# Assignment 12

Class - SE IV
Batch - F4

Roll NO - 21630
DOS - 2/12/2020

## Problem Statement -:

Write a program to implement priority queue in C++ using an inorder list to store the items in the queue. Create class that includes the data items (which should be template) and the priority (which should be int). The inorder list should contain these objects with operator <= overloaded so that the item with the highest priority appear at the start of the list (which will make it ~~reati~~ relatively easy to retrive the higher item.

## Objectives -:

TO understand insertion and deletion of nodes according to priorities.

## Theory -:

Priority Queue -:

priority queuqe is an extension of the queue that is similar to the queue in the certain aspects. and yet it differs from the ordinary queue in the following points.

1. Each item in the priority queue is associated with a priority.

2. The item with the highest priority is the first item to be removed from the queue

3. If more than one item has the same priority then that order in the queue is considered.

Operations on priority queue-:
(Implementation using array)

1. Initialize () = Make the queue empty()

2. Empty () - Determine if queue is empty or not

3. Full () · Determine if queue is full or not

4. enqueue() · Insert an element as per its priority.

5. Dequeue() - Delete the front element (Front element will have high priority)

6. Print() = prints all elements in the queue

* Application-:

1. Operating system load balancing and interupt handles.
2. Routing.
3. Hospital Emergency
4. Dijkstai's shortest path algorithm.

* Algorithm / Pseudocode -:

* ADT representation of class Node -:

```
template <class T>
class Node
{
   T data;         // data field
   int priority;   // priority of item
   node<T>next;       // next pointer

   template < class T1>
   friend class PrQueue;
   bool operator <= (const node & n1)
   {
      return this-priority <= n1.priority;
   }

};
```

* ADT representation of class PrQueue

```
template <class T>
Class PreQueue
{
  node<T> *front;  // front pointer of Queue
  enqueuePrQ();    // insert element by priority
  dequeuePrQ();    // delete element from front
  show();

}
```

```
* Procedure enqueuePrQ (Td int p)

      node<T> *temp = new Node<T>
      if ( front == NULL)
            front = temp;
            front -> next = NULL;
      else
          node<T> * r = front;
          if ( front <= temp)
          {
              temp -> next = front;
              front = temp;
          }
          else
          {
              node<T> *q = new Node<T>;
              q = r -> next;
              while( q != NULL && temp <= q)
              {
                  r = r -> next;
                  q = r -> next;
              }
              temp -> next = r -> next;
              r -> next = temp;
          }
      end else

  END procedure
```

```
* Procedure dequeuePrQ()

        Node<T>* temp=front
        if(front == NULL)
              Display "Queue is empty";
        front = front->next;
        delete temp:

  END procedure
```

```
* Procedure show()

        node <T> * P =front;
        if (front ==NULL)
              Display " Queue is empty";
        While( P!=NULL)
              Display P->data, P->priority.
              p=P->next;
        END While

  END Procedure
```

* Complexity-:

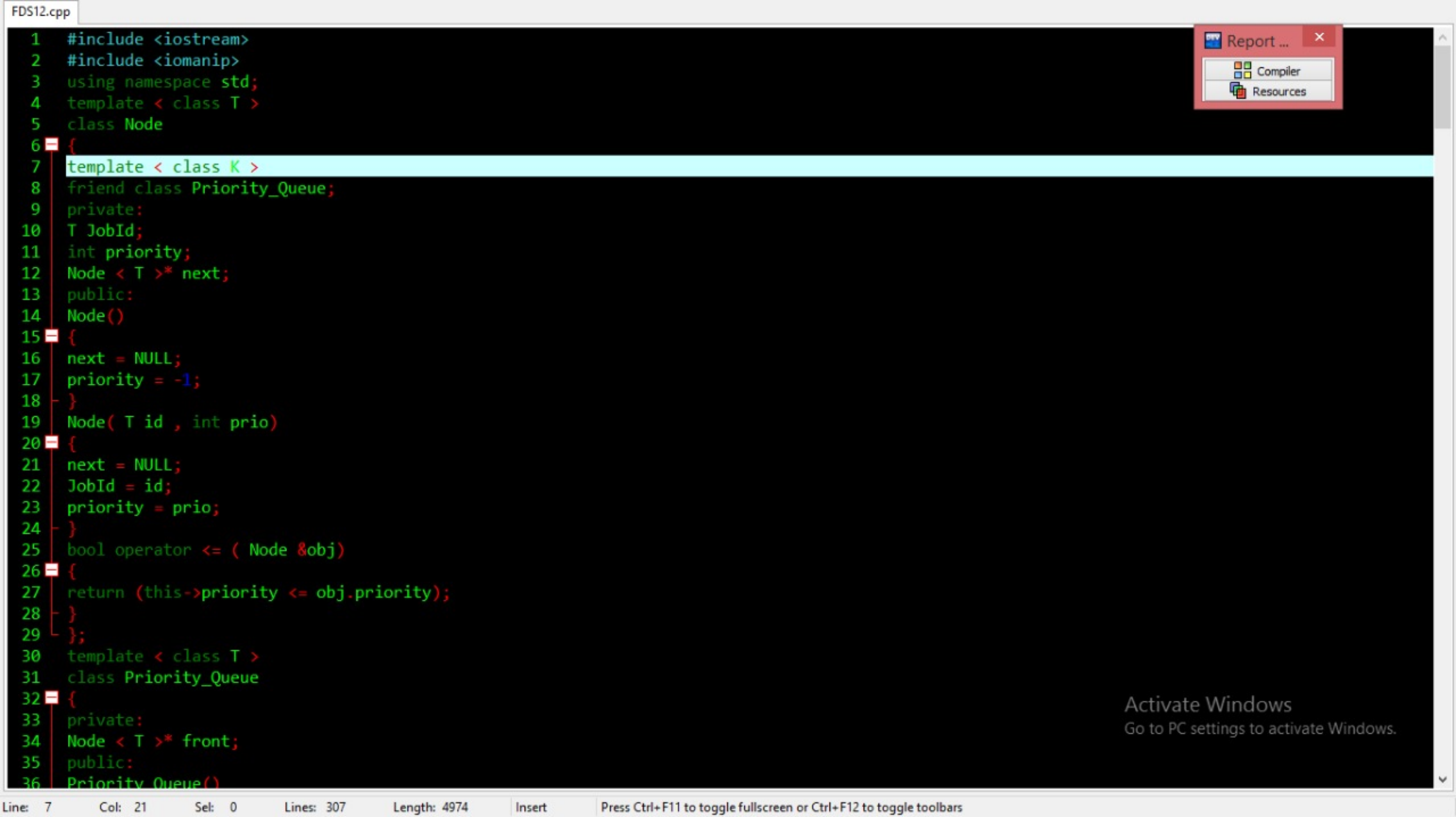
| | Function | Time Complexity |
|---|---|---|
| 1. | EnqueuePrQ() | o(logn) |
| 2. | dequeuePrQ() | o(logn) |

