Roll No.: SYCOA68

Name : Vedant Ghumade

Assignment 1

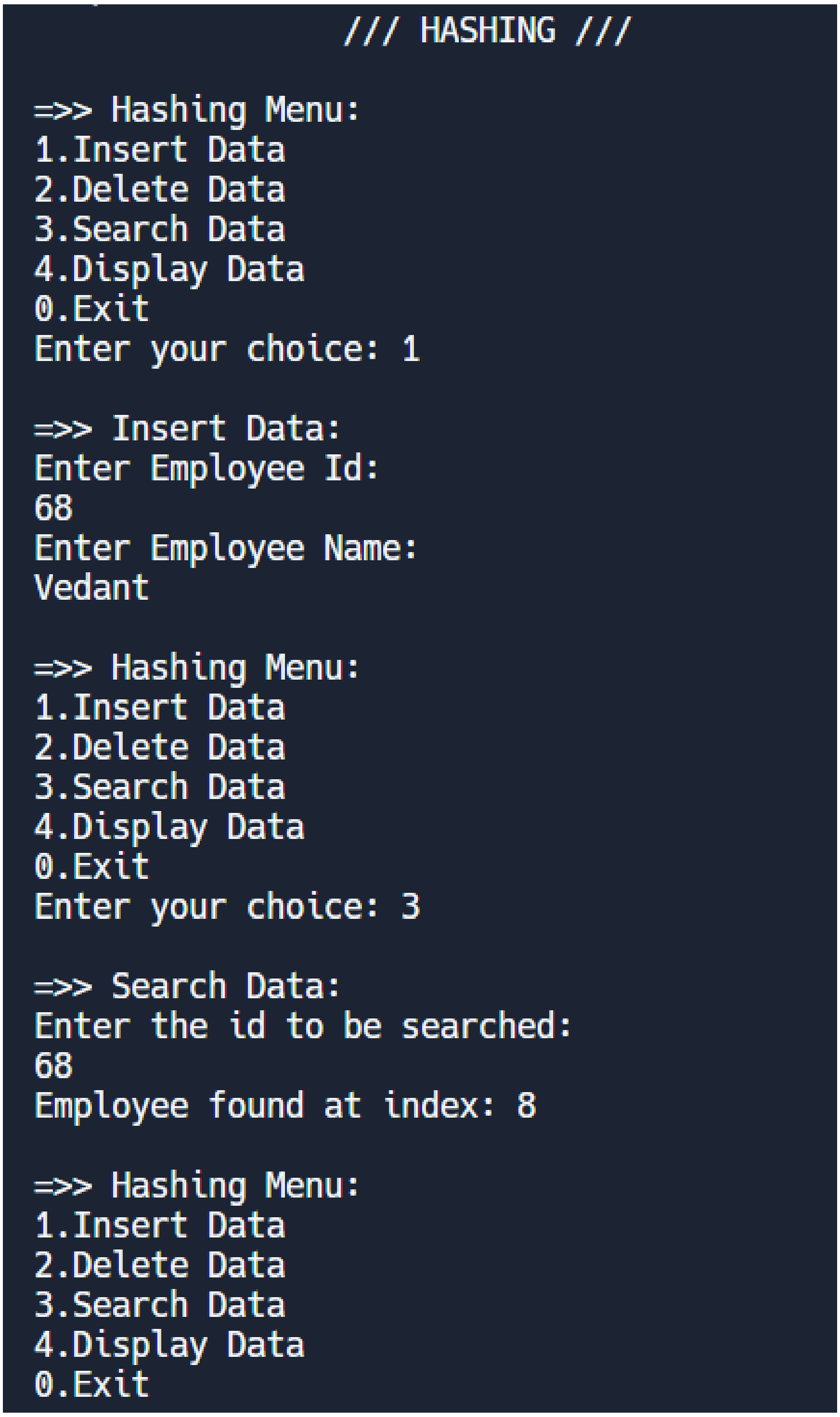
Consider an employee database of N employees. Make use of a hash table implementation to quickly look up the employee’s id number.

