CS20B: Homework 1 (50 points)

prof. dr. Irma Ravkic

1 Questions (35 points)

1. (3 points) What is the difference between an object and a class? Give two examples.

2. (8 points) Some aspect of each of the following can be modeled with a graph structure. Describe, in each case, what the nodes would represent and what the edges would represent.

   (a) Trips available using a specific airline

   (b) Countries and their borders

   (c) A collection of research articles about data structures

   (d) Actors (research the "six degrees of Kevin Bacon")

   (e) The computers at a university

   (f) A labyrinth

   (g) The Web

   (h) Social media (like Facebook)

3. (5 points) *Research:* Find the Java library description of the ArrayList class and answer the following questions:

   (a) What class does it directly inherit from?

   (b) How many direct subclasses does it have?

   (c) How many methods does it implement?

   (d) How many methods does it inherit?

   (e) If we invoke the *toString* method on an object of class ArrayList, which class's *toString* method will be used?

4. (10 points) Describe and specify the order of growth of each of the following code sections, using big-O notation:

   (a)

```
1  int count = 0;
   for (int i = 1; i <= N; i++)
3     count ++;
```

   (b)

```
        int count = 0;
        for (int i = 1; i <= N; i++)
             for(int j = 1; j <= N; j++)
                 count ++;
```

(c)

```
        int count = 0;
        for (int i = 1; i <= N; i++)
            cout ++;
        for (int j = 1; j <= N; j++)
            count ++;
```

(d)

```
        int count = 0;
        for (int i = 1; i <= N/2; i++)
           for(int i = 1; i <= N/2; i++)
                 count ++;
```

(e)

```
public static String reverse(String s) {
    int n = s.length();
    char[] a = new char[n];
    for (int i = 0; i < n; i++)
       a[i] = s.charAt(n-i-1);
    String reverse = new String(a);
    return reverse;
}
```

5. (6 points) Design a set of at least three classes related by inheritance from the world of
   - Banking
   - Gaming
   - Travel
6. (3 points) Describe the order of growth of each of the following functions using O notation
   - $N^2 + 3N$
   - $3N^2 + N$
   - $(N * (N - 1))/2$

2 Programming exercises (15 points)

1. (5 points) Create a linear time complexity method called *sum* that returns the sum of the integers between 1 and n. Create a second method that performs the same function but with a lower order of growth.

2. (10 points) Create a class that models a standard pair of dice in the following way:

   (a) Create a class called *PairOfDice* in Java. Objects of this class represent a single pair of six-sided dice. The only attributes of such an object are the face values of the dice. Provide a constructor. Provide a *roll* method that simulates rolling the dice (you need to use a random number generator). Provide a *value* method that returns the sum of the face values of the dice. Provide a *toString* method that returns a nicely formatted string representing the pair of dice, for example "5 : 3 = 8". Finally, create a "test driver" class containing the *main* method that demonstrates that your *PairOfDice* class performs correctly.

   (b) Now imagine that we also want to include an unfair or biased dice in our modeling. This means that rolling the dice doesn't have equal probabilities for each value (in a fair dice each value has the equal probability of 1/6 to be rolled). You don't need to implement the ubiased roll. The question is how would you reorganize your program to include this *BiasedDicePair* class? Add that new class to your project.