

Fadhil Muhamad Pratama 1301213126

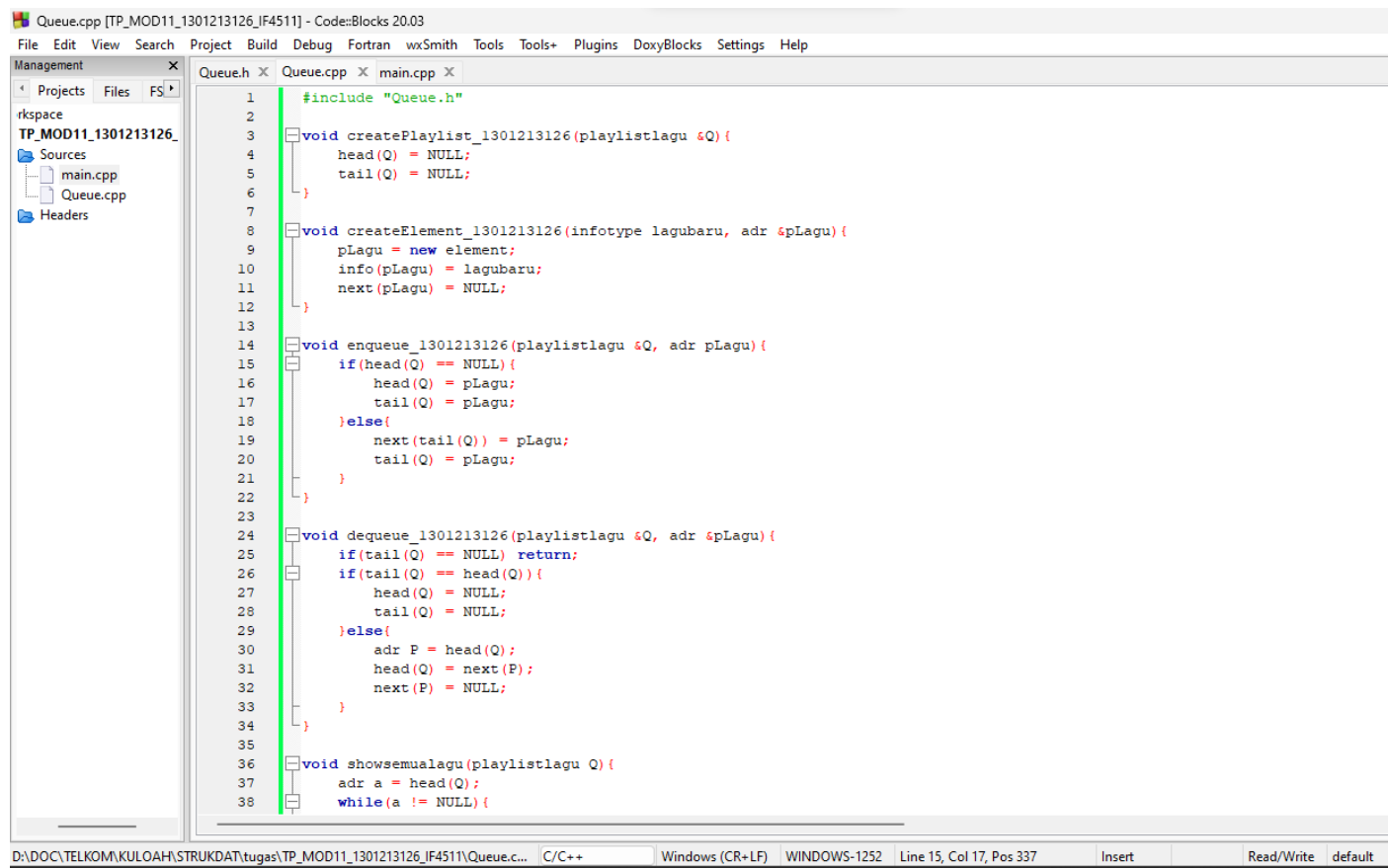
IF4511

## Queue.h

```
1  #ifndef QUEUE_H_INCLUDED
2  #define QUEUE_H_INCLUDED
3
4  #include <iostream>
5  using namespace std;
6
7  #define info(Q) (Q)->info
8  #define next(Q) (Q)->next
9  #define head(Q) (Q).head
10 #define tail(Q) (Q).tail
11
12 struct infotype{
13     string artis;
14     string judul;
15 };
16 typedef struct element *adr;
17
18 struct element{
19     infotype info;
20     adr next;
21 };
22
23 struct playlistlagu {
