File Management System

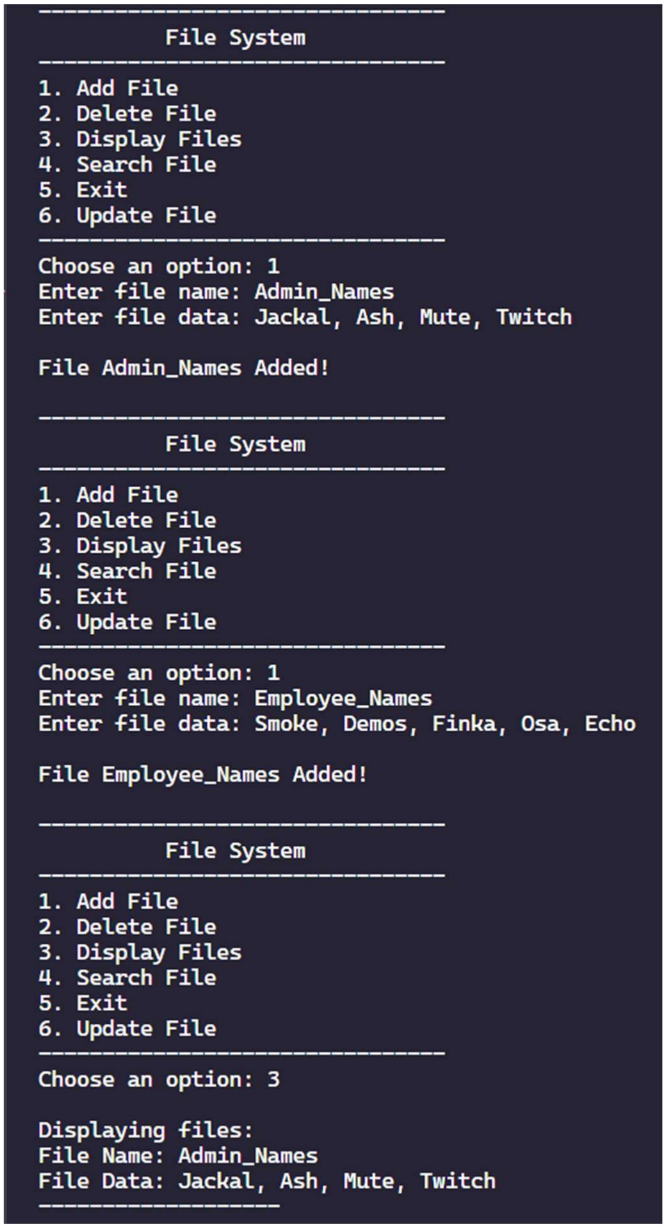
```
#include <iostream>

using namespace std;

struct Node
{
    string fname;
    string fdata;
    Node *next;

    Node(const string &nm, const string &d) : fname(nm), fdata(d), next(NULL){};
};

class FileSys
{

public:
    Node *head = NULL;

    void addFile(const string &nm, const string &d)
    {
        Node *newNode = new Node(nm, d);

        if (!head)
        {
            head = newNode;
    }
    else
    {
        Node *save = head;

        while (save->next)
        {
            save = save->next;
        }

        save->next = newNode;
    }

    cout << "\nFile " << nm << " Added!" << endl
         << endl;
}

void deleteFile(const string &nm)
{
    if (head == NULL)
    {
        cout << "Folder is Empty!" << endl;
    }
    else if (head->fname == nm)
    {
        head = head->next;
        cout << "File " << nm << " deleted!" << endl;
        return;
    }
    Node *save = head;

    while (save->next && save->next->fname != nm)
    {
        save = save->next;
    }
    if (save->next)
    {
        save->next = save->next->next;
        cout << "\nFile " << nm << " deleted!" << endl;
    }
    else
    {
        cout << "\nFile not found!";
    }
}

void searchFile(const string &nm)
{
    Node *current = head;

    while (current != NULL)
    {
        if (current->fname == nm)
        {
            cout << "\nFile found!";
            cout << "\nFile Name: " << current->fname;
            cout << "\nFile Data: " << current->fdata;
            return;
        }
        current = current->next;
    }

    cout << "\nFile not found!" << endl;
}

void displayFile()
{
    if (head == NULL)
    {
        cout << "\nFolder is Empty!" << endl;
        ;
        return;
    }

    Node *current = head;

    cout << "\nDisplaying files:" << endl;
    while (current != NULL)
    {
        cout << "File Name: " << current->fname << endl;
        cout << "File Data: " << current->fdata << endl;
        cout << "-------------------" << endl
             << endl;

        current = current->next;
    }
```

```
}

void updateFiledata(const string &nm, const string &newData)
{
    Node *current = head;

    while (current != NULL)
    {
        if (current->fname == nm)
        {
