**Bahria University**

**Lahore Campus**



***FINAL PROJECT OF DSA***

**INSTRUCTOR NAME:**

**Fatima Zulfiqar**

***NAME OF GROUP MEMBERS:***

1. ***AFFAN AHMAD (03-134221-003.)***
2. ***ZAIER SHAKEEL (03-134192-043.)***
3. ***ALI WASIF ZIA (03-134221-009).***

***Date: 12-06-2023***

***Table of Contents***

[ inrtoduction: 3](#_Toc61733599)

[ **Functionalities:** 3](#_Toc61733600)

[ ***Menu:*** 3](#_Toc61733601)

[ ***Classes:*** 3](#_Toc61733602)

[ ***Insert:*** 3](#_Toc61733603)

[ ***Update:*** 3](#_Toc61733604)

[ ***Show/Display:*** 3](#_Toc61733605)

[ ***Delete:*** 3](#_Toc61733606)

[ ***Search:*** 4](#_Toc61733607)

[ Class Diagrams: 4](#_Toc61733608)

[ Complete working code: 5](#_Toc61733609)

[Classes of the program: 5](#_Toc61733610)

[1) Tourist 5](#_Toc61733611)

[2) Admin 5](#_Toc61733612)

[3) Tourism 5](#_Toc61733613)

[Data member of the classes: 5](#_Toc61733614)

[1) Tourist: 5](#_Toc61733615)

[2) Admin: 5](#_Toc61733616)

[3) Tourism: 5](#_Toc61733617)

[ CODE OF THE PROGRAM: 5](#_Toc61733618)

[ Screenshot of the output: 10](#_Toc61733619)

[ Applications of the project: 13](#_Toc61733620)

[ Scope of the project: 14](#_Toc61733621)

[ Participation of each member: 14](#_Toc61733622)

TITLE: Tour Registration System:

# inrtoduction:

Tour registration system is basically a registration system that helps the admin and also tourist to registers the tour according to their required destination. In this project it could help the admin to easily work and registered the tourist with the given functionalities those are defined below that are:

* INSERT
* DELETE
* UPDATE
* SHOW
* SEARCH
* MENU

# **Functionalities:**

## ***Menu:***

We make **menu** so that user and admin could easily visualize the proper functionalities of the program and the given functionalities are given in this menu using switch feature in C++

## ***Classes:***

Here we make two main classes one for the node or also moving data into tree, so the total classes are three.

## ***Insert:***

In insert function we do the insertion work for admin and tourist so that he/she could easily insert the tourist very efficiently.

## ***Update:***

In update function we do the updating work for admin so that he/she could easily update the tourist according to the id that we give and with respect to that id admin easily changes the name tour destiny and registration number very efficiently.

## ***Show/Display:***

In the show/Display function admin can easily visualize all the tourist list those were entered for tours very easily and also it is display in inorder traversal tree.

## ***Delete:***

In delete function we do the deleting work for admin so that he/she could easily delete the tourist according to the id that we give and with respect to that id admin easily delete the name tour destiny and registration number very efficiently.

## ***Search:***

In the search function we do the searching using id that we assign to the tourist.

**We use C++ language and the data structure we used is DSA and in this particular project we worked in BST (Binary Search Tree).**

|  |
| --- |
| **Tourism** |
| Data::Admin |

# Class Diagrams:

|  |
| --- |
| **Turist** |
| -id: int  -name: string  -adress: string  -number: string |
| + insert(): Turist\*  +Display():void  +Search():Admin\*  +update():void  +Delete(): Admin\* |

|  |
| --- |
| **Admin** |
| +Data: Turist  +left: Admin\*  +right: Admin\* |

# Complete working code:

In the program we enter the data or store the data/Information of the user and also modify and deleted and also show the information. In the first class name is tourist and we declared the id, name, address, phone number, for getting input the data or information of the tourist or user. And second class well be used for handling the left and right of the bst (binary search tree) of the tourist and name of the second class is admin. And the last class name is tourism and in this class we will collect data and also it modify and search and display by menu. In this class the main part is to insert the functions and also it work by using the bst, and the main part is all the function are working in the recursive way because it is easy for us.

