Exam 1 - CS 112 - Fall 2016 - Section 1

Name:

You may ***not*** discuss this exam with any student in ***either*** section of the course who has not completed the exam. You are reminded that you are required to uphold the USF Academic Honor Pledge below.

"*I pledge to demonstrate the core values of the University of San Francisco by upholding the standards of honesty and integrity, excellence in my academic work, and respect for others in my educational experiences, including supporting USF's mission.*"

Question	**Awarded**	**Question**	**Awarded**
Multiple Choice and True/False - 3 points per question			
1		6	
2		7	
3		8	
4		9	
5		10	
Short Answer - 5 points per question			
11		15	
12		16	
13		17	
14		18	
Programming - 10 points per question			
19		21	
20			
Total			

1. Which of the following are valid ***variable*** names? Select all identifiers that the compiler will accept. *Hint, the compiler does not enforce the style guidelines required for your assignments.*
    a. `abc123`
    b. `abc_123`
    c. `abC123`
    d. `12ab3c`
    e. `ABC_123`

2. Which of the following are methods that may be called on an instance of `Scanner.` Select all that apply.
    a. `add`
    b. `next`
    c. `useDelimiter`
    d. `nextInt`
    e. `charAt`

3. Which of the following are valid declarations of an `ArrayList.` Select all that apply.
    a. `ArrayList<String> words = new ArrayList<String>();`
    b. `ArrayList<Student> students = new ArrayList<Student>();`
    c. `ArrayList<int> numbers = new ArrayList<int>();`
    d. `ArrayList<Double> scores = new ArrayList<scores>();`

4. What is the *return type* of the `equals` method of the `String` class?
    a. `char`
    b. `int`
    c. `boolean`
    d. `None of the above`

5. Which of the following classes must be imported before they may be used in a program? Select all that apply.
    a. `java.util.ArrayList`
    b. `java.lang.String`
    c. `java.io.File`
    d. `java.util.Scanner`

6. True / False - A class *must* have getters and setters for all data members.

7. True / False - A constructor takes no parameters.

8. True / False - All methods *must* have a `return` statement in the body.

9. True / False - The `main` method in java is declared `static`.

10. True / False - It is common practice to use all capital letters to name a constant variable.

11. List at least five of the eight *primitive types* in Java.

12. Briefly describe what happens during compilation.

13. What is the output of the following code fragment?

```
String s = "cat";
s.replaceAll("t", "b");
System.out.println(s);
```

14. We discussed several equivalent statements that will add one to or subtract one from a variable. Write a statement that is the equivalent to the following:

```
i++;
```

15. How many iterations will the following `for` loops execute?

```
a. for (int i = 0; i < 20; i++) {}
```

```
b. for (int i = 20; i > 0; i--) {}
```

16. The following code results in a compiler error. Explain why.

```
        public class Exam1 {
              public static void main(String[] args) {
                    x = 10;
                    System.out.println("x: " + x);
              }
        }
```

17. What is the output of the following code fragment?

```
  public class Exam1 {
        public static void main(String[] args) {

              String s1 = new String("hello");
              String s2 = s1;

              if(s1.equals(s2)) {
                    System.out.println("result of .equals is
  true");
              }
              if(s1 == s2) {
                    System.out.println("result of == is true");
              }

        }
  }
```

18. What is the output of the following code fragment?

```
int num1 = 2;
int num2 = 10;
if(num1 < num2) {
      System.out.println("red");
}
if((num1+5) < num2) {
      System.out.println("white");
} else {
      System.out.println("blue");
}
System.out.println("yellow");
```

19. Given the class `Name` defined below, complete the method `sameInitials` that takes no input and returns `true` if the first and last name begin with the same character and `false` otherwise.

```
public class Name {

      private String first;
      private String last;

      public Name(String first, String last) {
            this.first = first;
            this.last = last;
      }
      public boolean sameInitials() {
            //YOUR CODE HERE




















      }
}
```

20. Implement the following method `printEvensBackward`. The method takes as input an `ArrayList` of `Integer` and prints all even numbers in the list from last to first. Given the list `[3, 5, 2, 6, 1, 6, 4]` your method would print 4, 6, 6, 2. Note that you will not do any sorting for this method and you will need to iterate over the list from the last element to the first element.

```
public void printEvensBackward(ArrayList<Integer> numbers) {
      //YOUR CODE HERE











}
```

21. Implement a method that will take as input a `String` and will print the contents of the `String` one character per line. The method will not return anything.
Given the input “cat”, the method will print the following:
c
a
t