

CS 2334: Programming Structures and Abstractions: Exam 1  
October 2, 2017

General instructions:

- Please wait to open this exam booklet until you are told to do so.
- This examination booklet has 15 pages. You also have been issued a bubble sheet.
- Fill in the identifying information below (signature, name, ID and date) Also, write your name and ID number on your bubble sheet, and fill in the bubbles for your ID.
- You may have up to two pages of your own notes. No electronic devices or books may be used.
- The exam is worth a total of 137 points. Your grade counts for 10% of your final grade.
- You have 1.25 hours to complete the exam. Be a smart test taker: if you get stuck on one problem go on to the next.
- Use your bubble sheet to answer all multiple-choice questions. Make sure that the question number and the bubble row number match.
- Other than **this** page, you may tear any other page out of this booklet that does not contain numbered answers.
- If you cannot effectively erase erroneous answers from the bubble sheet, please clearly cross them out.

On my honor, I affirm that I have neither given nor received inappropriate aid in the completion of this exam.

**Signature:** \_\_\_\_\_ **Name:** \_\_\_\_\_

**ID Number:** \_\_\_\_\_ **Date:** \_\_\_\_\_

Question	Points	Score
Types and Objects	32	
Inheritance and Polymorphism	35	
UML and Object Oriented Design	13	
Abstract Classes and Interfaces	28	
Exceptions and Error Handling	25	
Memory Management	4	
Total:	137	

Part I. Types and Objects

1. (4 points) What is printed by this block of code?

```
1   int a = 2;
2   String b = "5";
3   int c = 9;
4   System.out.println(c + a + b);
```

A. 11   B. 16   **C. 115**   D. 925   E. Compilation error or answer not shown

2. (4 points) What is printed by this block of code?

```
1   int i1 = 7;
2   int i2 = 5;
3   System.out.println(i2/i1);
```

**A. 0**   B. 0.625   C. 1   D. 1.6   E. Compilation error or answer not shown

3. (4 points) What is printed by this block of code?

```
1   String s1 = "LuH";
2   String s2 = "luh";
3   s2 = s2.toUpperCase();
4   if (s1.equals(s2))
5   {
6       System.out.println("Yes:" + s1);
7   }
8   else
9   {
10      System.out.println("No:" + s2);
11  }
```

A. No:LuH   **B. No:LUH**   C. Yes:LuH   D. Yes:luh  
E. Compilation error or answer not shown

4. (4 points) What is printed by this block of code?

```
1   Double a = 1.4;
2   Integer b = -5.3;
3   Integer c = new Integer(19);
4   System.out.println(c + b + a);
```

A. 15.1   B. 15.4   C. 25.7   D. 19-5.31.4  
**E. Compilation error or answer not shown**

5. (4 points) What is printed by this block of code?

```
1   int a = 3;
2   int b = 7;
3   String c = 8;
4   System.out.println(c + a + b);
```

A. 18   B. 810   C. 837   D. 873   E. **Compilation error or answer not shown**

6. (4 points) What is printed by this block of code?

```
1   int a = 5;
2   String b = "20";
3   System.out.println(b + Integer.parseInt(b) + a);
```

A. 45   B. 2025   C. 5520   D. **20205**  
E. Compilation error or answer not shown

7. (4 points) What is printed by this block of code?

```
1   int a = 42;
2   String b = "8";
3   System.out.println(b + a);
```

A. 50   B. 336   C. 428   D. **842**   E. Compilation error or answer not shown

8. (4 points) What is printed by this block of code?

```
1   int i1 = 2;
2   int i2 = 3;
3   int i3 = 5;
4   System.out.println(i3 + i2 * i1);
```

A. **11**   B. 13   C. 16   D. 17   E. Compilation error or answer not shown

## Part II. Inheritance and Polymorphism

Consider the following class definitions:

```
public class X
{
    protected int id;

    public X(int id)
    {
        this.id = id;
    }

    public int getID()
    {
        return this.id;
    }

    public String toString()
    {
        return "X: " + id;
    }

    public String getDescriptor()
    {
        return "D: " + this.getID();
    }
}

public class Y extends X
{
    protected String s;

    public Y(String s, int val)
    {
        super(val);
        this.s = s;
    }

    public String toString()
    {
        return this.s + ": " + this.id;
    }

    public String superString()
    {
        return super.toString();
    }
}

public class Z extends Y
{
    private int id;

    public Z(String s, int id)
    {
        super(s, id+2);
        this.id = id;
    }

    public int getID()
    {
        return this.id;
    }
}
```

