

```

//=====
// Name      : Directory2018.cpp
// Author    :
// Version    : Simple Linear Probing
// Copyright  : Your copyright notice
// Description : Hello World in C++, Ansi-style
//=====

#include <iostream>
#include <iomanip>
using namespace std;

class RECORD          //class for person record
{
    string name;
    string mob;

public:
    RECORD()
    {
        name=" ";
        mob="0";
    }
    friend class DIRECTORY;
    friend int main();
};

class DIRECTORY        //class for directory
{
    RECORD HT[26];
public:
    void Insert(RECORD P);
    int search(string);
    void display_HT();
    friend int main();
};

void DIRECTORY::Insert(RECORD p)
{
    int hl,j,i;
    hl = int(p.name[0])%26;
    if(HT[hl].name==" ")           //Hashed location is empty.
    {
        HT[hl]=p;
    }
    else
    {
        for(i=1;i<26;i++)
        {
            j=(hl+i)%26;
            if(HT[j].name==" ")    // store the new record
            {
                HT[j]=p;
                break;
            }
        }
    }
}

```

```

        }
    }
    if(i==26)
        cout<<"\n Directory is full";

}

}

void DIRECTORY::display_HT()
{
    cout<<"\n                                HASH TABLE                                ";
    cout<<"\n-----";
    cout<<"\n| Location |      Names      |      Mobile No.      |";
    cout<<"\n-----";
    for(int i=0;i<26;i++)
    {
        if(HT[i].name==" ")
            cout<<"\n|      "<<setw(2)<<i<<"      |      --      |      --      |      ";
        else
            cout<<"\n|      "<<setw(2)<<i<<"      | "<<setw(10)<<left<<HT[i].name<<"
| "<<setw(10)<<right<<HT[i].mob<<"      |      ";
            cout<<"\n-----";
    }
}

int DIRECTORY::search(string s)
{
    int hl,i=0,j;
    j=hl=int(s[0])%26;

    while(i<26)
    {
        if(HT[j].name==s)        //Check home location contains desired record or
not
        {
            return j;
        }
        else
        {
            i++;
            j=(hl+i)%26;
        }
    }
    return -1;                    //Otherwise record not present
}

int main()
{
    DIRECTORY d;
    int ch;
    char c;
    string s;
    do
    {

```

```

cout<<"\n ----- MENU -----";
cout<<"\n 1. INSERT RECORD          ";
cout<<"\n 2. SEARCH RECORD          ";
cout<<"\n 3. DISPLAY HASH TABLE      ";
cout<<"\n 4. Exit                      ";
cout<<"\n -----";
cout<<"\n Enter your choice=>";
cin>>ch;
switch(ch)
{
    case 1:
        do
        {
            RECORD P;
            cout<<"\n Enter the name to insert=>";
            cin>>P.name;
            if((ch=d.search(P.name))!=-1)
            {
                cout<<"\n Enter the mobile no=>";
                cin>>P.mob;
                d.Insert(P);
            }
            else
            {
                cout<<"\n Record already present";
            }
            cout<<"\n Do you want to continue(y/n)=";
            cin>>c;
            }while(c=='y' || c=='Y');
            break;
    case 2:
        cout<<"\n Enter the name to search=>";
        cin>>s;
        ch=d.search(s);
        if(ch==-1)
        {
            cout<<"\n Name not present in directory";
        }
        else
        {
            cout<<"\n-----";
            cout<<"\n|      Name      |  Mobile No.  |";
            cout<<"\n-----";
            cout<<"\n|  "<<setw(10)<<left<<d.HT[ch].name<<"  |
"<<setw(10)<<right<<d.HT[ch].mob<<"  |";
            cout<<"\n-----";
        }
        break;
    case 3:
        d.display_HT();
        break;
    case 4:
        cout<<"\n Exiting.....";
        break;

    default:

```

