Jolie Khaled - 900211816

Assignment 4

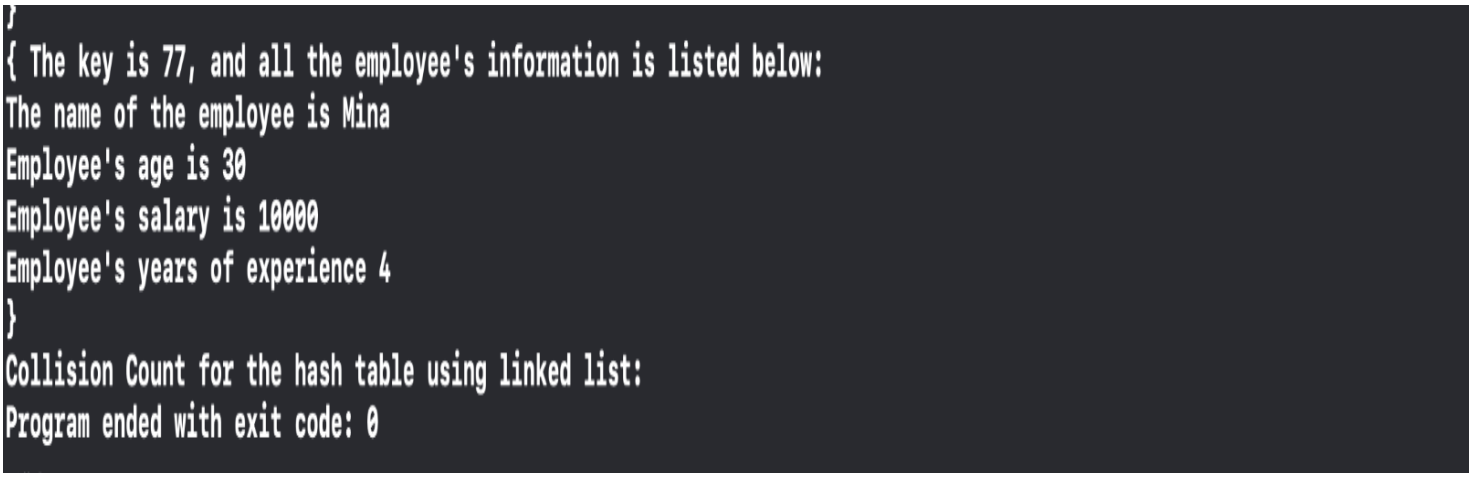
```
This is the output of all Employees with their information using dynamic array:

{ The key is 77 , and all the employee's information is listed below:
The name of the employee is Mariam
Employee's age is 32
Employee's salary is 8000
Employee's years of experience 2
}
{ The key is 65 , and all the employee's information is listed below:
The name of the employee is Ayman
Employee's age is 33
Employee's salary is 4000
Employee's years of experience 8
}
{ The key is 65 , and all the employee's information is listed below:
The name of the employee is Aya
Employee's age is 26
Employee's salary is 6000
Employee's years of experience 3
}
{ The key is 65 , and all the employee's information is listed below:
The name of the employee is Abdallah
Employee's age is 29
Employee's salary is 7000
Employee's years of experience 4
}
{ The key is 82 , and all the employee's information is listed below:
The name of the employee is Roshdy
Employee's age is 28
Employee's salary is 9000
Employee's years of experience 3
}
{ The key is 70 , and all the employee's information is listed below:
The name of the employee is Fawzy
Employee's age is 45
Employee's salary is 5000
Employee's years of experience 8
}
{ The key is 70 , and all the employee's information is listed below:
The name of the employee is Fatma
Employee's age is 21
Employee's salary is 3000
Employee's years of experience 1
}
```

```
{ The key is 89 , and all the employee's information is listed below:
The name of the employee is Yara
Employee's age is 19
Employee's salary is 2000
Employee's years of experience 0
}
{ The key is 77 , and all the employee's information is listed below:
The name of the employee is Mina
Employee's age is 30
Employee's salary is 10000
Employee's years of experience 4
}
Collision Count for the hash table using dynamic array:
0: 3
5: 1
12: 1
--------------------------------------------------

This is the output of all Employees with their information using Linked list:

{ The key is 65, and all the employee's information is listed below:
The name of the employee is Ayman
Employee's age is 33
Employee's salary is 4000
Employee's years of experience 8
}
{ The key is 82, and all the employee's information is listed below:
The name of the employee is Roshdy
Employee's age is 28
Employee's salary is 9000
Employee's years of experience 3
}
{ The key is 70, and all the employee's information is listed below:
The name of the employee is Fawzy
Employee's age is 45
Employee's salary is 5000
Employee's years of experience 8
}
{ The key is 89, and all the employee's information is listed below:
The name of the employee is Yara
Employee's age is 19
Employee's salary is 2000
Employee's years of experience 0
}
```



These screenshots are the output of a code that aims to print all the employee's information once using a dynamic array and another time using a linked list and we should calculate the collision rate if any duplicates have happened. To implement the hash table code, of course I have used many references and their links are listed below, and I decided that my main hash function will be calculating the key % capacity. I initialized that the capacity is 13, so for every key, it will use the remainder after dividing it by the capacity. So the main purpose of the hash function is to distribute the key throughout the hash table based on this equation. I used this function as I thought that it would help to search for the data more quickly and efficiently. In the instructions I learned that we could just insert the names of the employees to be easier but I worked on printing not just the names but also all the employee's information including the salary, age and years of experience.

As we can see in the output the screenshots are divided into 2 main parts. The first one is printing the employee's info using a dynamic array. So for every employee, it states the key in the hash table, the name, the salary, the age and the years of experience. And at the end of the first part, the collision rate was calculated, so at index 0, there were 3 collisions, at 5, there was one collision and at 12 there was one collision too, and we thankfully solved these collisions using linear probing. The second part is meant to do the same thing as the first but using a linked list instead of a dynamic array. It showed the key, name, age, years of experience for every employee. If any collisions have happened, I have handled this by implementing the separate chaining, but in the linked list, there wasn't any collison that has occurred as we can see in the last line in the output that there was no collision. We can conclude in this example that the hash table using the linked list is much better and efficient than the dynamic array, as in the

linked list there was no collision, on the contrary of the dynamic array that has many collisions.

References: C++ program for hashing with chaining - GeeksforGeekshttps://www.geeksforgeeks.org › c-program-hashing-ch...

Implementing own Hash Table with Open Addressing Linear ...https://www.geeksforgeeks.org › implementing-hash-ta…

C++ Program to Implement Hash Tables with ... - Tutorialspointhttps://www.tutorialspoint.com › cplusplus-program-to-...