* Test Cases-:

| NO. | Description | Input | Expected o/p | Actual o/p | Result |
|---|---|---|---|---|---|
| 1. | 1.Insert<br>2.Delete<br>3.Show<br>4.Exit | ch=1<br>item=1<br>prority=5<br>item=2<br>prority=10<br>item=3<br>priority=2<br>ch = 3 | item priority<br>2 10<br>1 5<br>3 2 | item priority<br>2 10<br>1 5<br>3 2 | pass |
| 2. | 1.Insert<br>2.Delete<br>3. Show<br>4.Exit | ch=2<br>ch=3 | 2 Deleted<br>item priority<br>1 5<br>3 2 | 2 Deleted<br>item priority<br>1 5<br>3 2 | Pass |

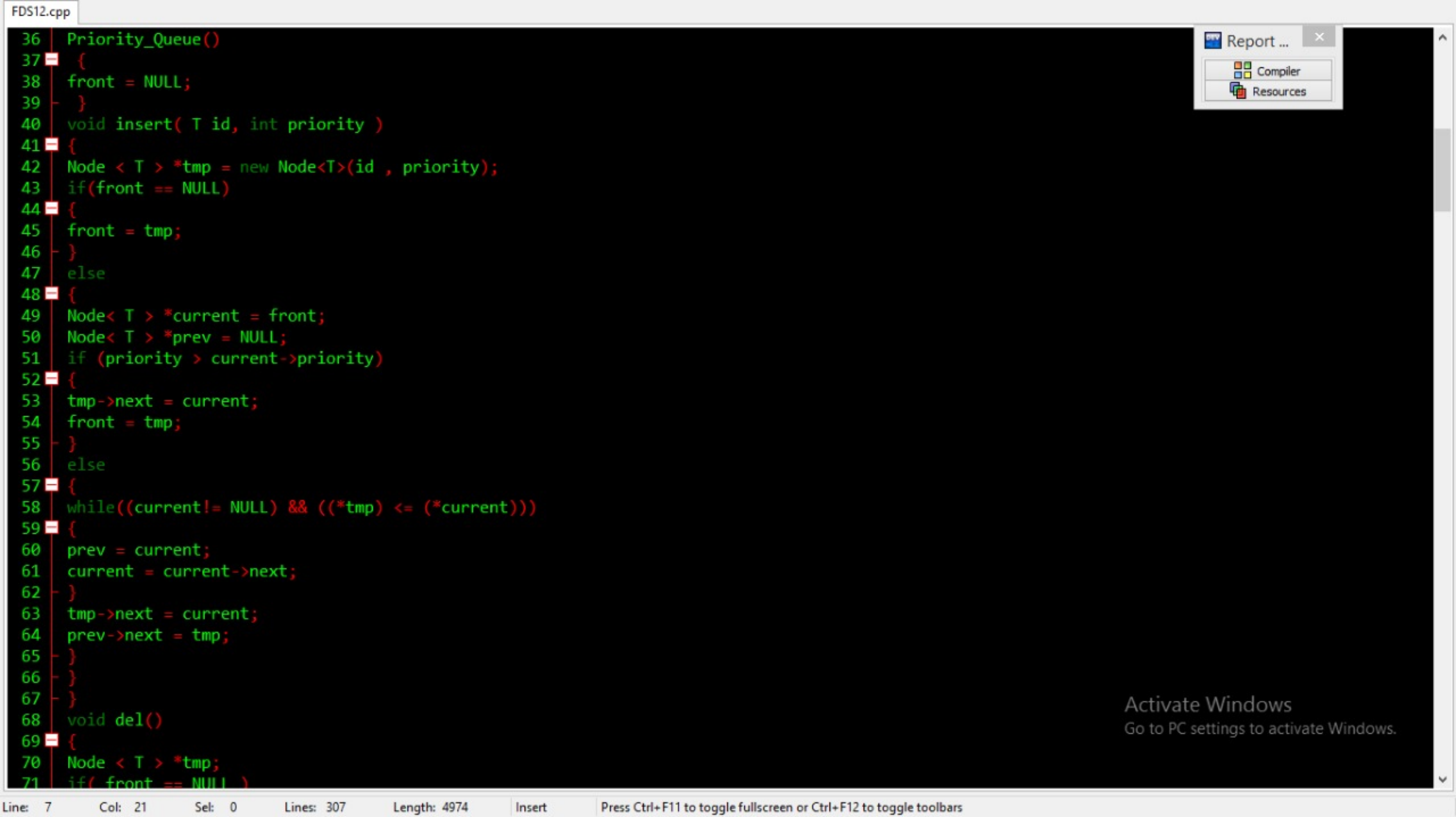
Conclusion-:

We learnt to implement inseration an deletion of nodes by prorities.