            current->fdata = newData;
            cout << "\nFile '" << nm << "' updated!" << endl;
            return;
        }
        current = current->next;
    }
    cout << "\nFile '" << nm << "' not found!";
}

void updateFilename(const string &nm, const string &newData)
{
    Node *current = head;

    while (current != NULL)
    {
        if (current->fname == nm)
        {
            current->fname = newData;
            cout << "\nFile '" << nm << "' renamed!" << endl;
```

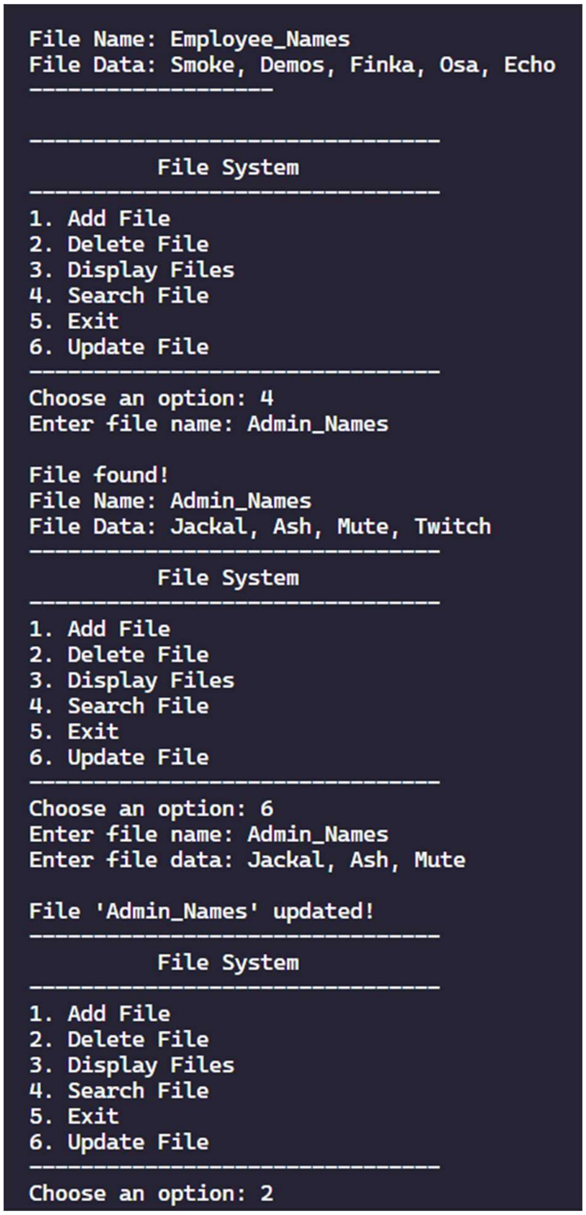
```
                return;
            }
            current = current->next;
        }
        cout << "\nFile '" << nm << "' not found!";
    }
};

int main()
{
    FileSys fs;
    string nm, d;
    int ch;

    do
    {
        cout << "--------------------------------" << endl;
        cout << "          File System           " << endl;
        cout << "--------------------------------" << endl;
        cout << "1. Add File" << endl;
        cout << "2. Delete File" << endl;
        cout << "3. Display Files" << endl;
        cout << "4. Search File" << endl;
        cout << "5. Update File" << endl;
        cout << "6. Rename File" << endl;
        cout << "7. Exit" << endl;
        cout << "--------------------------------" << endl;

        cout << "Choose an option: ";
```

```
cin >> ch;

switch (ch)
{
case 1:
    cout << "Enter file name: ";
    cin >> nm;

    cout << "Enter file data: ";
    cin.ignore();
    getline(cin, d);

    fs.addFile(nm, d);
    break;

case 2:
    cout << "Enter file name: ";
    cin >> nm;

    fs.deleteFile(nm);
    break;

case 3:
    fs.displayFile();
    break;

case 4:
    cout << "Enter file name: ";
    cin >> nm;
    fs.searchFile(nm);
    break;

case 5:
    cout << "Enter old file name: ";
    cin >> nm;

    cout << "Enter new file name: ";
    cin.ignore();
    getline(cin, d);

    fs.updateFilename(nm, d);
    break;

case 6:
    cout << "Enter file name: ";
    cin >> nm;

    cout << "Enter file data: ";
    cin.ignore();
    getline(cin, d);

    fs.updateFiledata(nm, d);
    break;

case 7:

    exit(0);
```

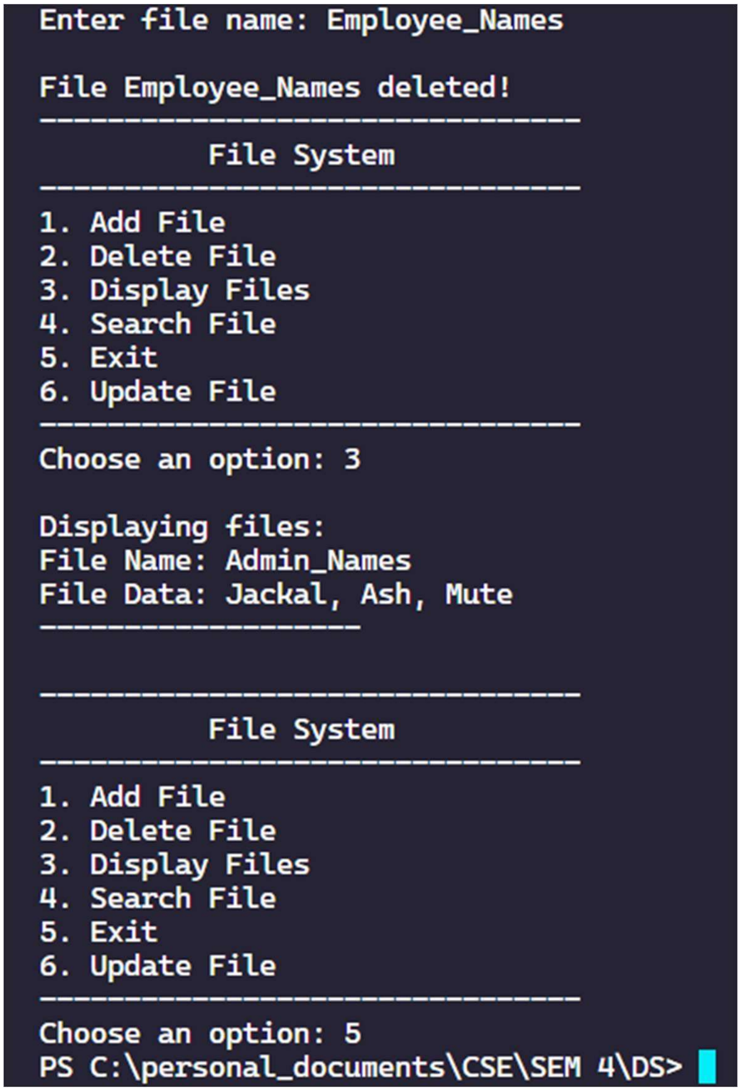
```
            break;
        default:
            cout << " Invalid Choice!";
            break;
        }
    } while (ch != 7);

