Name:_____ NetID:_____

Answer all questions in the space provided. Write clearly and legibly, you will not get credit for illegible or incomprehensible answers. This is a closed book exam. However, each student is allowed to bring one page of notes to the exam. Print your name at the top of every page.

Question:	1	2	3	4	5	6	7	8	Total
Points:	16	24	8	6	12	12	12	10	100
Score:									

1. Multiple choice questions: Select the single correct answer for each.

   (a) Which interface would be the best choice to associate student usernames with `Student` objects? (2)

   A. `Collection` B. `Deque` C. `List` D. `Map` E. `Set` F. `Queue`

   (b) Which interface would be the best choice to hold a collection of unique student usernames? (2)

   A. `Collection` B. `Deque` C. `List` D. `Map` E. `Set` F. `Queue`

   (c) Which interface would be the best choice as a parameter of a method that can accept any collection of student usernames, without assuming anything about order or uniqueness? (2)

   A. `Collection` B. `Deque` C. `List` D. `Map` E. `Set` F. `Queue`

   (d) Which interface would be the best choice if you wanted to handle items in a LIFO (last in, first out) order? (2)

   A. `Collection` B. `Deque` C. `List` D. `Map` E. `Set` F. `Queue`

   (e) Which interface would be the best choice if you wanted to be able to randomly shuffle the order of the items? (2)

   A. `Collection` B. `Deque` C. `List` D. `Map` E. `Set` F. `Queue`

   (f) Which type could `foo` be in the following code snippet? (2)

```
boolean success = foo.contains("hello");
```

   A. `Collection` C. `List` E. `Queue` G. Any of these.
   B. `Deque` D. `Set` F. `SortedSet` H. None of these.

   (g) Which type could `foo` be in the following code snippet? (2)

```
Object obj = foo.get(0);
```

   A. `Collection` C. `List` E. `Queue` G. Any of these.
   B. `Deque` D. `Set` F. `SortedSet` H. None of these.

   (h) Which of the following correctly declares and instantiates a map that associates String keys with Double values? (2)

   A. `Map<String, Double> map = new HashMap<String, Double>();`
   B. `HashMap<String, Double> map = new HashMap<String, Double>();`
   C. `HashMap<String, Double> map = new HashMap<>();`
   D. `Map<String, Double> map = new TreeMap<String, Double>();`
   E. `Map<String, Double> map = new HashMap<>();`
   F. `Map<String, Double> map = new TreeMap<>();`
   G. All of the above.
   H. None of the above.

2. Write the answer in the blank provided.

(a) What is the keyword used to make a variable that cannot be reassigned? (2)

(a) _____

(b) What is the keyword used to access a member variable hidden by a local variable? (2)

(b) _____

(c) What is the keyword used to call the parent class version of a method overridden in the child class? (2)

(c) _____

(d) What is the keyword used to make a class that cannot be instantiated? (2)

(d) _____

(e) What is the keyword used to test if an object is of a particular type? (2)

(e) _____

(f) What is the keyword used to make a variable or method belong to a *class* rather than an *instance*? (2)

(f) _____

(g) If I have a member variable in a top level class that is visible to methods within a class nested inside it, but not to methods within another top level class in the same package, what access modifier have I used on the member variable? (2)

(g) _____

(h) If I declare an `int` variable but do not initialize it, what is its value? (2)

(h) _____

(i) If I declare an `Integer` variable but do not initialize it, what is its value? (2)

(i) _____

(j) What is occurring when I have a method in a child class that has the same name as a method in its parent, but a different parameter list? (2)

(j) _____

(k) If I have a variable of type `Object`, an object of any type could be assigned to it. If I call the `toString` method on this variable, how does Java know which version of the method to execute? (2)

(k) _____

(l) What interface does a class implement if you can use an instance of the class in an enhanced for loop (also known as a for-each loop)? (2)

(l) _____

3. Why don't the following code snippets compile? Select the single correct answer for each.

(a) (2)

```
public class MyClass {
    private int x = 10;
    public static void main(String[] args) {
        x++;
        System.out.println(x);
    }
}
```

A. Cannot access private variable x from a public method.
B. Cannot access x without an instance of MyClass.
C. Variable x is a constant, so cannot be incremented in main.
D. The println method expects a String, not an int.
E. Some other error.
F. This code will successfully compile.

(b) (2)

```
List<boolean> truthValues;
```

A. The list is not initialized.
B. Cannot use interface List as variable type.
C. Cannot use primitive type boolean as generic type parameter.
D. List is an interface and cannot be instantiated.
E. Some other error.
F. This code will successfully compile.

(c) (2)

```
ActionListener listener = new ActionListener() {
    public void actionPerformed( ActionEvent ev ) {
      System.out.println( "Action happened" );
    }
};
```

A. ActionListener is an interface and cannot be instantiated.
B. Missing semicolon after call to ActionListener constructor.
C. Extra curly braces around actionPerformed method.
D. Need to add the listener to a JButton.
E. Some other error.
F. This code will successfully compile.

(d) (2)

```
public class MyListener implements ActionListener {
    private int count = 0;
    public void actionPerformed( ActionEvent ev ) {
      count++
    }
}
```

A. Missing constructor for MyListener class.
B. Should use "extends" instead of "implements" in first line.
C. Cannot access private variable count from a public method.
D. Variable count is out of scope in the actionPerformed method.
E. Some other error.
F. This code will successfully compile.

