AIM:-

Assignment 3

a. Create a database using array of structures and perform following operations on it:

i. Add record

ii. Display Database

iii. Search record (binary search)

b. For database implemented using array, perform

iv. Modify record

v. Delete record

vi. Sort records (Bubble sort)

OBJECTIVE:-

To create database with functions such as add record, display record, search record, modify record, delete record, sort record using array of structures.

THEORY:-

## Arrays of Structures

Since an array can contain similar elements, the combination having structures within an array is an array of structures. To declare an array of structures, you must first define a structure and then declare an array variable of that type. For example, to store addresses of 100 members of the council, you need to create an array.

Now, to declare a 100-element array of structures of type addr (defined in previous chapters), we will write :

addr mem\_addr [100];

This creates 100 sets of variables that are organised as defined in the structure addr. To access a specific structure, index the structure name. For instance, to print the houseno of structure 8, write :

cout << mem\_addr[7].houseno ;

ALGORITHM:-

Step 1 Create structure of student having variables as roll no. and name of the student.

Step 2 Get the input of the data of the student using objects created.

Step 3 Display the input data of the student.

Step 4 Using the switch statement display the functions insert, delete, display, search and modify.

Step 5 Input the choice of the user using the switch statement and respectively call the functions.

Step 6 for choice == 4 implement binary search for searching the element.

SOURCE CODE:-

#include <iostream>

# define max 10

using namespace std;

typedef struct student{

int roll;

char name[max];

} d;

d s[max];

int main()

{ int t,i,choice,m,k,j,temp;

cout<<" maximum number of student is 10 "<<endl;

cout<< "enter total number of students : "<<endl;

cin>> t;

for ( i=0;i<t;i++){

cout<<" enter roll number : "<<endl;

cin>>s[i].roll;

cout<<" enter name of student : "<<endl;

cin>>s[i].name;

}

for ( i=0;i<t;i++){

cout<<"roll number : "<<s[i].roll<<endl;

cout<<"name of student : "<<s[i].name <<endl;

}

cout<<"press 1 for insert "<<endl;

cout<<"press 2 for delete"<<endl;

cout<<"press 3 for display "<<endl;

cout<<"press 4 to search "<<endl;

cout<<"press 5 to modify "<<endl;

cout<<"press 6 to sort the list according to roll number "<<endl;

cout<<"press 0 to exit "<<endl;

cout<<" your choice : ";

cin>>choice;

while (choice != 0){

switch(choice){

case 1 : if (t>= 10){

cout<<"full "<<endl;

}

else{

cout<<"enter roll number"<<endl;

cin>>s[t].roll;

cout<<"enter name "<<endl;

cin>>s[t].name;

t=t+1;}

cout<<"enter choice"<<endl;

cin>>choice;

case 2: if(t==0){

cout<<"empty "<<endl;}

else{

t=t-1;

cout<<"deleted "<<endl;}

cout<<"enter choice"<<endl;

cin>>choice;

case 3:for ( i=0;i<t;i++){

cout<<"roll number : "<<s[i].roll<<endl;

cout<<"name of student : "<<s[i].name <<endl;

}

cout<<"enter choice"<<endl;

cin>>choice;

case 4: cout<<"enter roll number to be searched "<<endl;

cin>>m;

for ( i=0;i<t;i++){

if(s[i].roll == m){

cout<<" found "<<endl;

}}

cout<<"enter choice"<<endl;

cin>>choice;

case 5:cout<<"enter roll number to be modify "<<endl;

cin>>k;

for ( i=0;i<t;i++){

if(s[i].roll == k){

cout<<" found "<<endl;

cout<<"enter new roll number"<<endl;

cin>>s[i].roll;

cout<<"enter new name "<<endl;

cin>>s[i].name;

}}

cout<<"enter choice"<<endl;

cin>>choice;

case 6: for(i=0;i<t-1;i++){

for(j=0;j<t-1;j++){

if(s[j].roll>s[j+1].roll){

temp=s[j].roll;

s[j].roll= s[j+1].roll;

s[j+1].roll=temp;

}

}

} cout<<"enter choice"<<endl;

cin>>choice;

}}

return 0;

}

OUTPUT:-

maximum number of student is 10

enter total number of students :

2

enter roll number :

23

enter name of student :

Akshata

enter roll number :

274

enter name of student :

Siddhi

roll number : 23

name of student : Akshata

roll number : 274

name of student : Siddhi

press 1 for insert

press 2 for delete

press 3 for display

press 4 to search

press 5 to modify

press 6 to sort the list according to roll number

press 0 to exit

your choice : 1

enter roll number

33

enter name

Shruti

enter choice

3

deleted

enter choice

2

roll number : 23

name of student : Akshata

roll number : 274

name of student : Siddhi

enter choice

0

CONCLUSION:-

1. How to handle such huge kind of databases using array of structures.

2. To know about time complexity of binary search, which is O(1) in case of iterative implementation; and in case of recursive implementation, O(log n) recursion call stack space.