```
Script started on Fri 01 Apr 2016 01:08:29 AM CDT
\033]0;g_butler4@mars:~/csc122/hideNseek\007[g_butler4@mars hideNseek]$ pwd
/home/students/g_butler4/csc122/hideNseek
\033]0;g_butler4@mars:~/csc122/hideNseek\007[g_butler4@mars hideNseek]$ cat info.txt
Name:Gary Butler
For:CSC122-002
Project:Hide 'n' Go Seek
Levels Attempted:4.5
Description:This program reads in a list of names from a file and finds every instance
 of the given name.\033]0;g_butler4@mars:~/csc122/hideNseek\007[g_butler4@mars hideNse
ek]$ cat hns.cpp
#include <iostream>
#include <string>
#include <fstream>
#include <vector>

using namespace std;


int main()
{
        vector<int> pos;
        vector<int>::iterator i_p=pos.begin();

        fstream filei;
        string fn;
        string temp, find;
        bool found = false;
        bool again = true;
        bool p = false;
        char c_temp;
        int lines = 0;

        cout << "Enter name of file: ";
        getline(cin >> ws, fn);
        if (fn.find_first_of('.') == -1)
        {
                fn.append(".txt");
        }
        filei.open(fn.c_str());

        if (filei.fail())
        {
                while (filei.fail())
                {
                        filei.close();
                        filei.clear();
                        cout << "\nTry again";
                        cout << "\nEnter name of file: ";
                        getline(cin >> ws, fn);
                        if (fn.find_first_of('.') == -1)
                        {
                                fn.append(".txt");
                        }

                        filei.open(fn.c_str());
                }
        }
        else
        {
                cout << "Opening: " << fn<<"\n";
        }
        while (again == true)
        {
                cout << "Enter name to be searched for: ";
                getline(cin, find);
                c_temp = find[0];
                filei.peek();
                while (!(filei.eof()))
                {
                        getline(filei, temp);
                        lines++;
                        if ((temp[0] == c_temp) || (temp[0] == (toupper(c_temp))))
                        {
                                found = true;

                                pos.push_back(lines);//add line number

                                for (int a = 1; a < find.length(); a++)
                                {
                                        c_temp = find[a];
                                        if ((temp[a] != c_temp) && (temp[a] != (touppe
r(c_temp))))
                                        {
                                                if (p == false)
                                                {
                                                        cout << "Partal match on line:
 " << lines << "\n";
                                                        cout << "Did you mean: " << te
mp << "?\n";
                                                        p = true;
                                                }
                                                found = false;
                                        }
                                }
                                filei.peek();
                                c_temp = find[0];
                        }


                }

                if (found == false)
                {
                        cout << "name not found";
                }
                else
                {
                        cout << "Name found on line(s): ";
                        for (i_p = pos.begin(); i_p != pos.end(); i_p++)
                        {
                                cout << *i_p;
                                if ((i_p+1) != pos.end())
                                {
                                        cout << " & ";
                                }

                        }
                }
                cout << "\nAgain? (Y/N)";
                cin >> fn;
                if ((fn == "Y") || (fn == "y"))
                {
                        again = true;
                        found = false;
                        p = false;
                        lines=0;
                        pos.clear();
```

```
                        cin.clear();
                        filei.clear();
                        filei.seekg(0,ios::beg);
                        getline(cin,find);
                }
                else
                {
                        filei.close();
                        again = false;
                }
        }
        return 0;
}\033]0;g_butler4@mars:~/csc122/hideNseek\007[g_butler4@mars hideNseek]$ CPP\033[K033[
K033[K033[K\007\007./hns.out
Enter name of file: names
Opening: names.txt
Enter name to be searched for: tammy henry
Name found on line(s): 3 & 4
Again? (Y/N)y
Enter name to be searched for: Tammy Henry
Name found on line(s): 3 & 4
Again? (Y/N)y
Enter name to be searched for: tammy jones
Partal match on line: 3
Did you mean: Tammy Henry?
name not found
Again? (Y/N)y
Enter name to be searched for: Jason James
Name found on line(s): 1
Again? (Y/N)n
\033]0;g_butler4@mars:~/csc122/hideNseek\007[g_butler4@mars hideNseek]$ cat tpq.txt
1.I used a vector to push back every line number the name is found on.
2.I used a while loop to check for the end of the file, primed by a .peek().
3.If the name is not found, I tell the user (they have the right to know)
4.In a sense, my code will handle more than one name per line. However, it will only f
ind the first name of the line and disregard the rest of data. Otherwise, it will igno
re the line entirely.\033]0;g_butler4@mars:~/csc122/hideNseek\007[g_butler4@mars hideN
seek]$ exit
exit

Script done on Fri 01 Apr 2016 01:09:59 AM CDT
```