CCSF CS 111B Practice Test

Total points 12/17

Please allow yourself 30 minutes for this practice test.

A passing score is 13 correct. There is no penalty for guessing.

This is a closed book test; you may use no outside resources of any type for this test.

What is your CCSF student ID number? (If you aren't sure, you may put your @mail.ccsf.edu email address.) *

WA2762167

✓ Which of the following compares the contents of two String objects, string1 and string2, to see if they have the same contents? 1/1

- (●) a) if (string1.equals(string2)) ✓
- ( ) b) if (string1 == string2)
- ( ) c) if (string1 = string2)
- ( ) d) None of the above

For the next question, consider the following code:

```
public class TestIt extends Object {
    // ...
}
```

✓ Is the code above legal? 1/1

- (●) a) Yes ✓
- ( ) b) No

For the next question, consider the following code:

```
String[] s = new String[10];
System.out.println(s[0].length() );
```

✓ What does the above code print? 1/1

- ( ) a) 0
- ( ) b) 10
- ( ) c) Nothing is printed, and the code is executed successfully.
- (●) d) The code causes an error. ✓

For the next question, consider the following code:

```
public class Testing {
    public static void main (String [] args ) {
        try {
            System.out.print("Welcome");
            int i = 0, y = 2 / i;
            System.out.print("Hello");
    }
    catch (RuntimeException e) {
        System.out.print("Greetings");
    }
    finally {
        System.out.print("End");
    }
}
```

✗ What does the code above print? 0/1

- (●) a) WelcomeHello ✗
- ( ) b) WelcomeGreetingsEndHello
- ( ) c) WelcomeHelloGreetingsEnd
- ( ) d) WelcomeGreetingsEnd

Correct answer

- (●) d) WelcomeGreetingsEnd

For the next question, consider the following code:

```
int[] x = {5,6,7,8,9}
int[] y = x;
y[2] = 3;
```

✓ Which of the following statements is true about the code above, after it executes? 1/1

- ( ) a) x[2] has the value 7.
- (●) b) x[2] has the value 3. ✓
- ( ) c) x[2] has the value 6.
- ( ) d) x[3] has the value 7.

For the next question, consider the following method header:

```
void myM(int x, double y)
```

✓ Which of the following is an appropriate call for the method header above? 1/1

- ○ a) int x = myM(3, 0.5);
- ◉ b) myM(3, 0.5); ✓
- ○ c) System.out.println(myM(3, 0.5) * 10);
- ○ d) None of the above

For the next question, consider the following code:

```
public class Tester {
    public static void main( String [] args ) {
        int a[] = {11, 22, 33};
        thisMethod( a );
        System.out.println(a[1] + " " + a[2]);
    }
    public static void thisMethod(int arr[] ) {
        arr[1] = arr[0];
    }
}
```

✓ What is displayed after the program above is executed? 1/1

- ◉ a) 11 33 ✓
- ○ b) 11 22
- ○ c) 22 33
- ○ d) None of the above

For the next question, consider the following code:

```
public class Tester {
    public static void main( String [] args ) {
        int a[] = {11, 22, 33};
        thisMethod( a[1] );
        System.out.println(a[1] + " " + a[2]);
    }
    public static int thisMethod(int num ) {
        return num++;
    }
}
```

✗ What is displayed after the program above is executed? 0/1

- a) 11 22
- b) 22 33
- c) 12 33
- d) 23 33 ✗

Correct answer

- b) 22 33

✗ When a reference to object1 is reassigned to object2, and no other variable refers to object1, what happens? 0/1

- ( ) a) object1 must be deallocated by the programmer.
- ( ) b) Automatic garbage collection will free up the memory used by object1.
- ( ) c) object1 must remain in memory until the system is booted.
- (•) d) None of the above ✗

Correct answer

- (•) b) Automatic garbage collection will free up the memory used by object1.

✓ Complete the statement with the best option - a class that implements an interface "promises" to provide method definitions for: 1/1

- ( ) a) private methods within the interface.
- ( ) b) none of the methods described within the interface.
- (•) c) any undefined public methods described within the interface. ✓

For the next two questions, suppose the class Fruit is the parent of class Apple, and suppose the method makeAppleJuice is defined for the Apple class. Consider the following code:

```
Fruit fruit = new Apple( );
Apple pippen = new Apple( );
```

✗ Given the information above, which of the following statements is correct? 0/1

- a) fruit can invoke the method makeAppleJuice.
- b) pippen can invoke the method makeAppleJuice.
- c) (a) and (b) ✗
- d) None of the above

Correct answer

- b) pippen can invoke the method makeAppleJuice.