4. Consider the following code. (6)

```
import java.util.*;

public class MyPoint {

    private final int x;
    private final int y;

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

    public int getX() { return x; }
    public int getY() { return y; }

    public String toString() {
        return "(" + x + ", " + y + ")";
    }

    public static void main(String[] args) {

        List<MyPoint> points = new ArrayList<>();

        points.add(new MyPoint(1,2));
        points.add(new MyPoint(3,4));
        points.add(new MyPoint(5,6));

        System.out.println("points = " + points);

        MyPoint p = new MyPoint(1,2);
        System.out.println("Is " + p + " in points? " + points.contains(p));
    }
}
```

The output of this code is

```
points = [(1, 2), (3, 4), (5, 6)]
Is (1, 2) in points? false
```

- Why isn't my point found in the list?
- What would I need to do to fix this without changing the main method?

5. Consider the following classes. What is the output of this code? (12)

```
public class Foo {
    protected double x;
    protected static int y = 10;
    protected String z;

    public Foo() {
        this("Spring", 20);
        y++;
    }

    public Foo(String x) {
        this.x = x.length() / 4.0;
        this.z = x;
        y++;
    }

    public Foo(String x, int y) {
        this(x);
        y++;
        System.out.println(y);
    }

    public void print(String y) {
        System.out.println(x);
        System.out.println(y);
    }

    public void print(double z) {
        System.out.println(y);
        System.out.println(z);
    }
}
```

```
public class Bar extends Foo {

    public Bar(String x) {
        super(x);
        y++;
        System.out.println(z);
    }

    public Bar(String x, int y) {
        y++;
        System.out.println(x + y);
        System.out.println(z);
    }

    public void print(int x) {
        System.out.println(x + y);
        print(x * 1.2);
    }

    public void print(String z) {
        print(x+y);
        super.print(z);
    }

    public static void main(String[] args) {
        Foo x = new Bar("CS", 250);
        x.print("Final");
        Bar y = new Bar("Exam");
        y.print(4);
    }
}
```

6. Consider the following classes. (12)

```
public class Parent {

    public void methodA() {
        System.out.println("A");
    }

    public void methodB() {
        System.out.println("B");
    }
}
```

```
public class Child extends Parent {

    public void methodA(int n) { }
    public void methodB() { }
    public void methodC() { }

    public static void main(String[] args) {

        Parent x = new Parent();
        Child y = new Child();
        Parent z = new Child();

        // What works here?
    }
}
```

Which of the following lines of code would successfully compile and run when placed in the `main` method of `Child` after the code that is already there? Select all that apply.

A. `x.methodA();`	G. `x.methodB();`	M. `Parent p = x;`	S. `Child c = (Child)x;`
B. `y.methodA();`	H. `y.methodB();`	N. `Parent p = y;`	T. `((Child)x).methodC();`
C. `z.methodA();`	I. `z.methodB();`	O. `Parent p = z;`	U. `((Child)z).methodC();`
D. `x.methodA(42);`	J. `x.methodC();`	P. `Child c = x;`	V. `Object obj = new Parent();`
E. `y.methodA(42);`	K. `y.methodC();`	Q. `Child c = y;`	W. `Parent obj = new Parent();`
F. `z.methodA(42);`	L. `z.methodC();`	R. `Child c = z;`	X. `Child obj = new Parent();`

7. Consider the following interface describing a to-do list with tasks represented as Strings. (12)

```
import java.util.*;
public interface ToDoInterface {

    /**
     * Adds a task to the todo list.
     * @param task The task to add.
     */
    void addTask(String task);

    /**
     * Removes a task from the todo list after completion.
     * (It is possible the completed task is not the expected next task.)
     * @param task The completed task.
     */
    void completeTask(String task);

    /**
     * Get the next task from the todo list.
     * (Does not remove the task from the todo list.)
     * @return Next task to do, or null if none left.
     */
    String getNextTask();
```

```
    /**
     * Get all the tasks on the todo list.
     * @return All tasks we still have to do.
     */
    Collection<String> getAllTasks();
}
```

Write a simple class that implements this interface, handling the tasks in a first-in-first-out order. Use a private member variable of an appropriate collection type to hold the tasks. You do not need to include Javadoc comments.

8. Consider the following program. What would be displayed when it is run? Draw a picture to illustrate. (10)

```
import java.awt.*;
import javax.swing.*;

public class LayoutExample {

    public static void createAndShowGUI() {
        JFrame frame = new JFrame("This is the title");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JPanel x = new JPanel();
        x.add(new JButton("Button A"));
        x.add(new JButton("Button B"));

        JPanel y = new JPanel();
        y.setLayout(new BoxLayout(y, BoxLayout.PAGE_AXIS));
        y.add(new JButton("Button C"));
        y.add(new JButton("Button D"));
        y.add(new JButton("Button E"));

        frame.add(x, BorderLayout.PAGE_END);
        frame.add(y, BorderLayout.LINE_START);
        frame.add(new JButton("Big Button"), BorderLayout.CENTER);

        frame.pack();
        frame.setVisible(true);
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
                public void run() {
                    createAndShowGUI();
                }
            });
    }
}
```