ICS-202 Lab-10 Report

Mohammed Busaleh - 202158210

Task 1:

Code:

```java
public class Entry<T> {
    private T dataObject;
    private String status;

    public Entry(){
        status = "E";
    }

    public boolean setData(T dataObject){
        boolean taskDone = false;

        if(status.equals("E") || status.equals("D")) {
            this.dataObject = dataObject;

            taskDone = true;
        }

        return taskDone;
    }

    public void setStatus(String status) {
        this.status = status;
    }

    public String getStatus() {
        return status;
    }
```

```java
public T getDataObject() {
    return dataObject;
}

@Override
public int hashCode() {
    return super.hashCode();
}

@Override
public String toString() {
    Object[] entities = {dataObject, status};
    return Arrays.toString(entities);
}
```

Task 2:

Code:

```java
public class HashTable<T> {
    private Entry[] data;

    public HashTable(int size){
        data = new Entry[size];

        for(int i = 0; i < data.length; i++)
            data[i] = new Entry<>();
    }

    private int hashCodeGenerate(int number){
        return number%data.length;
    }

    public boolean insert(T dataObject){
        boolean taskDone;
        int num = (int)dataObject;
        int index = hashCodeGenerate(num);

        while(index < data.length
                && data[index].getStatus().equals("O")){
            num++;
            index = hashCodeGenerate(num);
        }


        try{
            data[index].setData(dataObject);
            data[index].setStatus("O");
            taskDone = true;
        }
        catch(Exception indexOutOfBounds){
            taskDone = false;
        }


        return taskDone;
    }
```

```java
public int findNextAvailableSlot(int currentslot){
    int nextAvailable = hashCodeGenerate(currentslot);

    while(nextAvailable < data.length
            && data[nextAvailable].getStatus().equals("O")){
        currentslot++;
        nextAvailable = hashCodeGenerate(currentslot);
    }

    if(nextAvailable>=data.length)
        nextAvailable = -1;

    return nextAvailable;
}

public boolean delete(T dataObject){
    int idx = find(dataObject);
    if(idx == -1){return false;}

    data[idx].setStatus("D");
    System.out.println("successfully deleted");
    return true;
}
```

```java
public int find(T dataObject){
    int num = (int)dataObject, idx = hashCodeGenerate(num);
    String st = data[idx].getStatus();

    while(idx < data.length && !st.equals("E")){
        if(st.equals("O")
                && data[idx].getDataObject().equals(dataObject)){
            System.out.println(dataObject + " was found at " + idx);
            return idx;
        }

        num++;
        idx = hashCodeGenerate(num);
        st = data[idx].getStatus();
    }

    System.out.println(dataObject + " is not found");
    return -1;
}

@Override
public String toString() {
    System.out.println("HASHTABLE");
    for(int i = 0; i < data.length; i++)
        System.out.println(i + ":  " + data[i]);

    return "\n";
}
```

Task 3:

Code:

```
public static void main(String[] args) {
    HashTable<Integer> myHT = new HashTable<>(13);
    myHT.insert(18);
    myHT.insert(26);
    myHT.insert(35);
    myHT.insert(9);
    System.out.println("After insertion:");
    System.out.println(myHT.toString());

    myHT.find(15);
    myHT.find(48);
    myHT.delete(35);
    myHT.find(9);

    myHT.insert(64);
    myHT.insert(47);
    System.out.println("After deletion & insertion:");
    System.out.println(myHT.toString());

    myHT.find(35);
}
```

Output:

```
After insertion:
HASHTABLE
0:  [26, 0]
1:  [null, E]
2:  [null, E]
3:  [null, E]
4:  [null, E]
5:  [18, 0]
6:  [null, E]
7:  [null, E]
8:  [null, E]
9:  [35, 0]
10:  [9, 0]
11:  [null, E]
12:  [null, E]


15 is not found
48 is not found
35 was found at 9
successfully deleted
```

```
9 was found at 10
After deletion & insertion:
HASHTABLE
0:  [26, 0]
1:  [null, E]
2:  [null, E]
3:  [null, E]
4:  [null, E]
5:  [18, 0]
6:  [null, E]
7:  [null, E]
8:  [47, 0]
9:  [35, D]
10:  [9, 0]
11:  [null, E]
12:  [64, 0]


35 is not found
```