**Assignment 10**

Done By: Nayan Man Singh Pradhan

**Problem 10.1 - Hash Tables**

**a)**

**Given,**

**We know, double hash uses the hash function:**

**First, we insert 3:**

Here, the value of as it is the initial iteration for this key. Therefore,

Now, we check position 3 in our hash table. Since position 3 is empty, we insert key 3 in position 3.

**Then, we insert 10:**

The value of as it is the initial iteration for this key. Therefore,

We check position 0 in our hash table. Since position 0 is empty, we insert key 10 in position 0.

**Then, we insert 2:**

The value of as it is the initial iteration for this key. Therefore,

We check position 2 in our hash table. Since position 2 is empty, we inert key 2 in position 2.

**Finally, we insert 4:**

The value of as it is the initial iteration for this key. Therefore,

Finally, we check position 4 in our hash table. Since position 4 is empty, we insert key 4 in position 4.

Therefore, there are no collisions while inserting the given data.

**b)**

Hash Table has been implemented in *hashTable.cpp* (execute make to run).

I selected linear probing with , where as I am only inserting a small number of keys (5 keys) to test the program, and hence, my algorithm will not suffer from a large amount of primary clustering.

**Problem 10.2 – Greedy Algorithm**

**a)**

The greedy algorithm in the activity-selection problem may fail at producing a globally optimal solution. An example of this scenario is given below:

|  |  |  |  |
| --- | --- | --- | --- |
| **(index)** | 1 | 2 | 3 |
| **(start time)** | 1 | 2 | 4 |
| **(end time)** | 4 | 4 | 8 |

The greedy algorithm always chooses the solution that is locally optimal. In this case, the greedy algorithm chooses the index of the first pair **(after the data is sorted in an ascending order according to end time)**. T

In the table,

T