**ASSIGNMENT 3**

**AIM:**

a. Create a data base using array of structures and perform following operations using binary search on it:

1. Add record
2. Display Database
3. Search record

b. For data base implemented using array, perform by bubble sort,

iv. Modify record

1. Delete record
2. Sort records

**OBJECTIVE:**

The objective of this program is to learn to create an efficient database system using array of structures and learn the searching and sorting techniques.

**THEORY:**

**Array of structures:**

An array of structures is simply an array in which each element is

a structure of the same type. The referencing and subscripting of these arrays (also

called structure arrays) follow the same rules as simple arrays.

**Binary Search:**

It is a Faster method of searching

BUT list must be ordered.

Sometimes called a binary chop as it splits the data list into two sublists and repeats the

process until a search key is found.

**Bubble sort:**

Bubble sort, sometimes referred to as sinking sort, is a simple sorting algorithm

that repeatedly steps through the list, compares adjacent elements and swaps them if they are

in the wrong order. The pass through the list is repeated until the list is sorted.

**CODE:**

#include<iostream>

using namespace std;

typedef struct stud

{

int roll;

char name[20],add[20];

}s1;

s1 s[10],temp;

int n;

void getdata()

{

cout<<"Enter number of records:";

cin>>n;

cout<<"\nEnter records:";

for(int i=0;i<n;i++)

{

cin>>s[i].roll>>s[i].name>>s[i].add;

}

}

void display()

{

cout<<"\nRecords are: \n";

for(int i=0;i<n;i++)

{

cout<<s[i].roll<<s[i].name<<s[i].add<<endl;

}

}

void bubble()

{

int i,j;

for(i=0;i<n-1;i++)

{

for(j=0;j<n-i-1;j++)

{

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

{

temp=s[j];

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

s[j+1]=temp;

}

}

}

}

void search()

{

int key,ub=n-1,lb=0,mid;

cout<<"\n Enter the rollno to be searched";

cin>>key;

while(lb<=ub)

{

mid=(lb+ub)/2;

if(s[mid].roll==key)

{ cout<<"The roll no is found\n";

break;}

else if(s[mid].roll<key)

lb=mid+1;

else

ub=mid-1;

}

if(lb>ub)

cout<<"The key is not found";

}

int main()

{

int ch;

while(ch!=5)

{cout<<"\n1.get data \t"<<"2.display\t"<<"3.bubble\t"<<"4.search\t"<<"5.exit"<<endl;

cout<<"Enter the choice:";

cin>>ch;

switch(ch)

{

case 1:getdata();

break;

case 2:display();

break;

case 3:bubble();

break;

case 4:search();

break;

case 5:return 0;

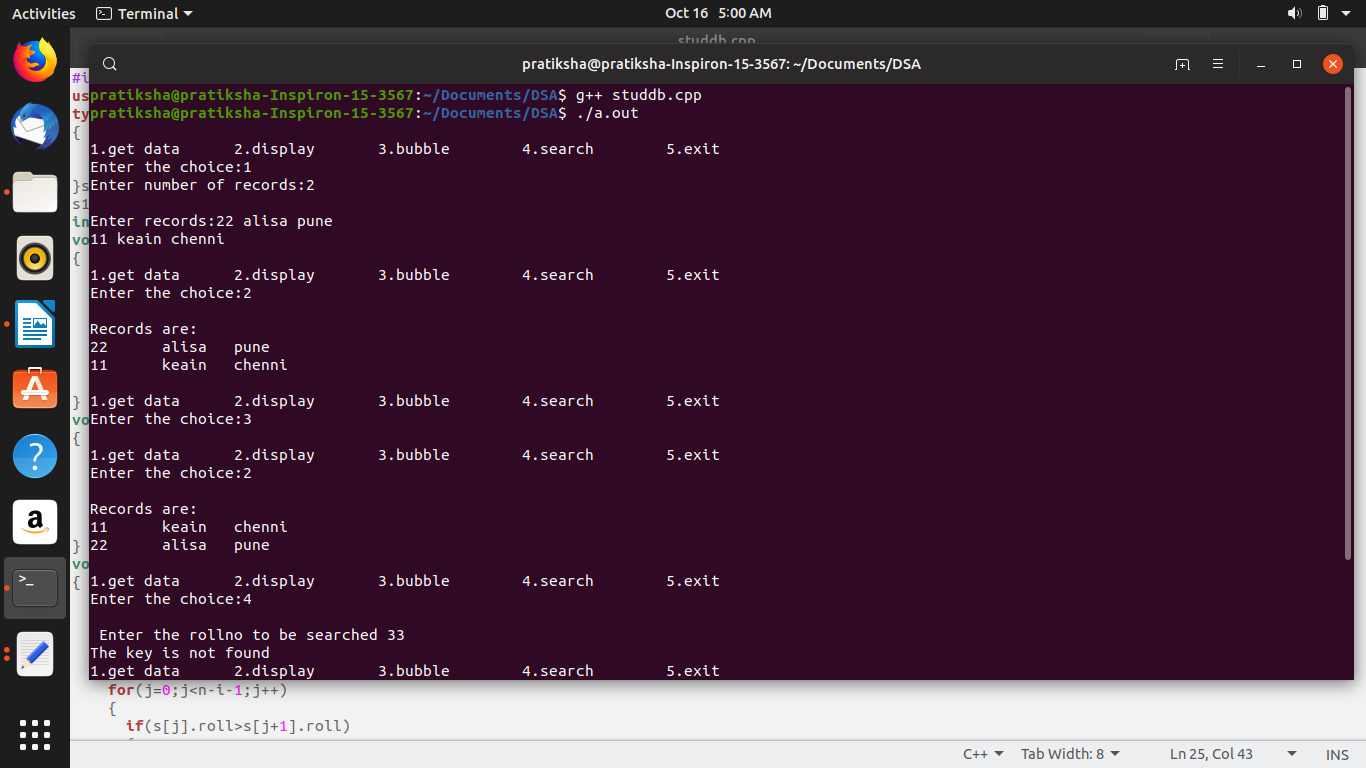
default:cout<<"\nWrong choice";

}}

return 0;

}

**OUTPUT:**



**COMPLEXITY:**

1) Bubble sort: O(n^2)

2) Binary Search: O(log n)

**CONCLUSION:**

In this program we learnt how to sort and search the program using bubble sort and

binary search. We also learnt how to perform using array of structures.

Last modified: 7:18 am