**SURIGAO STATE COLLEGE OF TECHNOLOGY**

*"For Nation's Greater Heights"*

| Document Code No. | FM-SSCT-ACAD-004 |
|---|---|
| Revision No. | 00 |
| Effective Date | 01 January 2019 |
| Page No. | 1 of 4 |

**COLLEGE OF ENGINEERING AND INFORMATION TECHNOLOGY**

Second Semester, AY 2021-2022

**TEST QUESTIONNAIRE**

**FINAL Examination in AC3/AC4 – OBJECT ORIENTED PROGRAMMING**

**Direction:  Write your answer in one long bond paper, if possible front page only.  Take picture and Turn in.**

**APPLICATION(Module 5) :  Apply what you have learned.**

1.      Write at least one program code with output specified in Module 5, Explain why and how the output is generated.

```
class Animal {

 // field and method of the parent class
 String name;
 public void eat() {
   System.out.println("I can eat");
 }
}


// inherit from Animal
class Dog extends Animal {
```

Prepared by: **UNIFE O. CAGAS, DTE**
               Faculty

Checked by: **MICHAEL P. YUJOCO, PhEdD**
               Program Chair, BEXET

**SURIGAO STATE COLLEGE OF TECHNOLOGY**

"For Nation's Greater Heights"

| Document Code No. | FM-SSCT-ACAD-004 |
| --- | --- |
| Revision No. | 00 |
| Effective Date | 01 January 2019 |
| Page No. | 2 of 4 |

```java
  // new method in subclass
  public void display() {
    System.out.println("My name is " + name);
  }
}

class Main {
  public static void main(String[] args) {

    // create an object of the subclass
    Dog labrador = new Dog();

    // access field of superclass
    labrador.name = "Rohu";
    labrador.display();

    // call method of superclass
    // using object of subclass
    labrador.eat();

  }
}
```

Output

My name is Rohu
I can eat


Prepared by: <u>**UNIFE O. CAGAS, DTE**</u>
                        Faculty

Checked by: **MICHAEL P. YUJOCO, PhEdD**
                    Program Chair, BEXET

**SURIGAO STATE COLLEGE OF TECHNOLOGY**

*"For Nation's Greater Heights"*

Since Dog inherits the field and method from Animal, we are able to access the field and method using the object of the Dog.

2.      (Module 6) Write at least one program code with output specified in Module 6, Explain why and how the output is generated.

```java
class Main {
  public static void main(String[] args) {

    try {

      // code that generate exception
      int divideByZero = 5 / 0;
      System.out.println("Rest of code in try block");
    }

    catch (ArithmeticException e) {
      System.out.println("ArithmeticException => " + e.getMessage());
    }
  }
}
```

Output

ArithmeticException => / by zero

we are trying to divide a number by 0. Here, this code generates an exception. To handle the exception, we have put the code, 5 / 0 inside the try block. Now when an exception occurs, the rest of the code inside the try block is skipped. The catch block catches the exception and statements inside the catch block is executed. If none of the statements in the try block generates an exception, the catch block is skipped.


**I.      ANALYSIS (Module 7):  Analyze and answer the question,  16  points**

Prepared by: <u>**UNIFE O. CAGAS, DTE**</u>
                 Faculty

Checked by: **MICHAEL P. YUJOCO, PhEdD**
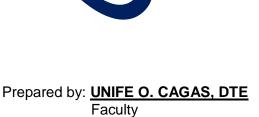                 Program Chair, BEXET

**SURIGAO STATE COLLEGE OF TECHNOLOGY**

*"For Nation's Greater Heights"*

Completely Explain AWT Packages in the development of graphic programs

The Abstract Window Toolkit (AWT) is Java's original platform-dependent windowing, graphics, and user-interface widget toolkit, preceding Swing. The AWT is part of the Java Foundation Classes (JFC) — the standard API for providing a graphical user interface (GUI) for a Java program. AWT is also the GUI toolkit for a number of Java ME profiles. For example, Connected Device Configuration profiles require Java runtimes on mobile telephones to support the Abstract Window Toolkit.

Prepared by: <u>**UNIFE O. CAGAS, DTE**</u>
 Faculty

Checked by: **MICHAEL P. YUJOCO, PhEdD**
 Program Chair, BEXET