9. (7 points) What is printed by this block of code?

```
X x = new X(17);  
System.out.println(x.superString());
```

A. D: 17   B. X: 17   C. D: 19   D. X: 19   **E. Answer not shown**

10. (7 points) What is printed by this block of code?

```
X x = new X(17);  
System.out.println(x.getDescriptor());
```

**A. D: 17**   B. X: 17   C. D: 19   D. X: 19   E. Answer not shown

11. (7 points) What is printed by this block of code?

```
X x = new Z("FOO", 42);  
System.out.println(x);
```

**A. FOO: 44**   B. D: 42   C. D: 44   D. X: 44   E. Answer not shown

12. (7 points) What is printed by this block of code?

```
Y y1 = new Y("BAR", 7);  
System.out.println(y1.superString());
```

A. BAR: 7   B. D: 7   **C. X: 7**   D. X: 9   E. Answer not shown

13. (7 points) What is printed by this block of code?

```
Y y1 = new Y("BAR", 7);  
System.out.println(y1);
```

**A. BAR: 7**   B. D: 7   C. X: 7   D. X: 9   E. Answer not shown

Part III. UML and Object Oriented Design

14. (5 points) Which UML diagram corresponds to the following code?

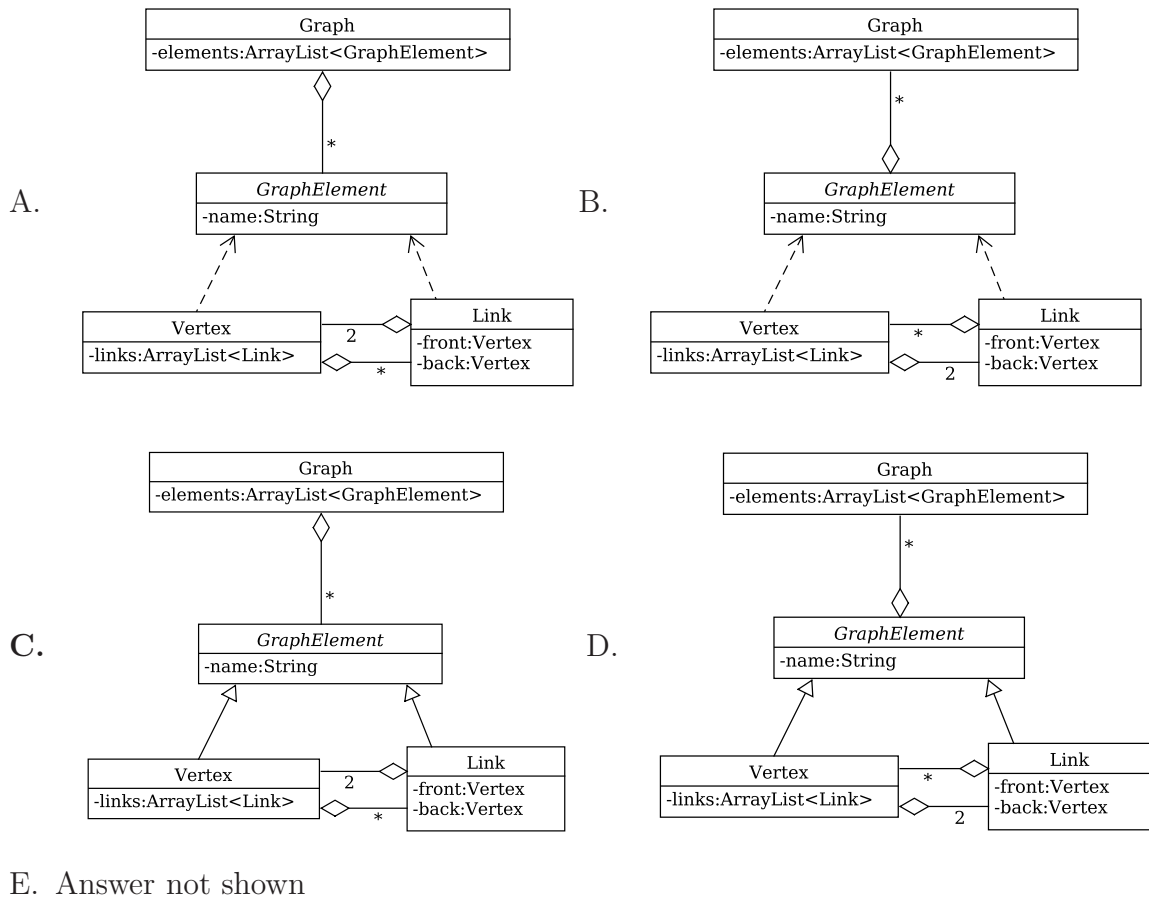
```

public class Graph
{
    private ArrayList<GraphElement> elements;
}

public abstract class GraphElement
{
    private String name;
}

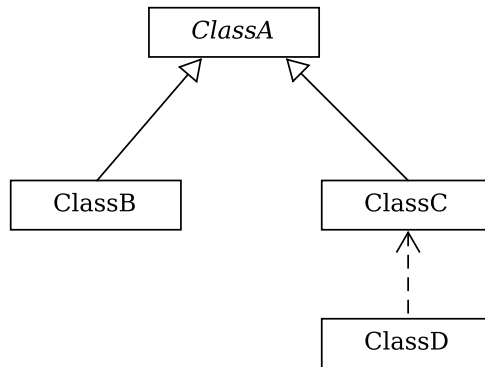
public class Vertex extends GraphElement
{
    private ArrayList<Link> links;
}

public class Link extends GraphElement
{
    private Vertex front;
    private Vertex back;
}
    
```