## Classes of the program:

### Tourist

### Admin

### Tourism

## Data member of the classes:

### Tourist:

#### Id

#### Name

#### Address

#### Phone number.

### Admin:

#### Tourist data.

#### Admin \*left.

#### Admin \*right.

### Tourism:

#### Admin \*root.

#### Function:

##### Insert

##### Search

##### Update/Modify.

##### Delete

##### Display/show (in order way).

##### Menu.

# CODE OF THE PROGRAM:

#include<iostream>

#include<string>

#include<fstream>

using namespace std;

class Turist {

public:

int id;

string name;

string adress;

string number;

};

class Admin {

public:

Turist data;

Admin\* left;

Admin\* right;

Admin() {

left = right = NULL;

}

};

class Tourism {

public:

Admin\* root;

Tourism() {

root = NULL;

}

Admin\* insert(Admin\* root, Admin\* ptr) {

if (root == NULL) {

return ptr;

}

else if (ptr->data.id < root->data.id)

root->left = insert(root->left, ptr);

else if (ptr->data.id > root->data.id)

root->right = insert(root->right, ptr);

return root;

}

Admin\* deleteNode(Admin\* root, int key) {

if (root == NULL)

return root;

if (key < root->data.id)

root->left = deleteNode(root->left, key);

else if (key > root->data.id)

root->right = deleteNode(root->right, key);

else {

if (root->left == NULL) {

Admin\* temp = root->right;

delete root;

return temp;

}

else if (root->right == NULL) {

Admin\* temp = root->left;

delete root;

return temp;

}

Admin\* temp = minValueNode(root->right);

root->data.id = temp->data.id;

root->right = deleteNode(root->right, temp->data.id);

}

return root;

}

Admin\* minValueNode(Admin\* node) {

Admin\* current = node;

while (current && current->left != NULL)

current = current->left;

return current;

}

Admin\* search(Admin\* root, int value) {

if (root == NULL) {

cout << "Not Found" << endl;

return NULL;

}

if (root->data.id == value)

return root;

if (value < root->data.id)

return search(root->left, value);

return search(root->right, value);

}

void display(Admin\* temp) {

if (temp == NULL) {

return;

}

display(temp->left);

cout << temp->data.id << endl;

cout << temp->data.name << endl;

cout << temp->data.adress << endl;

cout << temp->data.number << endl;

display(temp->right);

}

void insert\_record() {

Turist temp;

cout << "Enter ID Number: ";

cin >> temp.id;

cout << "Enter Your Name: ";

cin >> temp.name;

cout << "Enter Your Address: ";

cin >> temp.adress;

cout << "Enter Your Phone Number: ";

cin >> temp.number;

Admin\* ptr = new Admin();

ptr->data = temp;

root = insert(root, ptr);

cout << "Record added" << endl;

ofstream outFile("tourists.txt", ios::app);

if (outFile.is\_open()) {

outFile << temp.id << " " << temp.name << " " << temp.adress << " " << temp.number << endl;

outFile.close();

}

else {

cout << "Error: Unable to open the file." << endl;

}

}

void delete\_record() {

int id;

cout << "Enter ID Number to Delete: ";

cin >> id;

Admin\* temp = deleteNode(root, id);

if (temp != NULL)

cout << "Record Found and Deleted" << endl;

else

cout << "Record Not Found" << endl;

}

void search\_record() {

int value;

cout << "Enter ID Number to Search: ";

cin >> value;

Admin\* temp = search(root, value);

if (temp != NULL)

cout << "Record Found" << endl;

else

cout << "Record Not Found" << endl;

}

void update\_record() {

int id;

cout << "Enter ID Number to Update: ";

cin >> id;

Admin\* temp = search(root, id);

if (temp != NULL) {

cout << "Enter 1 to Change Name" << endl;

cout << "Enter 2 to Change Address" << endl;

cout << "Enter 3 to Change Phone Number" << endl;

int choice;

cin >> choice;

if (choice == 1) {

cout << "Enter New Name: ";

cin >> temp->data.name;

}

else if (choice == 2) {

cout << "Enter New Address: ";

cin >> temp->data.adress;

}

else if (choice == 3) {

cout << "Enter New Phone Number: ";

cin >> temp->data.number;

}

}

else {

cout << "Record Not Found" << endl;

}

}

void print\_inorder() {

display(root);

}

void read\_records\_from\_file() {

ifstream inFile("tourists.txt");

if (inFile.is\_open()) {

while (!inFile.eof()) {

Turist temp;

inFile >> temp.id >> temp.name >> temp.adress >> temp.number;

Admin\* ptr = new Admin();

ptr->data = temp;

root = insert(root, ptr);

}

inFile.close();

}

else {

cout << "Error: Unable to open the file." << endl;

}

}

};

void menu() {

Tourism t;

//t.read\_records\_from\_file();

int choice;

do {

cout << "\n\n\*\*\*\*\*\*\*\*\*\* Tour Registration System \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "1. Insert a record" << endl;

cout << "2. Display records (inorder traversal)" << endl;

cout << "3. Search for a record" << endl;

cout << "4. Update a record" << endl;

cout << "5. Delete a record" << endl;

cout << "6. Exit" << endl;