24     adr head;
25     adr tail;
26 };
27
28 void createPlaylist_1301213126(playlistlagu &Q);
29 void createElement_1301213126(infotype lagubaru, adr &pLagu);
30 void enqueue_1301213126(playlistlagu &Q, adr pLagu);
31 void dequeue_1301213126(playlistlagu &Q, adr &pLagu);
32 void showsemualagu(playlistlagu Q);
33
34 #endif // QUEUE_H_INCLUDED
35
```

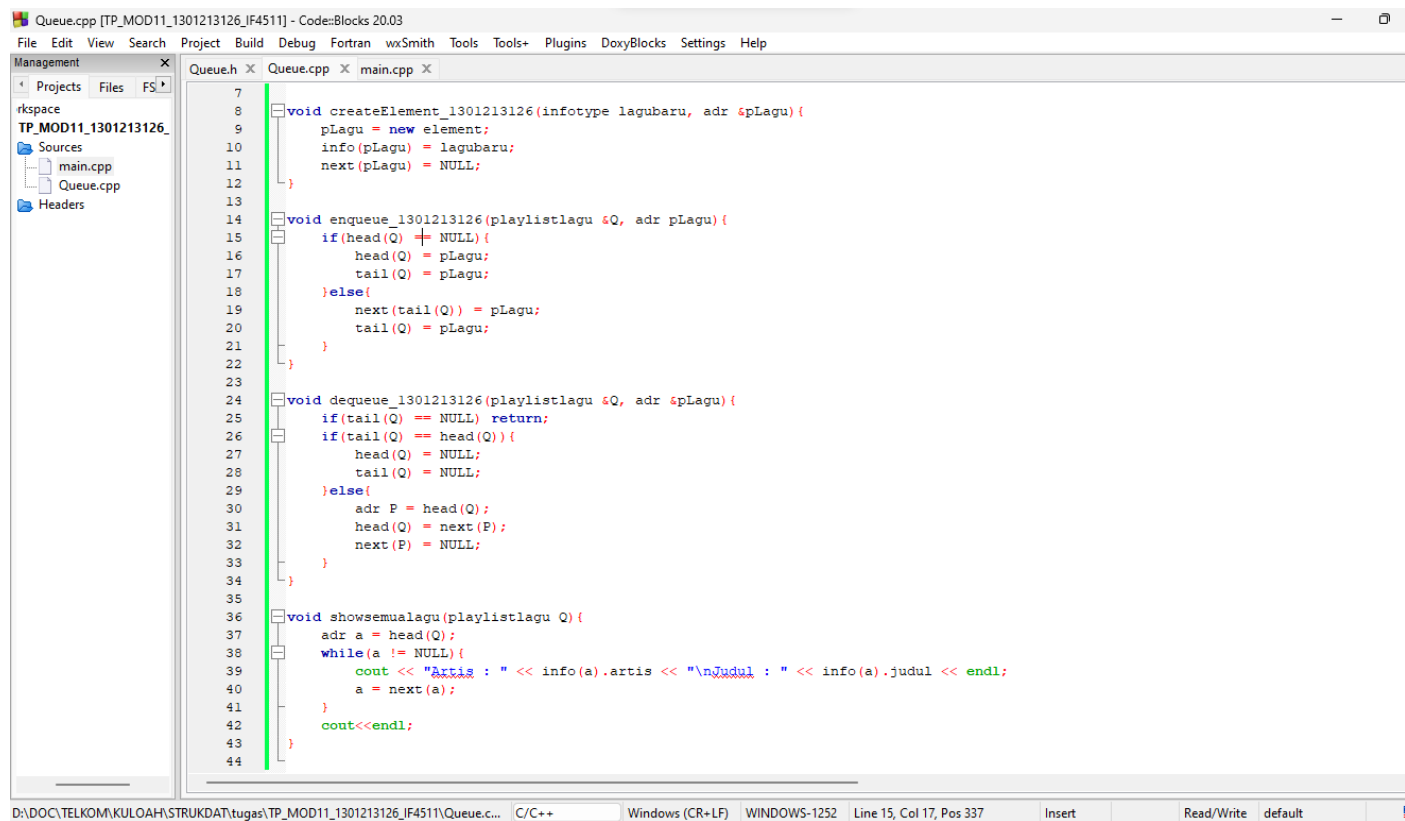
D:\DOC\TELKOM\KULOAH\STRUKDAT\tugas\TP\_MOD11\_1301213126\_IF4511\Queue.h C/C++ Windows (CR+LF) WINDOWS-1252 Line 31, Col 47, Pos 629 Insert

