Output:

Type of shape: Circle

Type of shape: Rectangle

Task 9: Complete the **Employee**, **Manager**, **Developer** classes. So that it matches the output.

```
public class EmployeeHierarchy {
    public static void main(String[] args) {
        Manager manager = new Manager("Alice", "HR");
        Developer developer = new Developer("Bob", "Java");

        manager.displayInfo();
        developer.displayInfo();
    }
}
```

Output:

Name: Alice

Department: HR

Name: Bob

Programming Language: Java

Task 10: Complete the **Account**, **SavingsAccount**, **CheckingAccount** classes. So that it matches the output.

```
public class BankAccountHierarchy {
    public static void main(String[] args) {
        SavingsAccount savingsAccount = new SavingsAccount("SA001", 1000.0, 0.05);
        CheckingAccount checkingAccount = new CheckingAccount("CA001", 500.0, 200.0);

        savingsAccount.displayBalance();
        savingsAccount.addInterest();
        savingsAccount.displayBalance();

        checkingAccount.displayBalance();
        checkingAccount.withdraw(700.0);
        checkingAccount.displayBalance();
    }
}
```

Output:

Account SA001 balance: 1000.0

Interest added to account SA001

Account SA001 balance: 1050.0

Account CA001 balance: 500.0

Withdrew 700.0 from checking account CA001

Account CA001 balance: -200.0