Class Test 04(A): CSI 203 - Object Oriented Programming, Fall 2019
Total Marks: 20, Time: 20 minutes

Name: ID:

1) Define an interface name **BankOperation** and add the following 3 methods [10]
    a. void deposit(double amt)
    b. void withdraw(double amt)
    c. double getBalance()
2) Now create a concrete class **BankAccount** which should implement the **BankOperation** interface. [10]

Class Test 04(B): CSI 203 - Object Oriented Programming, Fall 2019
Total Marks: 20, Time: 20 minutes

Name: ID:

1) Create a user-defined exception and name it **LazyStudentException**. The exception will take a double value ***atn*** as the parameter and set the exception message to "Attendance can't be less than [atn]" where [atn] is the value of the ***atn*** parameter. [10]
2) Given the interface below, define a class **Game** and implement the **Operation** interface in this **Game** class. [10]

```
public interface Operation {
	void start();
	void stop();
}
```

Class Test 04(C): CSI 203 - Object Oriented Programming, Fall 2019
Total Marks: 20, Time: 20 minutes

Name: ID:

1) Give the following abstract class, define a class name **Car** which should be a subclass of the abstract class **Vehicle**. In **Car** class add a new attribute ***enginePower*** and write necessary codes so that compiler doesn't give any error. [10]

```
public abstract class Vehicle {
	String make, model;
	public Vehicle(String make, String model) {
		this.make = make;
		this.model = model;
	}

	abstract void operation();
}
```

2) What is wrong with the code below? List the errors and fix those. Do not delete any line. [10]

```
import java.security.InvalidParameterException;
import java.util.Scanner;

public class TextException {
  public static void main(String[] a){
	System.out.println("Please enter an integer");
	Scanner scan= new Scanner(System.in);
	int a=0;

	try{
	   a = scan.nextInt();
	   if(a>100) throw new InvalidParameterException("Can't be greater than 100");
	}
	System.out.println(a);
	catch(Exception e){	}
	catch(InvalidParameterException e){		}
   }
}
```