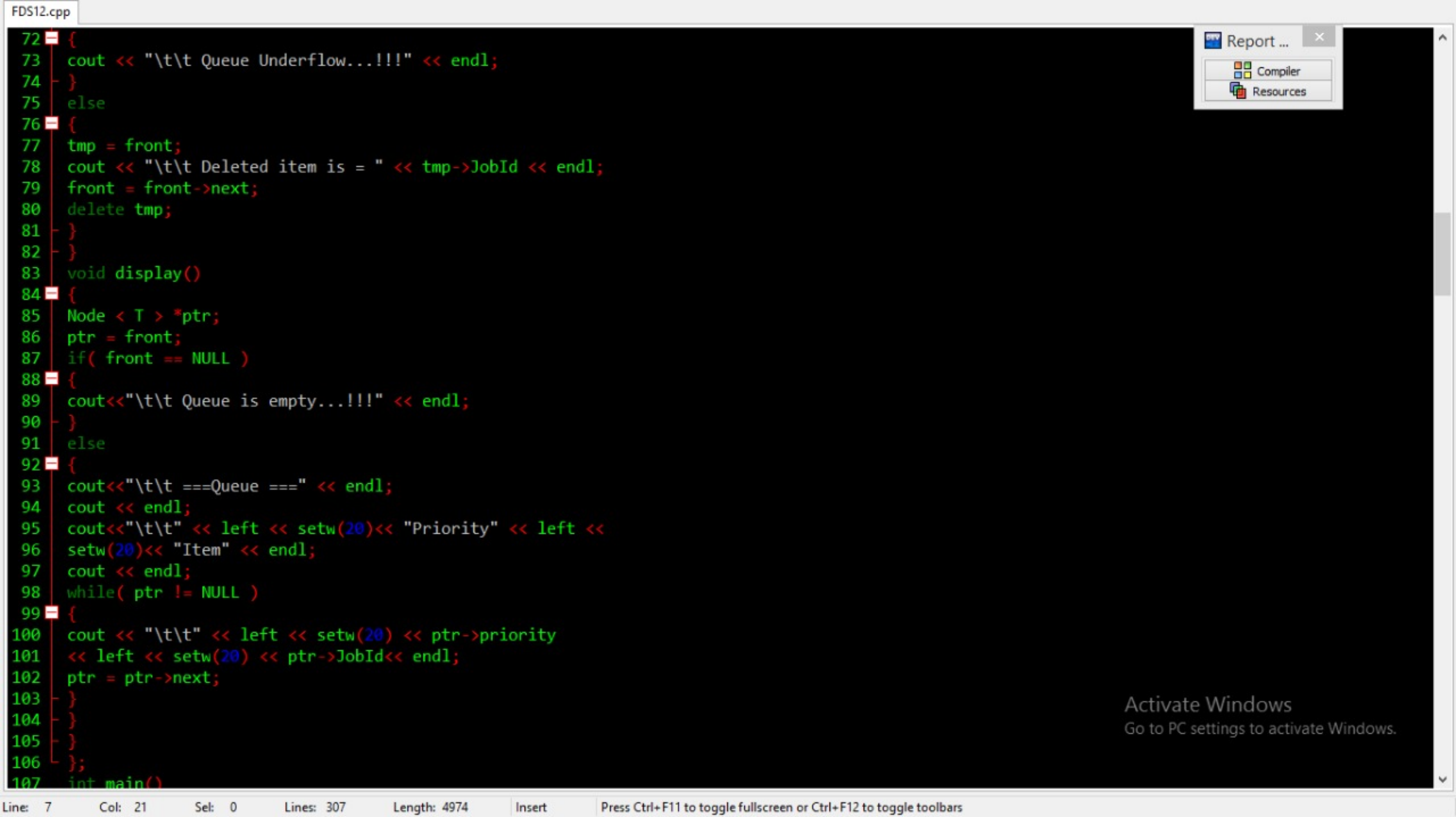
```cpp
#include <iostream>
#include <iomanip>
using namespace std;
template < class T >
class Node
{
template < class K >
friend class Priority_Queue;
private:
T JobId;
int priority;
Node < T >* next;
public:
Node()
{
next = NULL;
priority = -1;
}
Node( T id , int prio)
{
next = NULL;
JobId = id;
priority = prio;
}
bool operator <= ( Node &obj)
{
return (this->priority <= obj.priority);
}
};
template < class T >
class Priority_Queue
{
private:
Node < T >* front;
public:
Priority_Queue()
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars



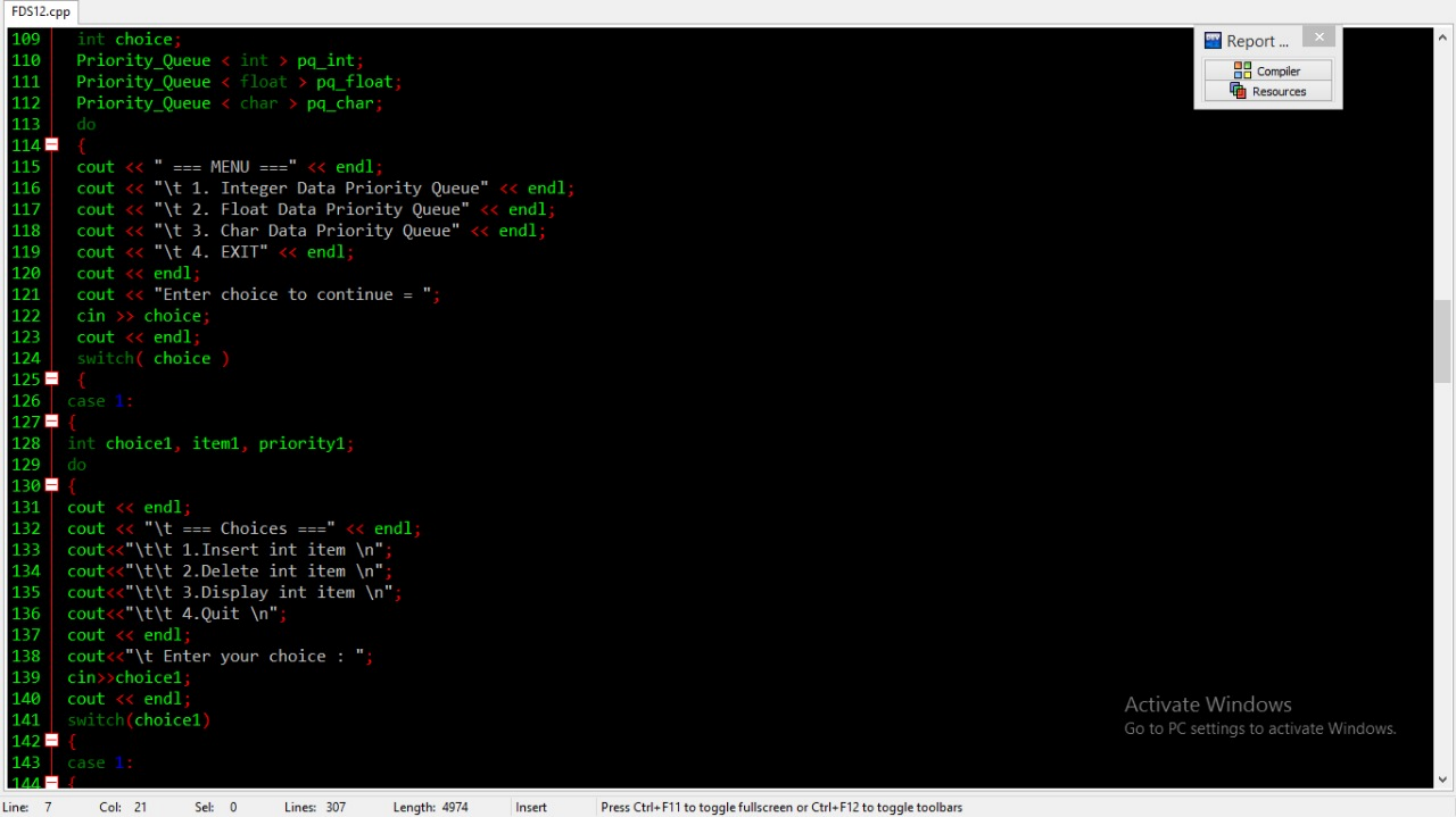
```
Priority_Queue()
 {
front = NULL;
 }
void insert( T id, int priority )
{
Node < T > *tmp = new Node<T>(id , priority);
if(front == NULL)
{
front = tmp;
}
else
{
Node< T > *current = front;
Node< T > *prev = NULL;
if (priority > current->priority)
{
tmp->next = current;
front = tmp;
}
else
{
while((current!= NULL) && ((*tmp) <= (*current)))
{
prev = current;
current = current->next;
}
tmp->next = current;
prev->next = tmp;
}
}
}
void del()
{
Node < T > *tmp;
if( front == NULL )
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