    return 0;
}
```



```
----------------------------------
          File System
----------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
----------------------------------
Choose an option: 1
Enter file name: Admin_Names
Enter file data: Jackal, Ash, Mute, Twitch

File Admin_Names Added!

----------------------------------
          File System
----------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
----------------------------------
Choose an option: 1
Enter file name: Employee_Names
Enter file data: Smoke, Demos, Finka, Osa, Echo

File Employee_Names Added!

----------------------------------
          File System
----------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
----------------------------------
Choose an option: 3

Displaying files:
File Name: Admin_Names
File Data: Jackal, Ash, Mute, Twitch
--------------------
```



```
File Name: Employee_Names
File Data: Smoke, Demos, Finka, Osa, Echo
--------------------

----------------------------------
          File System
----------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
----------------------------------
Choose an option: 4
Enter file name: Admin_Names

File found!
File Name: Admin_Names
File Data: Jackal, Ash, Mute, Twitch
----------------------------------
          File System
----------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
----------------------------------
Choose an option: 6
Enter file name: Admin_Names
Enter file data: Jackal, Ash, Mute

File 'Admin_Names' updated!
----------------------------------
          File System
----------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
----------------------------------
Choose an option: 2
```



```
Enter file name: Employee_Names

File Employee_Names deleted!
--------------------------------
           File System
--------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
--------------------------------
Choose an option: 3

Displaying files:
File Name: Admin_Names
File Data: Jackal, Ash, Mute
-------------------


--------------------------------
           File System
--------------------------------
1. Add File
2. Delete File
3. Display Files
4. Search File
5. Exit
6. Update File
--------------------------------
Choose an option: 5
PS C:\personal_documents\CSE\SEM 4\DS>
```