✓ Given the information in the previous question, which of the following best describes the reference "fruit"? 1/1

- a) interface reference
- b) polymorphic reference ✓
- c) abstract reference
- d) encapsulated reference

For the next question, consider the following code:

```
public class A1 {
    public int x;
    private int y;
    protected int z;
    // ...
}

public class A2 extends A1 {
    protected int a;
    private int b;
    // ...
}

public class A3 extends A2 {
    private int q;
    // ...
}
```

✓ Given the code above, which of the following lists of instance data is directly accessible within A3? 1/1

- a) x, y, z, a, b, q
- b) a, b, q
- c) a, q
- d) x, z, a, q ✓
- e) x, a, q

For the next three questions, consider the following code:

```
public class AClass {
	protected int x;
	protected int y;

	public AClass(int a, int b) {
		x = a;
		y = b;
	}

	public int addEm() {
		return x + y;
	}

	public void changeEm() {
		x++;
		y--;
	}

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

Suppose you want to extend AClass to BClass. BClass will have a third int instance data, z. Consider the following possible definitions of the BClass constructor, labeled with Roman numerals:

(i)
```
public BClass(int a, int b, int c) {
    super(a, b, c)
}
```

(ii)
```
public BClass(int a, int b, int c) {
    x = a;
    y = b;
    z = c;
}
```

(iii)
```
public BClass(int a, int b, int c) {
    z = c;
}
```

(iv)
```
public BClass(int a, int b, int c) {
    super(a, b);
    z = c;
}
```

(v)
```
public BClass(int a, int b, int c) {
    super();
}
```

✓ Which is the best definition above for the BClass constructor? 1/1

- a) (i)
- b) (ii)
- c) (iii)
- d) (iv) ✓
- e) (v)

✓ Assume you are still implementing BClass based on the definition of AClass above. Suppose that for BClass, you want addEm to add all three values and return the sum. Suppose you also want changeEm to change x and y, but leave z alone. Which of the following would be the best action to take? 1/1

- a) Redefine addEm and changeEm without referencing super.addEm() or super.changeEm().
- b) Redefine addEm to return the value of z + super.addEm(), but leave changeEm alone. ✓
- c) Redefine changeEm to call super.changeEm() and then set z = x + y, but leave addEm alone.
- d) Redefine addEm to return the value of z + super.addEm() and redefine changeEm to call super.changeEm() and then set z = x + y.
- e) Redefine changeEm to call super.changeEm() without doing anything to z, and redefine addEm to return super.addEm().

Assume you are still implementing BClass based on the definition of AClass above. Consider the following possible redefinitions of the toString() method in BClass, labeled with Roman numerals:

(i)
```
public String toString(int z) {
    return " " + x + " " + y + " " + z;
}
```

(ii)
```
public String toString() {
    return super.toString();
}
```

(iii)
```
public String toString() {
    return super.toString() + " " + z;
}
```

(iv)
```
public String toString() {
    return super.toString() + " " + x + " " + y + " " + z;
}
```

(v)
```
public String toString() {
    return " " + x + " " + y + " " + z;
}
```

✓ Which of the above options is the best redefinition of toString for BClass? 1/1

- a) (i)
- b) (ii)
- c) (iii) ✓
- d) (iv)
- e) (v)

For the next problem, consider the following code:

```
public class Figure {
	public void display() {
		System.out.println("Figure");
	}
}

public class Rectangle extends Figure {
	public void display() {
		System.out.println("Rectangle");
	}
}

public class Box extends Figure {
	public void display() {
		System.out.println("Box");
	}
}

public class Inherit {
	public static void main(String[ ] args) {
		Figure f = new Figure();
		Rectangle r = new Rectangle();
		Box b = new Box();
		f.display();
		f = r;
		f.display();
		f = b;
		f.display();
	}
}
```

Now, consider the following possible outputs for the code above, labeled with Roman numerals:

(i)
Figure
Rectangle
Box

(ii)
Rectangle
Box

(iii)
Figure
Figure
Figure

✗ Which of the following outputs above is correct for the code above? 0/1

- ○ a) (i)
- ○ b) (ii)
- ○ c) (iii)
- ◉ d) The code has a syntax error and won't compile or execute. ✗
- ○ e) None of the above

Correct answer

- ◉ a) (i)

Please indicate whether you followed all the rules for this test. The rules were: (1) no outside resources and (2) a 30-minute time limit. (Since this is only a practice test, it is okay if you did not follow the rules; however, we need to know this for data validation purposes.) *

- ◉ Yes, I followed all rules.
- ○ No, I did not follow all rules.