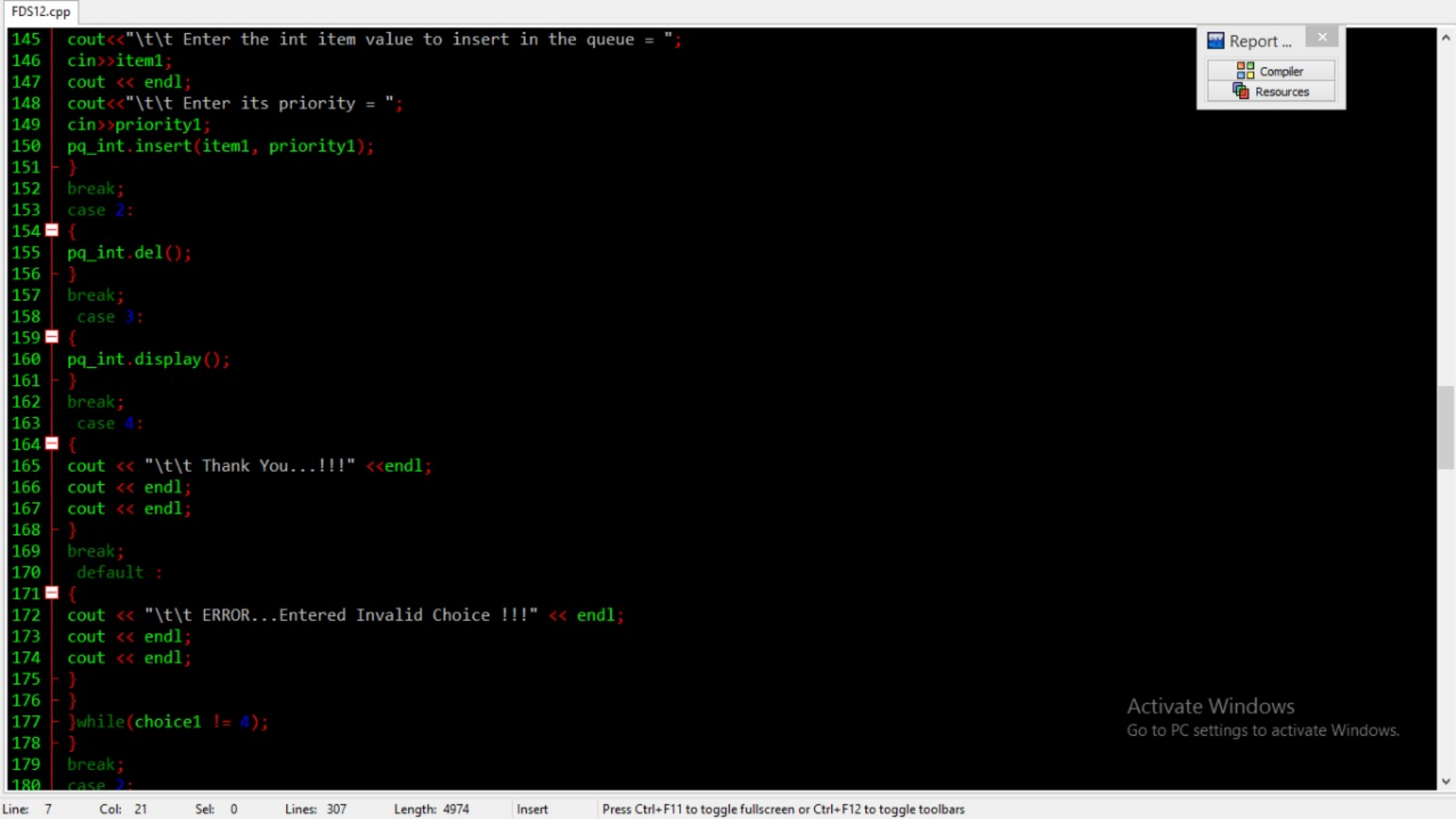


```cpp
{
cout << "\t\t Queue Underflow...!!!" << endl;
}
else
{
tmp = front;
cout << "\t\t Deleted item is = " << tmp->JobId << endl;
front = front->next;
delete tmp;
}
}
void display()
{
Node < T > *ptr;
ptr = front;
if( front == NULL )
{
cout<<"\t\t Queue is empty...!!!" << endl;
}
else
{
cout<<"\t\t ===Queue ===" << endl;
cout << endl;
cout<<"\t\t" << left << setw(20)<< "Priority" << left <<
setw(20)<< "Item" << endl;
cout << endl;
while( ptr != NULL )
{
cout << "\t\t" << left << setw(20) << ptr->priority
<< left << setw(20) << ptr->JobId<< endl;
ptr = ptr->next;
}
}
}
};
int main()
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

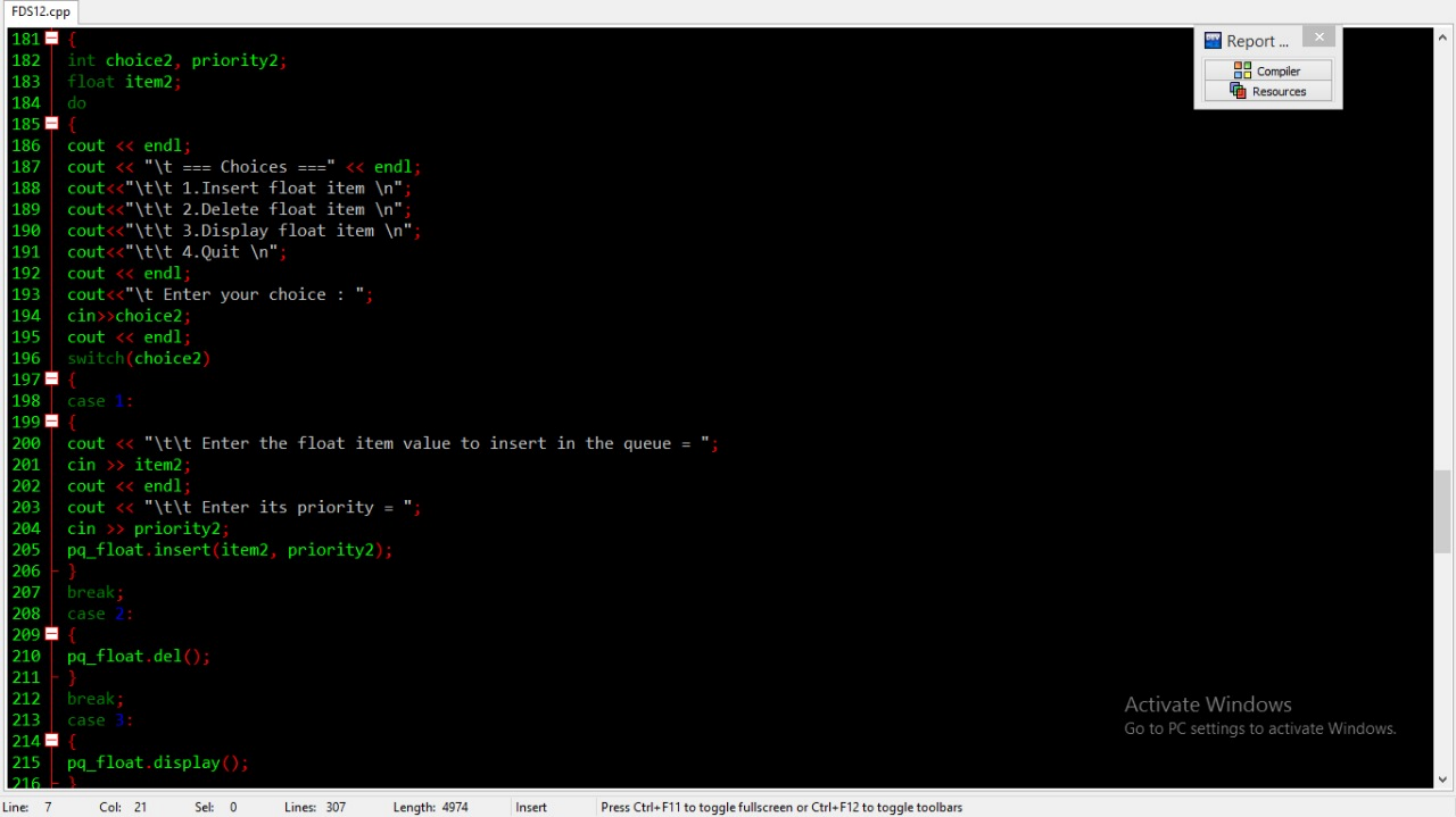

FDS12.cpp
Report ...
Compiler
Resources
Activate Windows
Go to PC settings to activate Windows.
Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

