Ex. No. 9	IMPLEMENTATION OF HASHING
06-09-2017	

Question:

To implement the working of implementation of hashing.

Algorithm:

1. Start.
2. For inserting an element,
3. If it is the first element, it is already sorted. return 1;
4. Pick next element.
5. Compare with all elements in the sorted sub-list
6. Shift all the elements in the sorted sub-list that is greater than the value to be sorted
7. Insert the value
8. Repeat until list is sorted
9. For selection sort, follow the steps:
10. Set MIN to location 0
11. Search the minimum element in the list
12. Swap with value at location MIN
13. Increment MIN to point to next element
14. Repeat until list is sorted
15. End.

Program:

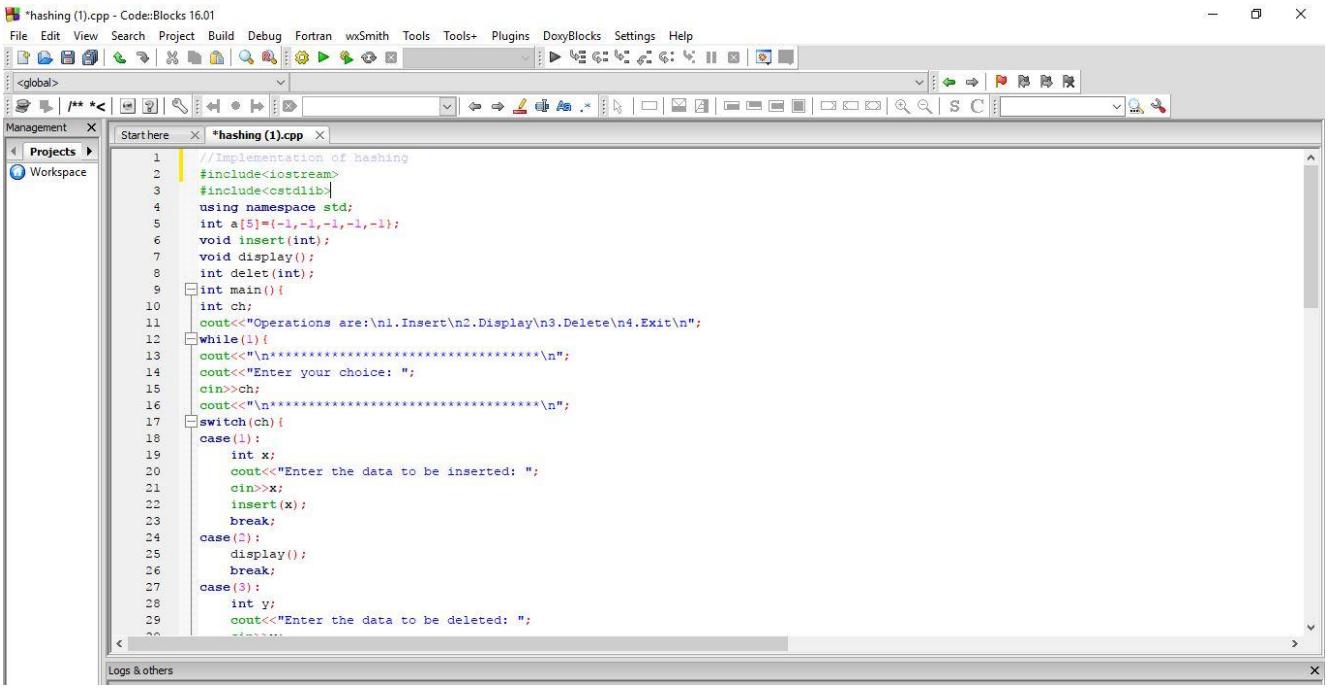
```
#include<iostream>
#include<cstdlib>
using namespace std;
int a[5]={-1,-1,-1,-1,-1};
void insert(int);
void display();
int delet(int);
int main(){
int ch;
cout<<"Operations are:\n1.Insert\n2.Display\n3.Delete\n4.Exit\n";
while(1){
cout<<"\n**********************************\n";
cout<<"Enter your choice: ";
cin>>ch;
cout<<"\n**********************************\n";
switch(ch){
case(1):
   int x;
   cout<<"Enter the data to be inserted: ";
   cin>>x;
   insert(x);
```