## Queue.cpp



The screenshot shows the Code::Blocks IDE with the file Queue.cpp open. The left sidebar displays the project structure for 'TP\_MOD11\_1301213126', including 'Sources' (main.cpp, Queue.cpp) and 'Headers'. The main editor window shows the implementation of a queue using a linked list. The code includes a header file 'Queue.h' and defines several functions: createPlaylist, createElement, enqueue, dequeue, and showsemualagu. The enqueue function handles the case where the queue is empty (head and tail are NULL) and the case where it is not empty. The dequeue function handles the case where the queue is empty (head and tail are NULL) and the case where it is not empty. The showsemualagu function displays the elements of the queue.

```
1 #include "Queue.h"
2
3 void createPlaylist_1301213126(playlistlagu &Q){
4     head(Q) = NULL;
5     tail(Q) = NULL;
6 }
7
8 void createElement_1301213126(infotype lagubaru, adr &pLagu){
9     pLagu = new element;
10    info(pLagu) = lagubaru;
11    next(pLagu) = NULL;
12 }
13
14 void enqueue_1301213126(playlistlagu &Q, adr pLagu){
15     if(head(Q) == NULL){
16         head(Q) = pLagu;
17         tail(Q) = pLagu;
18     }else{
19         next(tail(Q)) = pLagu;
20         tail(Q) = pLagu;
21     }
22 }
23
24 void dequeue_1301213126(playlistlagu &Q, adr &pLagu){
25     if(tail(Q) == NULL) return;
26     if(tail(Q) == head(Q)){
27         head(Q) = NULL;
28         tail(Q) = NULL;
29     }else{
30         adr P = head(Q);
31         head(Q) = next(P);
32         next(P) = NULL;
33     }
34 }
35
36 void showsemualagu(playlistlagu Q){
37     adr a = head(Q);
38     while(a != NULL){
```



The screenshot shows the Code::Blocks IDE with the file Queue.cpp open. The left sidebar displays the project structure for 'TP\_MOD11\_1301213126', including 'Sources' (main.cpp, Queue.cpp) and 'Headers'. The main editor window shows the implementation of a queue using a linked list. The code includes a header file 'Queue.h' and defines several functions: createElement, enqueue, dequeue, and showsemualagu. The enqueue function handles the case where the queue is empty (head and tail are NULL) and the case where it is not empty. The dequeue function handles the case where the queue is empty (head and tail are NULL) and the case where it is not empty. The showsemualagu function displays the elements of the queue using cout statements.

```
7 void createElement_1301213126(infotype lagubaru, adr &pLagu){
8     pLagu = new element;
9     info(pLagu) = lagubaru;
10    next(pLagu) = NULL;
11 }
12
13 void enqueue_1301213126(playlistlagu &Q, adr pLagu){
14     if(head(Q) == NULL){
15         head(Q) = pLagu;
16         tail(Q) = pLagu;
17     }else{
18         next(tail(Q)) = pLagu;
19         tail(Q) = pLagu;
20     }
21 }
22
23 void dequeue_1301213126(playlistlagu &Q, adr &pLagu){
24     if(tail(Q) == NULL) return;
25     if(tail(Q) == head(Q)){
26         head(Q) = NULL;
27         tail(Q) = NULL;
28     }else{
29         adr P = head(Q);
30         head(Q) = next(P);
31         next(P) = NULL;
32     }
33 }
34
35 void showsemualagu(playlistlagu Q){
36     adr a = head(Q);
37     while(a != NULL){
38         cout << "Artist : " << info(a).artis << "\nJudul : " << info(a).judul << endl;
39         a = next(a);
40     }
41     cout<<endl;
42 }
43
44 }
```