Code:

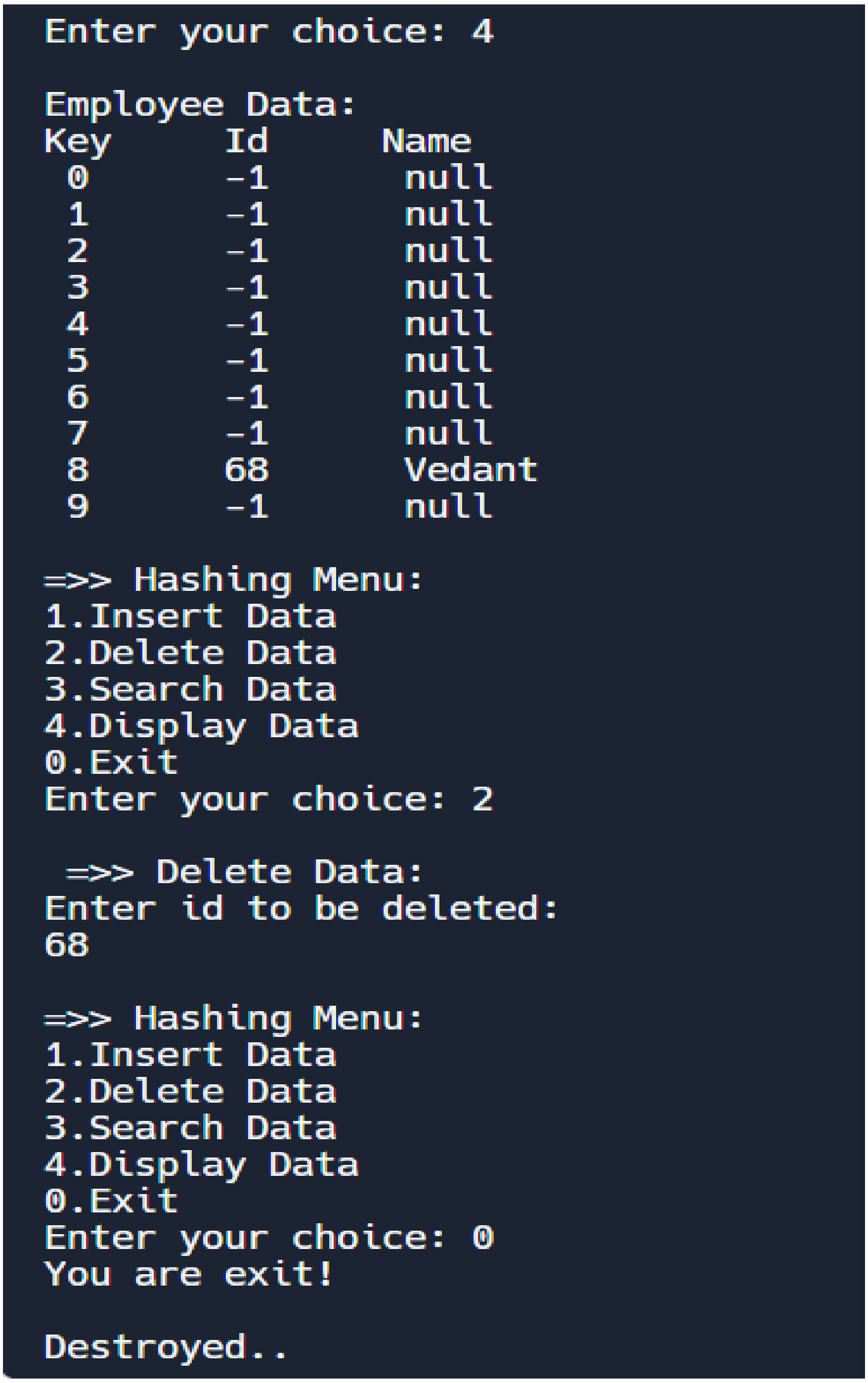
```
#include <iostream>

using namespace std;

struct Employee

{

   int id;

   string name;

   int key;

};

class Hashtable

{

private:

   Employee Table[10];


public:

   void insertion();

   void deletion();
```

```
    void searching();

    void display();

    int hashfunction(int key);

    Hashtable();

    ~Hashtable();

};

void Hashtable ::deletion()

{

    int x, i;

    cout << "\n =>> Delete Data: " << endl;

    cout << "Enter id to be deleted:" << endl;

    cin >> x;

    for (i = 0; i < 10; i++)

    {

        if (x == Table[i].id)

        {

            Table[i].id = -1;

            Table[i].key = -1;

            Table[i].name = "null";

        }

    }

}

void Hashtable ::searching()

{

    int iKey, j;

    cout << "\n=>> Search Data: " << endl;

    cout << "Enter the id to be searched: " << endl;
    cin >> iKey;
   for (j = 0; j < 10; j++)
    {
        if (iKey == Table[j].id)
        {
            cout << "Employee found at index: " << j << endl;
        }

       }

}
int Hashtable ::hashfunction(int key)
{
    return key % 10;
}
Hashtable ::Hashtable()
{
    for (int i = 0; i < 10; i++)
    {
        Table[i].id = -1;
        Table[i].key = -1;
        Table[i].name = "null";
    }

}
Hashtable ::~Hashtable()
{
    cout << "\nDestroyed..\n"
         << endl;
}
void Hashtable ::insertion()
{
    int id, flag = 0;
    string name;
    cout << "\n=>> Insert Data: " << endl;
    cout << "Enter Employee Id: " << endl;
    cin >> id;
    cout << "Enter Employee Name: " << endl;
    cin >> name;
    int index = hashfunction(id);
    int c = index;
   while (Table[index].key != -1 && flag == 10)
    {
        index = (index + 1) % 10;
        flag++;
    }
    Table[index].id = id;
    Table[index].name = name;
    Table[index].key = c;
}
void Hashtable ::display()
{
    cout << "\nEmployee Data: " << endl;
    cout << "Key     Id     Name " << endl;
    int j = 0;
    for (j = 0; j < 10; j++)
    {
        cout << " " << j << "   " <<"\t"<< Table[j].id << "   " <<"\t"<< Table[j].name << endl;
    }
}

int main()
{
    int ch;
    Hashtable h;
    cout<<"\t\t\t\t/// HASHING ///"<<endl;
    do
    {
        cout << endl;
        cout << "=>> Hashing Menu:" << endl;
        cout << "1.Insert Data" << endl;
        cout << "2.Delete Data" << endl;
        cout << "3.Search Data" << endl;
        cout << "4.Display Data" << endl;
        cout << "0.Exit" << endl;

        cout << "Enter your choice: ";
        cin >> ch;

        switch (ch)
        {
        case 1:
            h.insertion();
            break;

        case 2:
            h.deletion();
            break;

        case 3:
            h.searching();
            break;

        case 4:
            h.display();
            break;

        case 0:
            goto end;
            break;

        default:
            cout << "Incorrect Choice!" << endl;
        }
    } while (1);

end:
    cout << "You are exit!" << endl;

    return 0;

}
```