```
 int choice;
 Priority_Queue < int > pq_int;
 Priority_Queue < float > pq_float;
 Priority_Queue < char > pq_char;
 do
 {
 cout << " === MENU ===" << endl;
 cout << "\t 1. Integer Data Priority Queue" << endl;
 cout << "\t 2. Float Data Priority Queue" << endl;
 cout << "\t 3. Char Data Priority Queue" << endl;
 cout << "\t 4. EXIT" << endl;
 cout << endl;
 cout << "Enter choice to continue = ";
 cin >> choice;
 cout << endl;
 switch( choice )
 {
case 1:
{
int choice1, item1, priority1;
do
{
cout << endl;
cout << "\t === Choices ===" << endl;
cout<<"\t\t 1.Insert int item \n";
cout<<"\t\t 2.Delete int item \n";
cout<<"\t\t 3.Display int item \n";
cout<<"\t\t 4.Quit \n";
cout << endl;
cout<<"\t Enter your choice : ";
cin>>choice1;
cout << endl;
switch(choice1)
{
case 1:
{
```



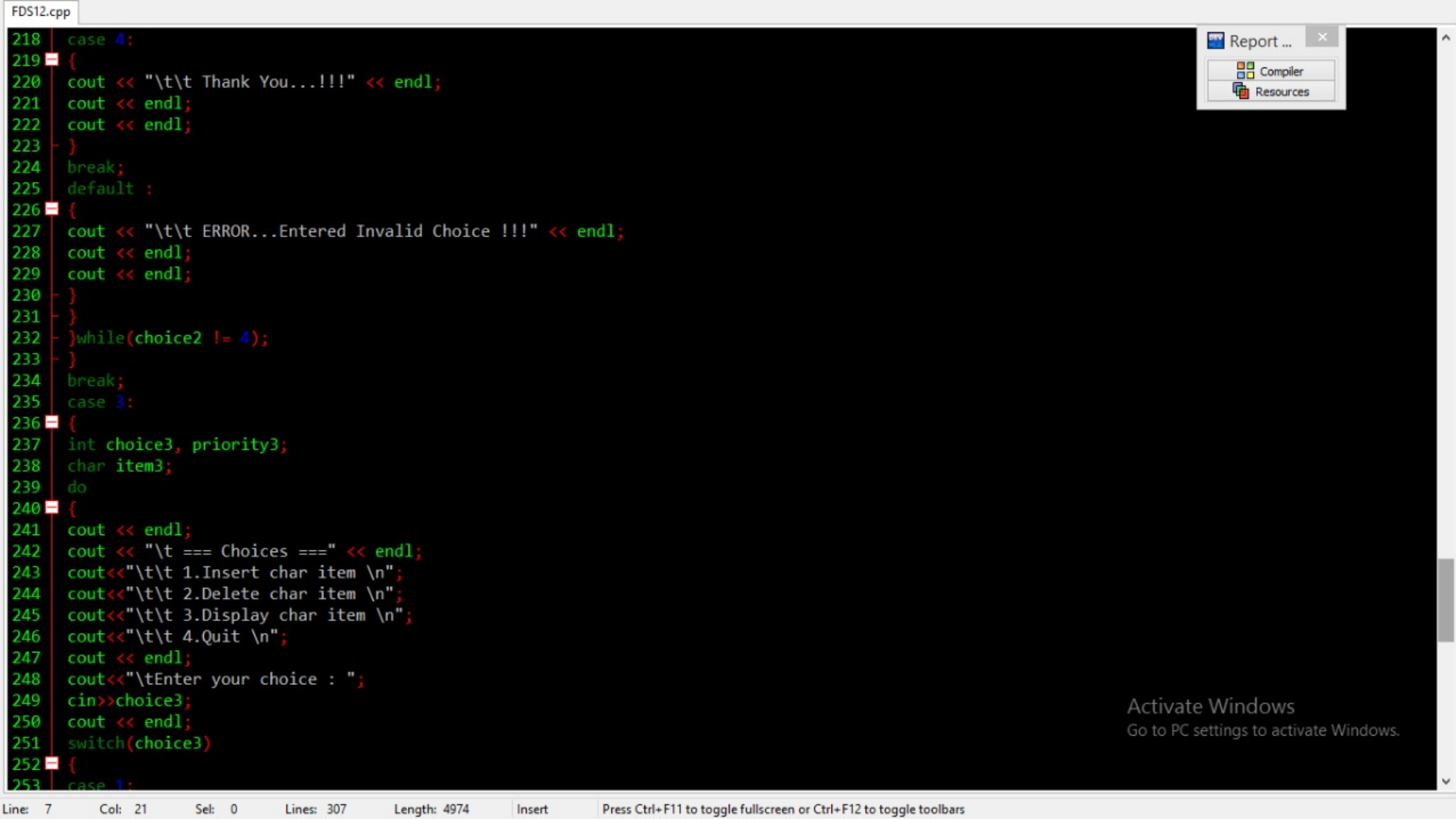
```
cout<<"\t\t Enter the int item value to insert in the queue = ";
cin>>item1;
cout << endl;
cout<<"\t\t Enter its priority = ";
cin>>priority1;
pq_int.insert(item1, priority1);
}
break;
case 2:
{
pq_int.del();
}
break;
 case 3:
{
pq_int.display();
}
break;
 case 4:
{
cout << "\t\t Thank You...!!!" <<endl;
cout << endl;
cout << endl;
}
break;
 default :
{
cout << "\t\t ERROR...Entered Invalid Choice !!!" << endl;
cout << endl;
cout << endl;
}
}
}while(choice1 != 4);
}
break;
case 2:
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