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

cout << endl;

t.insert\_record();

break;

case 2:

cout << "\n\*\*\*\*\*\* Inorder Traversal \*\*\*\*\*\*" << endl;

t.print\_inorder();

break;

case 3:

cout << endl;

t.search\_record();

break;

case 4:

cout << endl;

t.update\_record();

break;

case 5:

cout << endl;

t.delete\_record();

break;

case 6:

cout << "\nExiting the program." << endl;

break;

default:

cout << "\nInvalid choice. Please try again." << endl;

break;

}

} while (choice != 6);

}

int main() {

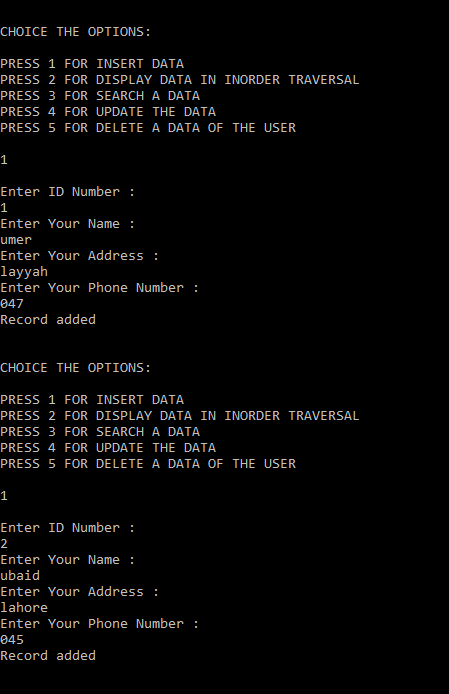
menu();

return 0;

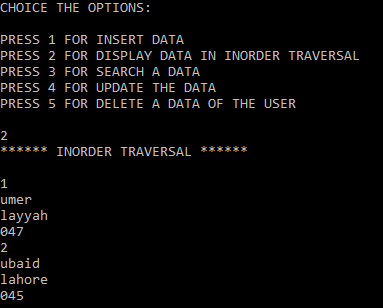
}

# Screenshot of the output:

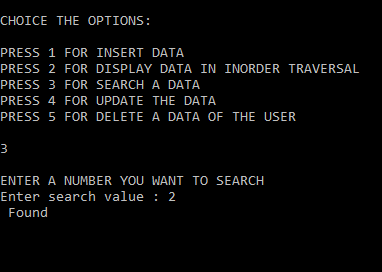
* Insert the record for two time:



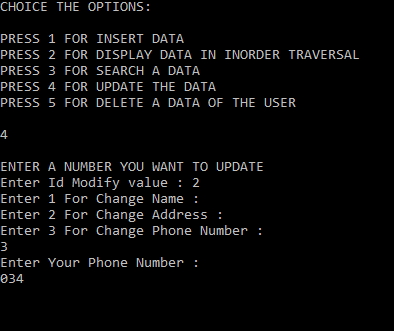
* Display the added record:



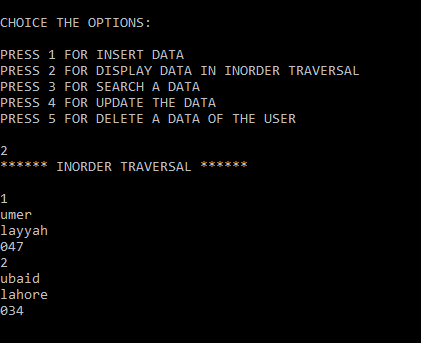
* Search the record:



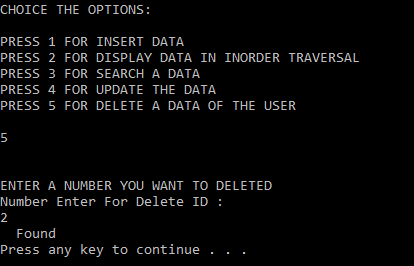
* Modify the record:



* Again display the added record:



* Delete the record.



# Applications of the project:

* For the admin:
* Save time.
* Easy administration of the user.
* Improve efficiency and increase value creation
* Increase the marketing effect
* Secure data processing
* Reduce time spent on billing
* Increase the probability of registration
* **For participants:**
* Save time
* Improve customer service
* User-friendly payment solution
* Easy to stay updated
* Registration of several participants at the same time
* Secure data processing

# Scope of the project:

Actually this program will help the user to findtour. The tourism is considered to be important for growing the economic and also it will help to develop the country or nation. According to the [**United Nations World Tourism Organization (UNWTO)**](http://www.abto.org.bt/2010/06/unwto-tourism-2020-vision-forecast-released/)**,** global tourism is expected to reach 1.6 billion (in terms of international arrivals) by the year 2020. Tourism management is generally considered a bright and potential employment sector as it offers a wide variety of career opportunities in both the private and public sector.

# Participation of each member:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | Members Name | ZAIER (043) | AFFAN (045) | ALI WASIF (009) |
| 2 | Participation in the project | Display and menu of the program. | Search and deleted information/file handling | Insert and update information |