**Solution:** GraphElement.name was declared on the exam with default visibility. This makes answer E viable.

15. (4 points) Which set of class definitions corresponds to the following UML diagram?



- A. 

```
public abstract class ClassA {...}
public class ClassB extends ClassA {...}
public class ClassC extends ClassA {...}
public class ClassD extends ClassA {...}
```
- B. 

```
public abstract class ClassA {...}
public class ClassB extends ClassA {...}
public class ClassC extends ClassA {...}
public class ClassD {...}
```
- C. 

```
public abstract class ClassA {...}
public class ClassB extends ClassA {...}
public class ClassC extends ClassA {...}
public class ClassD implements ClassC {...}
```
- D. 

```
public abstract class ClassA {...}
public class ClassB extends ClassA {...}
public class ClassC extends ClassA {...}
public class ClassD extends ClassC {...}
```
- E. Answer not shown



16. (4 points) **Carefully examine** the following UML models and select the one that corresponds to the following code.

```

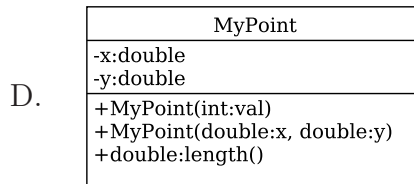
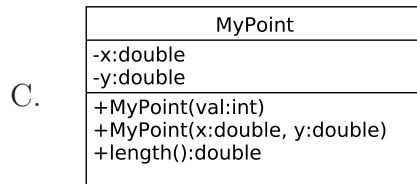
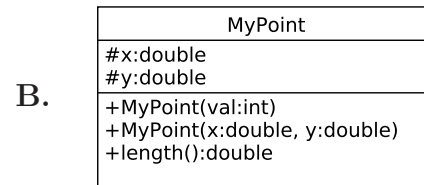
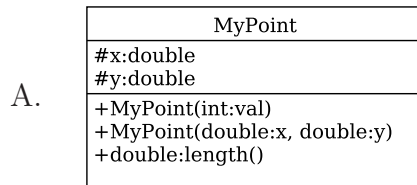
public class MyPoint
{
    protected double x;
    protected double y;

    public MyPoint(int val)
    {
        this.x = this.y = val;
    }

    public MyPoint(double x, double y)
    {
        this.x = x;
        this.y = y;
    }

    public double length()
    {
        return(Math.sqrt(x * x + y * y));
    }
}

```



E. Answer not shown

Part IV. Abstract Classes and Interfaces

Consider the following class definition for the next three questions:

```
public class NamedInteger implements Comparable<NamedInteger>
{
    private Integer val;
    private String name;

    public NamedInteger(int val, String name)
    {
        this.val = val;
        this.name = name;
    }

    public int compareTo(NamedInteger ni)
    {
        int ret = val.compareTo(ni.val);
        if(ret != 0)
        {
            return ret;
        }
        else
        {
            return -name.compareTo(ni.name);
        }
    }

    public static void main(String[] args)
    {
        NamedInteger i1 = new NamedInteger(5, "Bob");
        NamedInteger i2 = new NamedInteger(7, "Sarah");
        NamedInteger i3 = new NamedInteger(5, "Skip");
        NamedInteger i4 = new NamedInteger(42, "Alice");

        System.out.println(i1.compareTo(i2)); // Line 1
        System.out.println(i1.compareTo(i3)); // Line 2
        System.out.println(i4.compareTo(i2)); // Line 3
    }
}
```