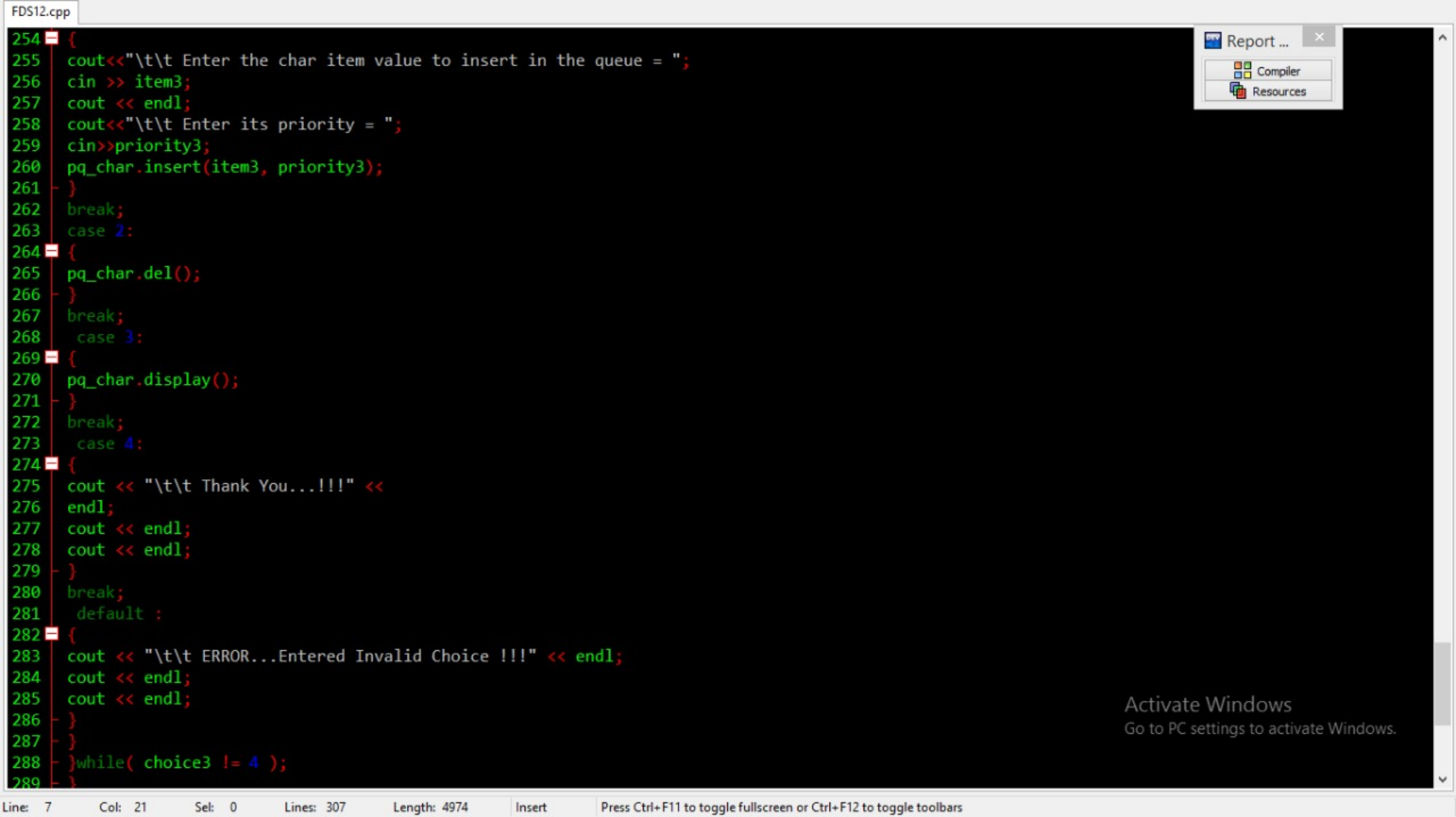


```cpp
{
int choice2, priority2;
float item2;
do
{
cout << endl;
cout << "\t === Choices ===" << endl;
cout<<"\t\t 1.Insert float item \n";
cout<<"\t\t 2.Delete float item \n";
cout<<"\t\t 3.Display float item \n";
cout<<"\t\t 4.Quit \n";
cout << endl;
cout<<"\t Enter your choice : ";
cin>>choice2;
cout << endl;
switch(choice2)
{
case 1:
{
cout << "\t\t Enter the float item value to insert in the queue = ";
cin >> item2;
cout << endl;
cout << "\t\t Enter its priority = ";
cin >> priority2;
pq_float.insert(item2, priority2);
}
break;
case 2:
{
pq_float.del();
}
break;
case 3:
{
pq_float.display();
}
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

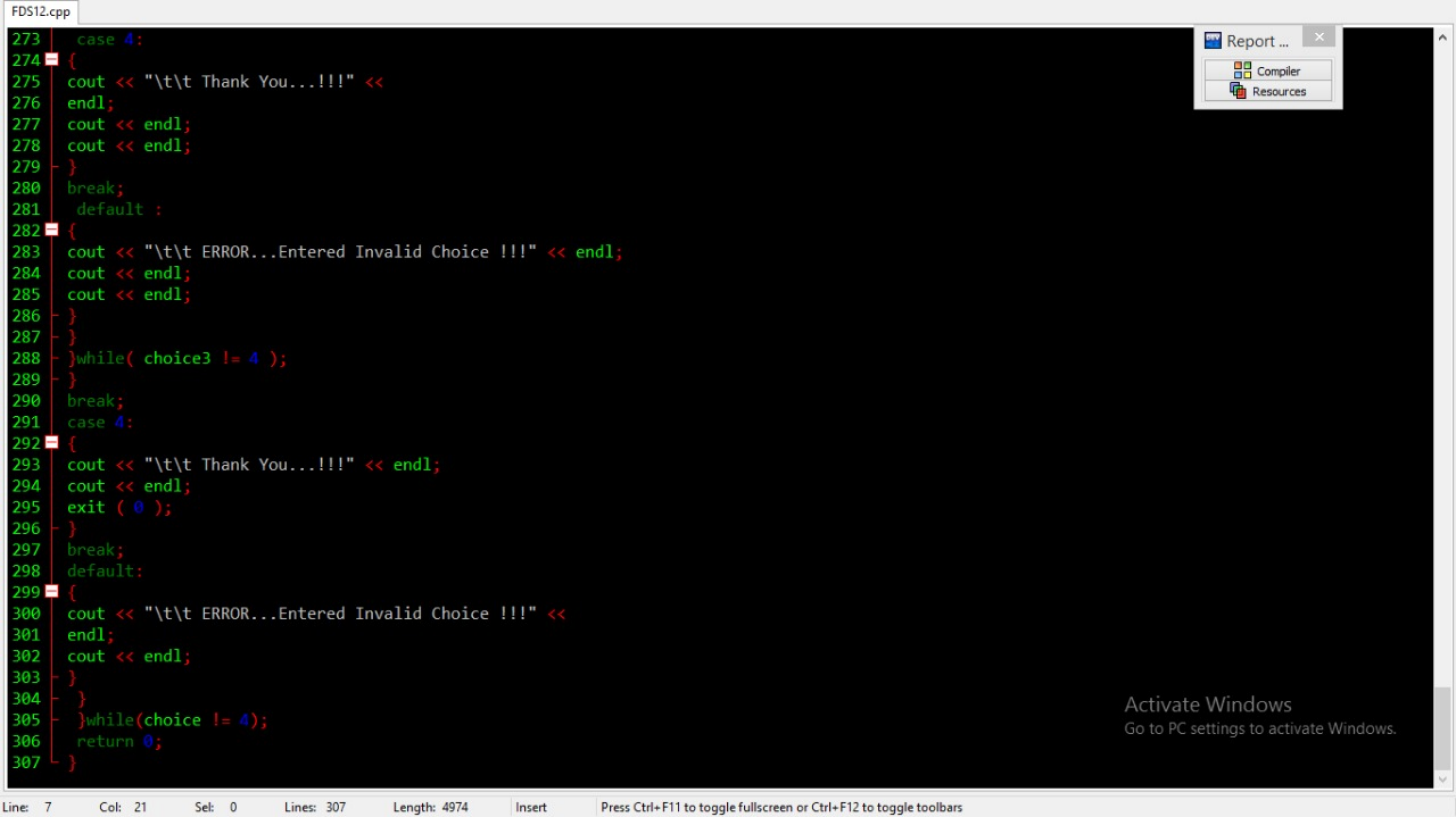

FDS12.cpp
Report ...
Compiler
Resources
Activate Windows
Go to PC settings to activate Windows.
Line: 7
Col: 21
Sel: 0
Lines: 307
Length: 4974
Insert
Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