Output :



```
                  /// HASHING ///

=>> Hashing Menu:
1.Insert Data
2.Delete Data
3.Search Data
4.Display Data
0.Exit
Enter your choice: 1

=>> Insert Data:
Enter Employee Id:
68
Enter Employee Name:
Vedant

=>> Hashing Menu:
1.Insert Data
2.Delete Data
3.Search Data
4.Display Data
0.Exit
Enter your choice: 3

=>> Search Data:
Enter the id to be searched:
68
Employee found at index: 8

=>> Hashing Menu:
1.Insert Data
2.Delete Data
3.Search Data
4.Display Data
0.Exit
```



```
Enter your choice: 4

Employee Data:
Key     Id      Name
 0      -1       null
 1      -1       null
 2      -1       null
 3      -1       null
 4      -1       null
 5      -1       null
 6      -1       null
 7      -1       null
 8      68       Vedant
 9      -1       null

=>> Hashing Menu:
1.Insert Data
2.Delete Data
3.Search Data
4.Display Data
0.Exit
Enter your choice: 2

 =>> Delete Data:
Enter id to be deleted:
68

=>> Hashing Menu:
1.Insert Data
2.Delete Data
3.Search Data
4.Display Data
0.Exit
Enter your choice: 0
You are exit!

Destroyed..
```