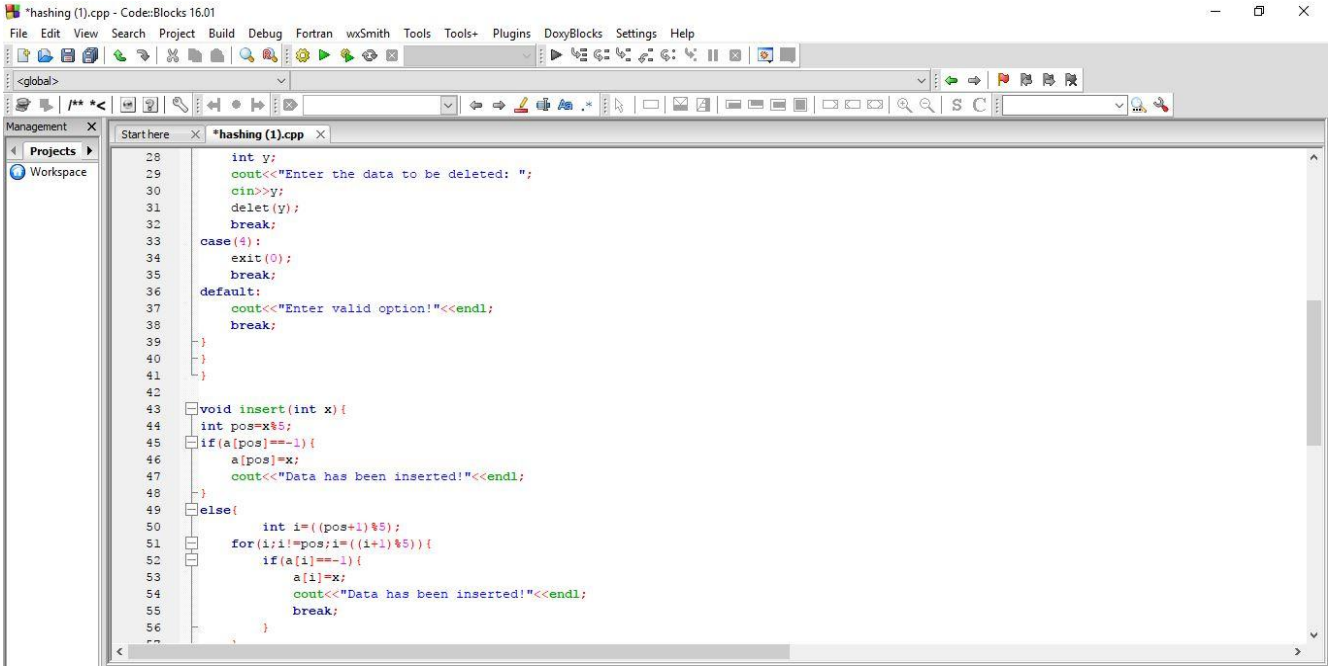
```
    break;
case(2):
    display();
    break;
case(3):
    int y;
    cout<<"Enter the data to be deleted: ";
    cin>>y;
    delet(y);
    break;
case(4):
    exit(0);
    break;
default:
    cout<<"Enter valid option!"<<endl;
    break;
}
}
}
void insert(int x){
int pos=x%5;
if(a[pos]==-1){
```