```cpp
case 4:
{
cout << "\t\t Thank You...!!!" << endl;
cout << endl;
cout << endl;
}
break;
default :
{
cout << "\t\t ERROR...Entered Invalid Choice !!!" << endl;
cout << endl;
cout << endl;
}
}
}while(choice2 != 4);
}
break;
case 3:
{
int choice3, priority3;
char item3;
do
{
cout << endl;
cout << "\t === Choices ===" << endl;
cout<<"\t\t 1.Insert char item \n";
cout<<"\t\t 2.Delete char item \n";
cout<<"\t\t 3.Display char item \n";
cout<<"\t\t 4.Quit \n";
cout << endl;
cout<<"\tEnter your choice : ";
cin>>choice3;
cout << endl;
switch(choice3)
{
case 1:
```



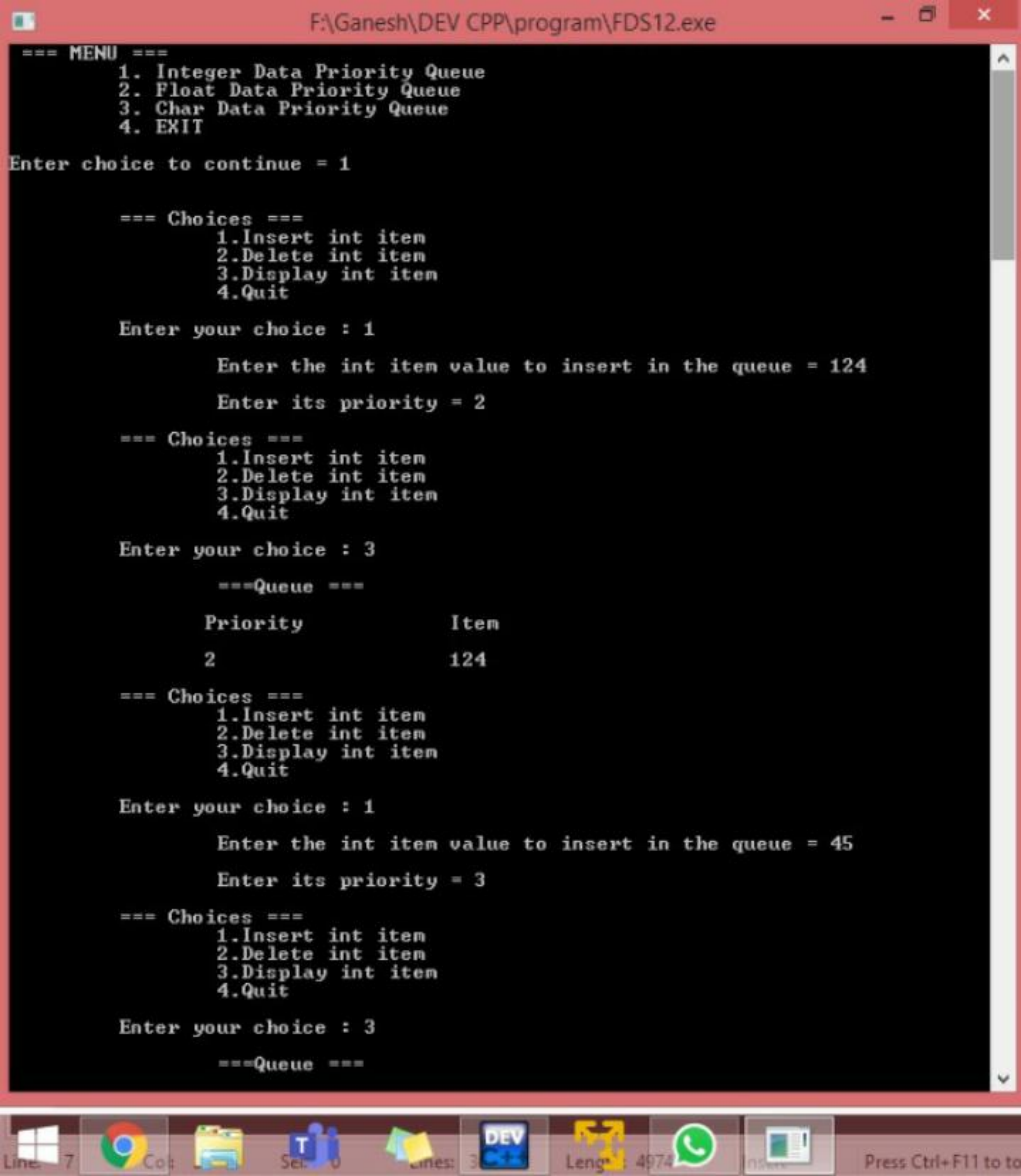
```cpp
{
cout<<"\t\t Enter the char item value to insert in the queue = ";
cin >> item3;
cout << endl;
cout<<"\t\t Enter its priority = ";
cin>>priority3;
pq_char.insert(item3, priority3);
}
break;
case 2:
{
pq_char.del();
}
break;
 case 3:
{
pq_char.display();
}
break;
 case 4:
{
cout << "\t\t Thank You...!!!" <<
endl;
cout << endl;
cout << endl;
}
break;
 default :
{
cout << "\t\t ERROR...Entered Invalid Choice !!!" << endl;
cout << endl;
cout << endl;
}
}
}while( choice3 != 4 );
}
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars

