General Java Questions

1. (a) Consider the valid expression below. Circle what best describes the value of the expression.

   i. \( i \times i \geq i \) where \( i \) is a variable of type int

   \[ \begin{array}{ccc}
   \text{true} & \text{false} & \text{depends}
   \end{array} \]

   Answer: depends. If \( i \) is very large integer, then result of \( i \times i \) will be outside the integer range and it will appear as a negative number which is lesser than \( i \).

   ii. \( p \lor \neg p \) where \( p \) is a boolean variable

   \[ \begin{array}{ccc}
   \text{true} & \text{false} & \text{depends}
   \end{array} \]

   Answer: true. One of the operands is always true and the result of OR will be true.

(b) What is the value of variable \( s \) shown below:

   \[ \text{String s;} \]

   Answer: null. String is an Object type, and currently reference variable \( s \) is not referring to any created string object.

(c) When an object is created, it is stored in which section of the computer’s memory?

   Answer: Heap

(d) What is the outcome when the following code is executed?

   ```java
   int val = 5;
   for(int i = val; i > 0; i--){
     for(int j = 1; j <= i; j++){
       System.out.print(i);
     }
     System.out.println();
   }
   ```

   55555
   4444
   333
   22
   1
(e) The goal is to reverse the contents of the array. E.g. If the array contents are 1,2,3,4 then the reversed array is 4,3,2,1. The method below is written to reverse the original array. State whether the method is successful in delivering the end outcome. If not, how would you fix the code.

```java
public static void reverse(int [] data){
    for(int i = 0; i < data.length; i++){
        int temp = data[i];
        data[i] = data[data.length - 1 - i];
        data[data.length - i - 1] = temp;
    }
}
```

Not successful as even though it seems like we are reversing the array, after halfway point we reversing it back again, giving us back the original array. Solution:

```java
for(int i = 0; i < data.length/2;  i++){
```

2. See Reference Sheet - I for this question

A common task in information processing is to count the number of like items in a collection of items. For example, given a poll where people answer a question by indicating a number between 0 and 4 inclusive and an array which contains all the answers, we want to count the number of 0’s, the number of 1’s and so on. We can store the results in a results array with indices 0 through 4, where results[0] holds the number of people who answered 0, results[1] holds the number of people who answered 1, and so on.

Write a static method called `stats` in the Poll class that would perform the above described task. It should accept as input an array of integers and assume that its length is positive and all of its elements are greater than or equal to 0 (i.e. no need for error checking). It should return an array which contains a count of each integer entry in the range represented in the input array with no additional entries.

```java
// Method stats
public static int[] stats(int[] input) {
    // Find the max to determine the length
    int max = input[0];
    for (int i = 0; i < input.length; i++) {
        if (input[i] > max) {
            max = input[i];
        }
    }

    // Create a new array of size max + 1 due to indexing offset
    int[] result = new int[max + 1];

    // Poll answers
    for (int i = 0; i < input.length; i++) {
        result[input[i]]++;
    }

    return result;
}
```


Has-A Relationship

3. Consider the Point and Circle class interactions provided on Reference Sheet II.

(a) Write method called move(..) in the Point class that will move the point by dx in the x direction and dy in the y direction.

```java
  public void move(int dx, int dy) {
    x += dx;
    y += dy;
  }
```

(b) Write method called move(..) in Circle class that moves the Circle's center.

```java
  //Circle's center is Point object
  public void move(int dx, int dy){
    center.move(dx, dy);
  }
```

(c) Write another constructor in the Circle class that takes in three parameters as shown.

```java
  public Circle(int cx, int cy, int radius) {
    center = new Point(cx, cy);
    this.r = radius;
  }
```

(d) Write method liesWithin(..) in the Circle class that will return true if the input Point lies within or on the circle, otherwise returns false. Hint: Distance between Circle's center and input Point can be determined by Pythagorean theorem.

```java
  public boolean liesWithin(Point p1){
    int val = (p1.getX() - center.getX())*(p1.getX() - center.getX())
    + (p1.getY() - center.getY())*(p1.getY() - center.getY());
    return (val <= r*r);
  }
```