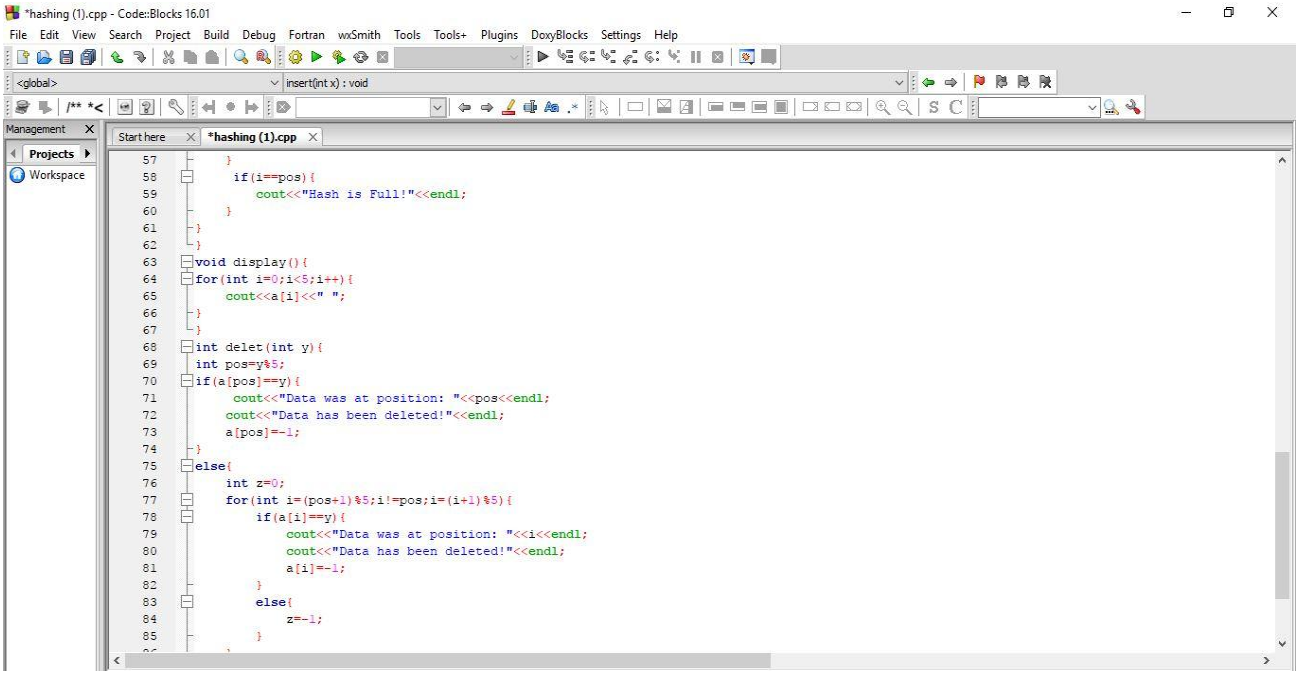
```
    a[pos]=x;
    cout<<"Data has been inserted!"<<endl;
}
else{
        int i=((pos+1)%5);
    for(i;i!=pos;i=((i+1)%5)){
        if(a[i]==-1){
            a[i]=x;
            cout<<"Data has been inserted!"<<endl;
            break;
        }
    }
     if(i==pos){
        cout<<"Hash is Full!"<<endl;
    }
}
}
void display(){
for(int i=0;i<5;i++){
    cout<<a[i]<<" ";
}
}
```

```
int delet(int y){
int pos=y%5;
if(a[pos]==y){
     cout<<"Data was at position: "<<pos<<endl;
    cout<<"Data has been deleted!"<<endl;
    a[pos]=-1;
}
else{
    int z=0;
    for(int i=(pos+1)%5;i!=pos;i=(i+1)%5){
        if(a[i]==y){
            cout<<"Data was at position: "<<i<<endl;
            cout<<"Data has been deleted!"<<endl;
            a[i]=-1;
        }
        else{
            z=-1;
        }}
if(z==-1){
    cout<<"Data was not found!"<<endl;
}}}
```

