**BIT 142**

Intermediate Programming

*Mini Assignment 3*

Please show all of your work for each problem:   
Please type your answers in

• Times New Roman Font

• Size 12

• No bolding, italics, and/or underlining

**• In your own words, define polymorphism (3 points):**it’s like using one block of code and changing which version of that block of code we’re using, depending on the inputs that we are giving it.

**• In your own words, define each principle of polymorphism (3 points):**

**Static polymorphism:** implementing multiple methods within the same class that use the same name but a different set of parameters. That is called method overloading and represents a static form of polymorphism.

**Dynamic polymorphism:** This form of polymorphism doesn’t allow the compiler to determine the executed method.

**• What is a base class and what is a derived class? (3 points)**

A base class is a class, in an object-oriented programming language, from which other classes are derived. It facilitates the creation of other classes that can reuse the code implicitly inherited from the base class (except constructors and destructors). A programmer can extend base class functionality by adding or overriding members relevant to the derived class.

A derived class is a class created or derived from another existing class. The existing class from which the derived class is created through the process of inheritance is known as a base class or superclass.

**• When would I use each?**It is sometimes the case that we have a class is nearly what we need, Derived classes acquire the properties of an existing class, The original class is called the base class.

**• What is a search algorithm? (3 points)**a search algorithm is any algorithm which solves the search problem, namely, to retrieve information stored within some data structure, or calculated in the search space of a problem domain, either with discrete or continuous values.

**• Solve the following inheritance mystery: (2** at 5 points each**)**

(see next page)

public class Bird

{

    public virtual void fly(){

        System.out.println("Bird can fly");

    }

}

public class Parrot : Bird

{

    public override void fly()

    {

        System.out.println("Parrot can fly");

    }

}

public class Program

{

    public static void Main(string[] args)

    {

        Parrot p = new Parrot();

        p.fly();

    }

}

**Output:**

**Bird Can Fly**

public class X

{

    public virtual void foo()

    {

        Console.WriteLine("Parent");

    }

}

public class Y : X

{

    public override void foo()

    {

        Console.WriteLine("Child");

    }

}

public class Program

{

    public static void Main(string[] args)

    {

        X c = new Y();

        c.foo();

    }

}

**Output:**

**Child**

**• What is a super method? (3 points)**calls the parent constructor with no arguments.

**• What is run-time? (3 points)**

Runtime is when a program is running or being executable.

**• Why is run-time important?**

 if errors are found in the code the program will throw runtime errors.

**• Answer the following questions about runtime and sorting (6 points)**

**• Show each step to find the value 47 in the list below using Binary Search ? (3 points)**

Min=2 Max=94

Mid=47

47=47

**• What is the run time and why?**  N/2 because n/2 items remaining and it only ran once.  
 **• What is the run-time of the following algorithm (6 points)**

a for (int i = 1; i <= 50; i++) {

......

}

**N**

--------------------------------------------------

b. for (int i = 1; i <= n; i++) {

for (int j = 1; j <= n; j++) {

. . . . .

}

}

**N^2**

--------------------------------------------------

c. for (int i = 1; i <= n; i++) {

. . . . . .

}

for (int j = n; j >= 1; j--) {

. . . . . .

}

**N**

--------------------------------------------------

d. for (int i = 1; i <= n; i++) {

for (int j = 1; j <= n; j++) {

for (int k = 1; k <= n; k++) {

}

}

}

**N^3**

--------------------------------------------------

• Polymorphism **(7 points)**

• Write a class named Animal

• Prints “I am an animal”

• Write a class named Giraffe

• Prints “I am animal”

• Has a height of 10 ft

• Weighs 80lbs

• Prints “I am big and tall!”

• Write a class named Cat

• Prints “I am a cat and I lovee to sleep…”

• Prints “zzz…” 24 times

• Write a class named Hippo

• Prints “I am animal”

• Has a height of 10 ft

• Weighs 80lbs

• Prints “I am big and tall!”

• Prints “My name is Happy the Hippo!” 100 times

• Create a main program that calls the 4 classes in any way you like to produce a unique output

class Animal {

public static void a() {

System.out.println("I am an animal");

}

}

class Giraffe {

public static void a() {

System.out.println("I am an animal");

System.out.println("Has a height of 10 ft");

System.out.println("Weighs 80lbs");

System.out.println("I am big and tall!");

}

}

class Cat {

public static void a() {

System.out.println("I am a cat and I lovee to sleep…");

for (int i = 1; i <= 24; i++) {

System.out.println("zzz...");

}

}

}

class Hippo {

public static void a() {

System.out.println("I am an animal");

System.out.println("Has a height of 10 ft");

System.out.println("Weighs 80lbs");

System.out.println("I am big and tall!");

for (int i = 1; i <= 100; i++) {

System.out.println("My name is Happy the Hippo!");

}

}

}

public class Test {

public static void main(String[] args) {

Animal.a();

Giraffe.a();

Cat.a();

Hippo.a();

}

}