17. (4 points) What integer value is printed at Line 1?  
A. Positive value    B. Zero    **C. Negative value**    D. Answer not shown
18. (4 points) What integer value is printed at Line 2?  
**A. Positive value**    B. Zero    C. Negative value    D. Answer not shown
19. (4 points) What integer value is printed at Line 3?  
**A. Positive value**    B. Zero    C. Negative value    D. Answer not shown

20. (4 points) Which one line (if any) will cause the program not to compile?

```
1 public interface InterfaceB
2 {
3     public abstract void foo(int i);
4 }
5
6 public abstract class ClassB implements InterfaceB
7 {
8     private double val;
9
10    public ClassB(double val)
11    {
12        this.val = val;
13    }
14
15    public abstract void foo(double val);
16 }
```

A. 1 B. 3 C. 6 D. 15 E. This code will compile

21. (4 points) Which one line (if any) will cause the program not to compile?

```
1 public interface InterfaceA
2 {
3     public abstract void mult(int i);
4 }
5
6 public class ClassA implements InterfaceA
7 {
8     double i;
9
10    public void mult(double i)
11    {
12        this.i *= i;
13    }
14 }
```

A. 1 B. 8 C. 10 D. 12 E. This code will compile

22. (4 points) Which one line (if any) will cause the program not to compile?

```
1 public class AbstractClass
2 {
3     private String name;
4
5     public AbstractClass(String name)
6     {
7         this.name = name;
8     }
9
10    public abstract String getName();
11 }
```

A. 1 B. 3 C. 5 D. 7 E. This code will compile

23. (4 points) Any class that implements an interface must provide implementations for all of the abstract methods.

A. True    **B. False**

**Solution:** If an abstract class extends an interface, then it may leave some of the interface's methods unimplemented.

## Part V. Exceptions and Error Handling

Consider the following program:

```
public class ExceptionTest
{
    public static int foo(Integer a)
    {
        if(a > 8)
        {
            throw new IllegalArgumentException("Too big");
        }
        return 5 / (a - 2);
    }

    public static int bar(Integer b)
    {
        Integer a = null;

        try
        {
            if(b > 5)
            {
                return(foo(b+2));
            }
            else if(b < 0)
            {
                return(foo(a));
            }
            else
            {
                return(foo(b));
            }
        }
        catch(NullPointerException e)
        {
            return 5;
        }
    }
}
```

Note that *IllegalArgumentException* is a *RuntimeException*.

24. (6 points) What is displayed when the following code is executed?

```
System.out.println(bar(2));
```

A. 0   B. 2   C. 5   D. IllegalArgumentException   **E. Answer not shown**

25. (6 points) What is displayed when the following code is executed?

```
System.out.println(bar(7));
```

A. 0   B. 2   C. 5   **D. IllegalArgumentException**   E. Answer not shown

26. (6 points) What is displayed when the following code is executed?

```
System.out.println(bar(null));
```

A. 0   B. 2   **C. 5**   D. IllegalArgumentException   E. Answer not shown

27. (7 points) What is displayed when the following code is executed?

```
System.out.println(bar(4));
```

A. 0   **B. 2**   C. 5   D. IllegalArgumentException   E. Answer not shown

## Part VI. Memory Management

Consider the following program:

```
1  public class MyClass
2  {
3      private double val;
4      protected static char c = 'a';
5
6      public MyClass(double val) {
7          this.val = val;
8      }
9
10     private void print() {
11         System.out.println(c);
12     }
13
14     public double doubleValue() {
15         return val * 2;
16     }
17
18     public static void main(String[] args)
19     {
20         MyClass m = new MyClass('a', 7.2);
21         int a = 4;
22
23         System.out.println(m + a);
24     }
25 }
```

28. (2 points) In which part of memory is the variable declared on line 6 stored?  
A. Heap    **B. Stack**    C. Answer not shown
29. (2 points) In which part of memory is the variable declared on line 4 stored?  
**A. Heap**    B. Stack    C. Answer not shown