Answer

SOLUTION:-

```
#include <iostream>
#include "unsortedtype.h"
#include "unsortedtype.cpp"
using namespace std;


//class studentInfo
class studentInfo {
public:
int ID;
string name;
double cgpa;

studentInfo() {
ID = 0;
name = "";
cgpa = 0.0;
}

studentInfo(int i, string n, double c) {
ID = i;
name = n;

cgpa = c;
}
bool operator==(studentInfo &other) {
return ID==other.ID && name == other.name && cgpa == other.cgpa;
}
bool operator!=(studentInfo &other) {
return ID!=other.ID || name != other.name || cgpa != other.cgpa;
}
void print() {
cout << ID << ", " << name << ", " << cgpa << endl;
}
};
int main()
{
//Create a list of integers
UnsortedType<int> lst;

//Insert 4 items
lst.InsertItem(5);
```

```
lst.InsertItem(7);
lst.InsertItem(6);
lst.InsertItem(9);

//Print list
int x;
for(int i=0;i<lst.LengthIs();i++) {
lst.GetNextItem(x);
cout << x << " ";
}

//Print length of list
cout << endl << lst.LengthIs() << endl;

//Insert one item
lst.InsertItem(1);


//print list
lst.ResetList();
for(int i=0;i<lst.LengthIs();i++) {
lst.GetNextItem(x);
cout << x << " ";
}

//Retrieve 4
x = 4;
bool f;
lst.RetrieveItem(x, f);
if(f) cout << "\nItem is found\n";
else cout << "\nItem is not found\n";

x = 5;
lst.RetrieveItem(x, f);
if(f) cout << "Item is found\n";
else cout << "Item is not found\n";

x = 9;
lst.RetrieveItem(x, f);
if(f) cout << "Item is found\n";
else cout << "Item is not found\n";

x = 10;
lst.RetrieveItem(x, f);
if(f) cout << "Item is found\n";
else cout << "Item is not found\n";

//print if the list is full or not
```

```
if(lst.IsFull()) cout << "List is full\n";
else cout << "List is not full\n";

//Delete 5
lst.DeleteItem(5);

//print if the list is full or not
if(lst.IsFull()) cout << "List is full\n";
else cout << "List is not full\n";

//Delete 1
lst.DeleteItem(1);

//print list
lst.ResetList();
for(int i=0;i<lst.LengthIs();i++) {
lst.GetNextItem(x);
cout << x << " ";
}

cout << endl;

//Delete 6
lst.DeleteItem(6);

//print list
lst.ResetList();
for(int i=0;i<lst.LengthIs();i++) {
lst.GetNextItem(x);
cout << x << " ";
}

cout << endl;

//List of type studentInfo
UnsortedType<studentInfo> students;
studentInfo s1(15234, "Jon", 2.6);
studentInfo s2(13732, "Tyrion", 3.9);
studentInfo s3(13569, "Sandor", 1.2);
studentInfo s4(15467, "Ramsey", 3.1);
studentInfo s5(16285, "Arya", 3.1);

//Insert 5 records
students.InsertItem(s1);
students.InsertItem(s2);
students.InsertItem(s3);
students.InsertItem(s4);
students.InsertItem(s5);
```

```
//Delete record with ID 15467
students.DeleteItem(s4);

//Retrieve ID 13569]
students.RetrieveItem(s3, f);
if(f) {
cout << "Item is found\n";
s3.print();
} else {
cout << "Item is not found\n";
}

studentInfo s;
//print the list
for(int i=0;i<students.LengthIs();i++) {
students.GetNextItem(s);
s.print();
}
return 0;
}
```

Output

```
5 7 6 9
4
5 7 6 9 1
Item is not found
Item is found
Item is found

Item is not found
List is full
List is not full
9 7 6
9 7
Item is found
13569, Sandor, 1.2
15234, Jon, 2.6
13732, Tyrion, 3.9
13569, Sandor, 1.2
16285, Arya, 3.1
```