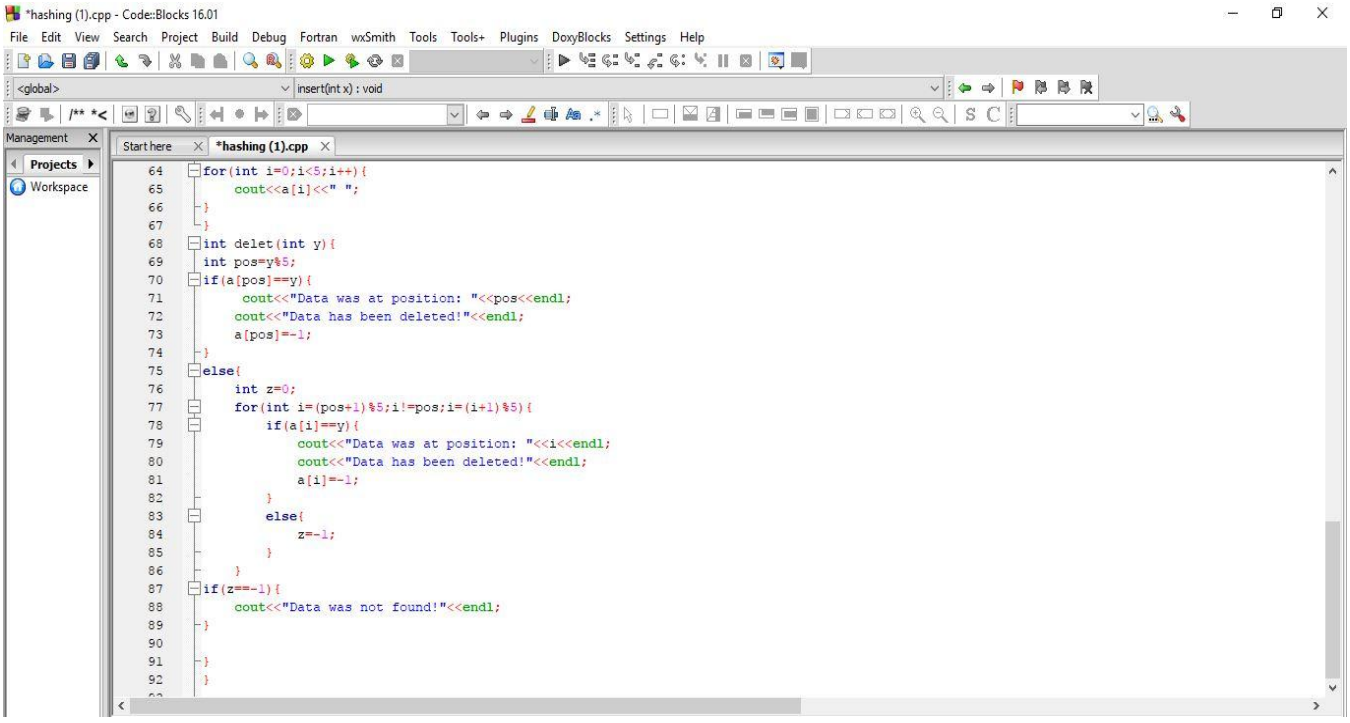
Output:


*hashing (1).cpp - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
Projects
Workspace
Start here
*hashing (1).cpp
//Implementation of hashing
#include<iostream>
#include<cstdlib>
using namespace std;
int a[5]={-1,-1,-1,-1,-1};
void insert(int);
void display();
int delet(int);
int main(){
int ch;
cout<<"Operations are:\n1.Insert\n2.Display\n3.Delete\n4.Exit\n";
while(1){
cout<<"\n**********\n";
cout<<"Enter your choice: ";
cin>>ch;
cout<<"\n**********\n";
switch(ch){
case(1):
int x;
cout<<"Enter the data to be inserted: ";
cin>>x;
insert(x);
break;
case(2):
display();
break;
case(3):
int y;
cout<<"Enter the data to be deleted: ";
Logs & others


