Assignment 1 - Requirements Elicitation and Documentation

Exercise 1 - Design and Documentation

You are given the following classes:

```java
public class Node{
  public Node(){...}
  ...
}

public class Edge{
  Node from;
  Node to;
  int distance;

  public Edge(Node from, Node to, int distance){...}
}

public class STGraph{
  List<Edge> edges;
  Node source;
  Node target;

  public STGraph(Node source, Node target){
    this.source = source;
    this.target = target;
    this.edges = new ArrayList<Edge>();
  }

  public void setST(Node source, Node target){
    this.source = source;
    this.target = target;
  }

  public void addEdge(Edge e){
    edges.add(e);
  }

  public List<Node> shortestPath(){...}
}
```

1. Under what use cases would the above design decisions (passing the source and the target as arguments to the STGraph's constructor and using a setter method for modifying them afterwards) make sense?

2. Does the shortest path always exist? What information should be provided in the client documentation of the method shortestPath with respect to the return value?

3. Since computing the shortest path in a graph is an expensive operation, we would like to avoid re-computing it unless it is necessary.

    (a) Come up with a design to accomplish this.

    (b) How does this design influence the client-visible documentation for the class STGraph?

    (c) What contracts should be provided to document your design for the implementer/maintainer?

    (d) Are there alternative designs that achieve comparable results?

Exercise 2 - Design

You have seen different list implementations in the lecture, one of which uses reference counting to reduce the need for expensive cloning operations:

```
class List<E>{
  E[] elems;
  int len;
  boolean shared;

  List(E[] e, int l){
    elems = e;
    len = l;
    shared = true;
  }

  void set(int index, E e){
    if(shared){
      elems = elems.clone();
      shared = false;
    }
    elems[index] = e;
  }

  List<E> take(){
    shared = true;
    return new List<E>(elems, len - 1);
  }
}
```

1. Come up with a scenario in which this implementation performs unnecessary cloning operations, when the set method is called.

2. How could you modify the implementation to eliminate this inefficiency?

3. Is your proposed solution efficient in the case in which unused objects are removed by the garbage collector? If yes, explain why. If not, explain how could you modify it to also efficiently handle this scenario.

Exercise 3 - Requirements Elicitation

You have been given the role of specifying the requirements document for an online flower shop.

> The client has been running this shop for many years and has become friends with both his customers and his suppliers. However, the client would like to streamline the process of delivering the flowers and the purchasing of bulk flowers. He currently has one messenger for the delivery of flowers and only three suppliers of bulk flowers that he receives from the Netherlands. One of his biggest concerns in "modernizing" his flower shop is that he may loose the personal touch offered by his shop.

> His current customers appreciate his touch for determining the right flowers for the right moments in their lives. The frequent customers are normally given an in-house account, which they may pay at the end of the month. New customers must pay either at delivery or by credit card over the phone if the flowers are a gift.

> He has found out that most of the mistakes in the process of ordering flowers happen with new or occasional customers. They have to provide an address and possibly a credit card every time they call for a delivery. Furthermore, if there is a discrepancy in the address or credit card, the flower shop owner noticed that the flowers are usually delivered too late and the client has already found another shop to purchase them from. Or, that the client did not exist at all!

> He is also interested in determining which flowers are popular during the major events of the year. One of the issues in working with his suppliers is to determine which flowers to get, and the right amount for each type of flowers. Waste of flowers is very expensive and he would like to use this "modernization" as a way of helping cut down on this waste.

> He is currently more interested in modernizing his relationship with his clients and wishes to see how this works before risking a modernization with his suppliers.

> He currently has an internet connection at the flower shop attached to his older PC. His shop already has a point of sales system for in store sales. The interface for the online store will be handled by his daughter who is currently taking lessons in web design.

It is your responsibility to elicit the requirements and to write a document capturing them to begin the discussions with the customer. Start by determining the actors of the system. Write down the scenarios of the system (incl. exceptional one) and determine any open issues. Once you have determined the open issues, discuss these with the flower shop owner (your assistant). Think about how you could generalize the scenarios into use cases. Afterwards, identify the non-functional requirements of the system.