Exam 3 – Offline Version

1. What does short circuit evaluation mean in the following code?

```
if (a < b && c != d)
```

   a. if a < b is false it evaluates c != d
   b. if a < b is false it doesn't evaluate c != d
   c. if c != d is false it evaluates a < b
   d. if c != d is false it doesn't evaluate a < b
   e. if a < b is true it doesn't evaluate c != d

2. Which if statement below tests if the variable letter holds the char value #?
   a. if (letter == #)
   b. if (letter == "#")
   c. if (letter == '#')
   d. if (letter >= '#')
   e. if (letter = '#')

3. Consider the following code:

```
int a [] = {2, 6, 8, 10, 12, 14, 16, 18};
int sum = 0;
for (int i = 0; i < a.length; i++) {
      if (a[i]%3 == 0)
            sum += a[i];
}
System.out.println(sum);
```

What is output?
   a. 20
   b. 26
   c. 28
   d. 36
   e. 38

4. What is output by the following code?

```
for (int i = 0; i < 5; i++)
      System.out.print(i + " ");
```

   a. 0 1 2 3 4 5
   b. 0 1 2 3 4
   c. 1 2 3 4
   d. 1 2 3 4 5
   e. 1 2 3 4 5 6

5. Which of the following would print the numbers, 32 54 76 98?

I.
```
for (int t = 32; t <= 100; t += 22)
      System.out.print(t + " ");
```

II.
```
int t = 10;
while (t < 90) {
      t += 22;
      System.out.print(t + " ");
}
```

III.
```
int t = 32;
while (t < 100) {
      t += 22;
      System.out.print(t + " ");
}
```

a. I only
b. II only
c. III only
d. I and II only
e. I, II and III

6. Assume that x and y are boolean variables and have been properly initialized.

```
(x && y) && !(x || y)
```

Which of the following best describes the result of evaluating the expression above?

a. Always true
b. Always false
c. true only when x is true and y is true
d. true only when x and y have the same value
e. true only when x and y have different values

7. Consider the following boolean statement:

```
!( x > y && w == z )
```

Which of the following will produce the same result?

a. x <= y && w == z
b. x >= y || w != z
c. x <= y || w != z
d. x <= y && w != z
e. x < y && w != z

8. What is 0 0 1 1 1 0 0 0 in base ten?

a. 54 b. 55 c. 56 d. 57 e. 58

9. What is 67 in binary?
   a. 0 1 0 0 0 0 1 0
   b. 0 1 0 0 0 0 1 1
   c. 0 1 0 0 0 1 1 0
   d. 0 1 0 0 0 1 1 1
   e. 0 1 0 0 1 1 1 0

10. Consider the following code:

```
String q = "power";
String r = "brown";
System.out.println(q.charAt( r.indexOf('n')));
```

What is output?

a. 2 b. 3 c. e d. r e. w

11. Consider the following code:

```
String words[] = {"avalanche", "budget", "cannot", "center", "outside",
"meaning", "clear", "furniture", "deep", "piccolo", "friendly",
"poison"};
int c = 0;
for (int i = 0; i < words.length; i++) {
      if (words[i].substring(3).indexOf('o') >= 0)
            c++;
}
System.out.println(c);
```

What is output?

a. 0
b. 1
c. 2
d. 3
e. None of the above

12. Consider the following code intended to implement a linear search:

```
int [] d = /* Assume array is correctly initialized */;
int num = /* Input from the keyboard */;
int found = -1;
for (int i = 0; i < d.length; i++) {
      if ( d[i] == num )
            /* Missing Code */
}
```

Which could be used to replace /* Missing Code */ so that the code works as intended?

a. found = 1;
b. found = i;
c. i = found;
d. i = -1;
e. d[i] = found;

13. Consider the following code:

```
String w = "Rapunzel";
for (int i = 0; i < w.length(); i++) {
      System.out.print(w.charAt(i) + " ");
      if (i%4 == 3)
            System.out.println();
}
```

What is output?

a. R a p u
   n z e l
b. R a p
   n z e l
c. R a p u n z e l
d. u l
e. R a p n z e

14. Consider the following code, intended to count the number of times that a given string appears in an array:

```
String n[] = /* Assume array initialized correctly */;
int c = 0;
String name = /* Input from the keyboard */;
for (int i = 0; i < n.length; i++) {
      if (/* Missing Code */ )
            c ++;
}
System.out.println(name + " appeared " + c  + " times in the array.");
```

What could be used to replace /* Missing Code */ so that the code works as intended?

a. n.equals(name)
b. name.equals(n)
c. n[i].equals(name)
d. i.equals(name)
e. ! n[i].equals(name)

15. The following is intended to count the number of times a score less than 70 is found in an array of test scores:

```
int [] d = /* Assume array is correctly initialized */;
int failing = 0;
for (int i = 0; i < d.length; i++) {
      if (/* Missing Code */)
            failing++;
}
System.out.println("Number of failing scores: " + failing);
```

Which of the following could replace /* Missing Code */ so that the code works as intended?

a. d != 70
b. failing < 70
c. d < 70
d. d[i] < 70
e. i < 70

16. What does the String method substring() do?

a. Returns a portion of the String.
b. Tests two String objects for equality.
c. Compares this String with a second String for greater than, equal to, or less than.
d. Returns the character at a certain location as a char value.
e. Returns the length of a String.

17. What does the following code do?

```
String w3 = "aardvark";
System.out.println(w3.charAt(w3.length() - 2));
```

a. Prints the second letter in the String.
b. Prints the second to last letter in the String.
c. Prints the last letter in the String.
d. Prints the first letter in the String.
e. Causes an index out of bounds error.

18. Consider the following code:

```
String major = "Computer Science";
```

What is returned by the method call major.charAt(2)?

a. 'C'   b. 'o'   c. 'm'   d. "C"   e. "Science"

19. Consider the following code:

```
    int list [] = new int [30];
```

The index of the first value is ______ and the last index is ______.

  a. 0, 29
  b. 0, 30
  c. 1, 29
  d. 1, 30
  e. 1, 31

20. When should you use a for loop instead of a while loop?

  a. When you are doing repeated calculations.
  b. When you do not know how many times a loop will repeat.
  c. When you have an exact starting and stopping point.
  d. When working with numbers.

When doing user input.