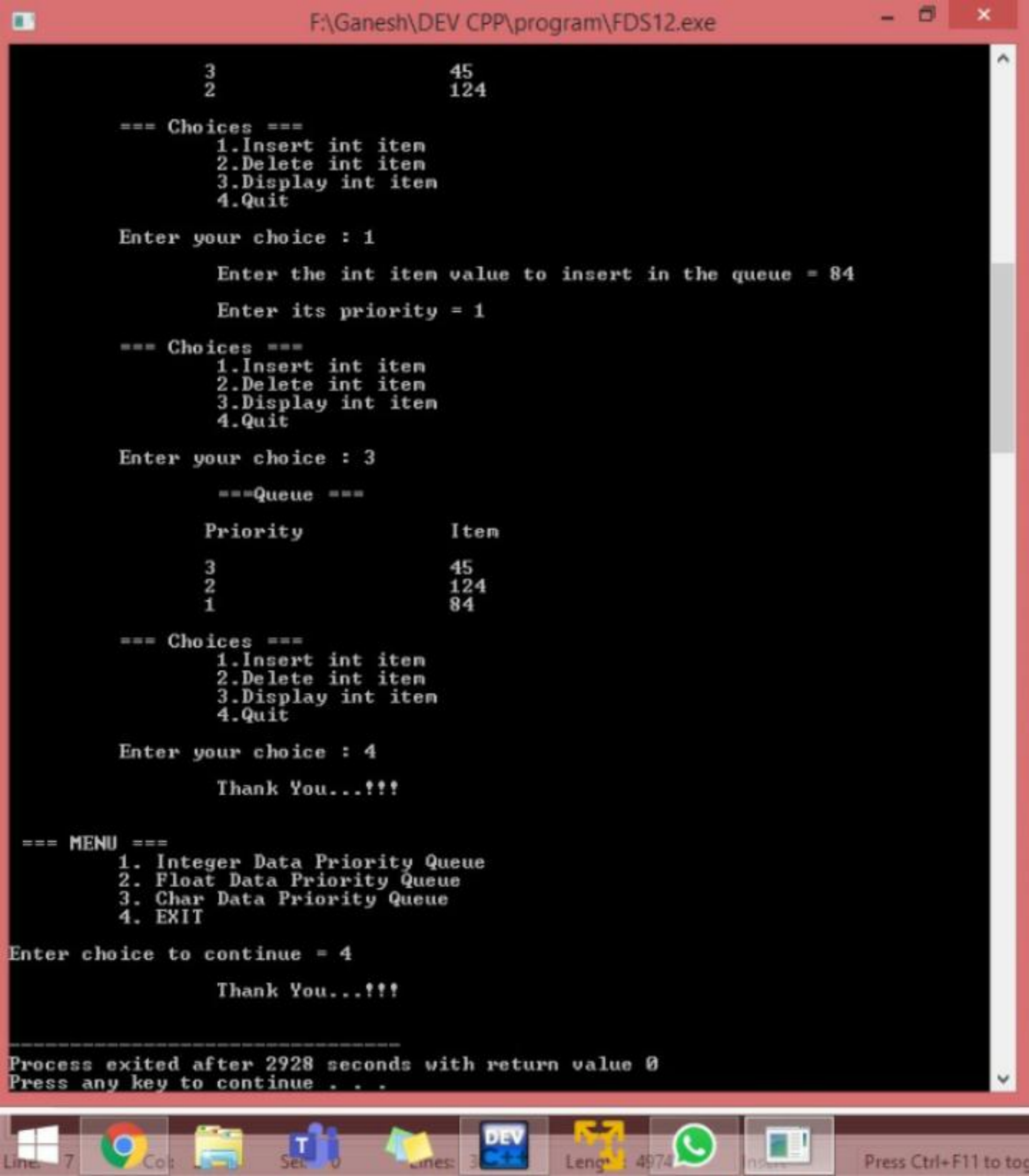

FDS12.cpp
Report ...
Compiler
Resources

```
 case 4:
{
cout << "\t\t Thank You...!!!" <<
endl;
cout << endl;
cout << endl;
}
break;
 default :
{
cout << "\t\t ERROR...Entered Invalid Choice !!!" << endl;
cout << endl;
cout << endl;
}
}
}while( choice3 != 4 );
}
break;
case 4:
{
cout << "\t\t Thank You...!!!" << endl;
cout << endl;
exit ( 0 );
}
break;
default:
{
cout << "\t\t ERROR...Entered Invalid Choice !!!" <<
endl;
cout << endl;
}
 }
 }while(choice != 4);
 return 0;
}
```

Line: 7 Col: 21 Sel: 0 Lines: 307 Length: 4974 Insert Press Ctrl+F11 to toggle fullscreen or Ctrl+F12 to toggle toolbars



```
 === MENU ===
         1. Integer Data Priority Queue
         2. Float Data Priority Queue
         3. Char Data Priority Queue
         4. EXIT

Enter choice to continue = 1


         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 1

                 Enter the int item value to insert in the queue = 124

                 Enter its priority = 2

         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 3

                 ===Queue ===

                Priority                Item

                2                       124

         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 1

                 Enter the int item value to insert in the queue = 45

                 Enter its priority = 3

         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 3

                 ===Queue ===
```



```
                3                   45
                2                   124

         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 1

                 Enter the int item value to insert in the queue = 84

                 Enter its priority = 1

         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 3

                 ===Queue ===

                Priority            Item

                3                   45
                2                   124
                1                   84

         === Choices ===
                 1.Insert int item
                 2.Delete int item
                 3.Display int item
                 4.Quit

         Enter your choice : 4

                 Thank You...!!!


 === MENU ===
         1. Integer Data Priority Queue
         2. Float Data Priority Queue
         3. Char Data Priority Queue
         4. EXIT
Enter choice to continue = 4

                 Thank You...!!!


--------------------------------
Process exited after 2928 seconds with return value 0
Press any key to continue . . .
```