*hashing (1).cpp - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
Projects
Workspace
Start here
*hashing (1).cpp
int y;
cout<<"Enter the data to be deleted: ";
cin>>y;
delet(y);
break;
case(4):
exit(0);
break;
default:
cout<<"Enter valid option!"<<endl;
break;
}
}
}
void insert(int x){
int pos=x%5;
if(a[pos]==-1){
a[pos]=x;
cout<<"Data has been inserted!"<<endl;
}
else{
int i=((pos+1)%5);
for(i;i!=pos;i=((i+1)%5)){
if(a[i]==-1){
a[i]=x;
cout<<"Data has been inserted!"<<endl;
break;
}



```
    }
     if(i==pos){
        cout<<"Hash is Full!"<<endl;
    }
}
}
void display(){
for(int i=0;i<5;i++){
    cout<<a[i]<<" ";
}
}
int delet(int y){
 int pos=y%5;
if(a[pos]==y){
     cout<<"Data was at position: "<<pos<<endl;
    cout<<"Data has been deleted!"<<endl;
    a[pos]=-1;
}
else{
    int z=0;
    for(int i=(pos+1)%5;i!=pos;i=(i+1)%5){
        if(a[i]==y){
            cout<<"Data was at position: "<<i<<endl;
            cout<<"Data has been deleted!"<<endl;
            a[i]=-1;
        }
        else{
            z=-1;
        }
```



```
for(int i=0;i<5;i++){
    cout<<a[i]<<" ";
}
}
int delet(int y){
 int pos=y%5;
if(a[pos]==y){
     cout<<"Data was at position: "<<pos<<endl;
    cout<<"Data has been deleted!"<<endl;
    a[pos]=-1;
}
else{
    int z=0;
    for(int i=(pos+1)%5;i!=pos;i=(i+1)%5){
        if(a[i]==y){
            cout<<"Data was at position: "<<i<<endl;
            cout<<"Data has been deleted!"<<endl;
            a[i]=-1;
        }
        else{
            z=-1;
        }
    }
if(z==-1){
    cout<<"Data was not found!"<<endl;
}

}
}
```

```
"C:\Users\JUSTIN PAUL\Downloads\hashing (1).exe"
Operations are:
1.Insert
2.Display
3.Delete
4.Exit

**********************************
Enter your choice: 1

**********************************
Enter the data to be inserted: 12
Data has been inserted!

**********************************
Enter your choice: 1

**********************************
Enter the data to be inserted: 10
Data has been inserted!

**********************************
Enter your choice: 1

**********************************
Enter the data to be inserted: 23
Data has been inserted!

**********************************
Enter your choice: 1

**********************************
Enter the data to be inserted: 34
Data has been inserted!

**********************************
Enter your choice: 1

**********************************
Enter the data to be inserted: 45
Data has been inserted!

**********************************
Enter your choice: 1
```

```
"C:\Users\JUSTIN PAUL\Downloads\hashing (1).exe"

**********************************
Enter the data to be inserted: 56
Hash is Full!

**********************************
Enter your choice: 2

**********************************
10 45 12 23 34
**********************************
Enter your choice: 3

**********************************
Enter the data to be deleted: 10
Data was at position: 0
Data has been deleted!

**********************************
Enter your choice: 2

**********************************
-1 45 12 23 34
**********************************
Enter your choice: 3

**********************************
Enter the data to be deleted: 23
Data was at position: 3
Data has been deleted!

**********************************
Enter your choice: 2

**********************************
-1 45 12 -1 34
**********************************
Enter your choice: 4

**********************************

Process returned 0 (0x0)   execution time : 77.177 s
Press any key to continue.
```

VIDEO URL:

https://youtu.be/TcTiq5PtQbs

RESULT:

The program of implementation of hash function is implemented successfully and the output is verified.