## Main.cpp



```
main.cpp [TP_MOD11_1301213126_JF4511] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
  Projects Files FS
  rkspace
  TP_MOD11_1301213126_
    Sources
      main.cpp
      Queue.cpp
    Headers
Queue.h Queue.cpp main.cpp
4 using namespace std;
5
6 int main()
7 {
8     playlistlagu Q;
9     createPlaylist_1301213126(Q);
10    infotype x;
11    adr P, K;
12    x.artis = "Frank Ocean";
13    x.judul = "Ivy";
14    createElement_1301213126(x, P);
15    enqueue_1301213126(Q,P);
16
17    x.artis = "The Weeknd";
18    x.judul = "Starboy";
19    createElement_1301213126(x, P);
20    enqueue_1301213126(Q,P);
21
22    x.artis = "Neckdeep";
23    x.judul = "Wish you were here";
24    createElement_1301213126(x, P);
25    enqueue_1301213126(Q,P);
26
27    x.artis = "Daniel Caesar";
28    x.judul = "Japanese Denim";
29    createElement_1301213126(x, P);
30    enqueue_1301213126(Q,P);
31
32    x.artis = "Rex Orange County";
33    x.judul = "Pluto Projector";
34    createElement_1301213126(x, P);
35    enqueue_1301213126(Q,P);
36
37    x.artis = "Simple Plan";
38    x.judul = "i'm just a kid";
39    createElement_1301213126(x, P);
40    enqueue_1301213126(Q,P);
41
D:\DOC\TELKOM\KULOAH\STRUKDAT\tugas\TP_MOD11_1301213126_JF4511\main.cpp C/C++ Windows (CR+LF) WINDOWS-1252 Line 6, Col 1, Pos 67 Insert Read/Write default
```



```
main.cpp [TP_MOD11_1301213126_JF4511] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
  Projects Files FS
  rkspace
  TP_MOD11_1301213126_
    Sources
      main.cpp
      Queue.cpp
    Headers
Queue.h Queue.cpp main.cpp
39 createElement_1301213126(x, P);
40 enqueue_1301213126(Q,P);
41
42 x.artis = "Bring Me The Horizon";
43 x.judul = "Throne";
44 createElement_1301213126(x, P);
45 enqueue_1301213126(Q,P);
46
47 x.artis = "The Weeknd";
48 x.judul = "The Hills";
49 createElement_1301213126(x, P);
50 enqueue_1301213126(Q,P);
51
52 x.artis = "Frank Ocean";
53 x.judul = "Lost";
54 createElement_1301213126(x, P);
55 enqueue_1301213126(Q,P);
56
57 x.artis = "Reality Club";
58 x.judul = "Telenovia";
59 createElement_1301213126(x, P);
60 enqueue_1301213126(Q,P);
61
62 x.artis = "Boy Pablo";
63 x.judul = "Sick Feeling";
64 createElement_1301213126(x, P);
65 enqueue_1301213126(Q,P);
66
67 x.artis = "Kanye West";
68 x.judul = "Heartless";
69 createElement_1301213126(x, P);
70 enqueue_1301213126(Q,P);
71
72 x.artis = "Joli";
73 x.judul = "Sanctuary";
74 createElement_1301213126(x, P);
75 enqueue_1301213126(Q,P);
76
D:\DOC\TELKOM\KULOAH\STRUKDAT\tugas\TP_MOD11_1301213126_JF4511\main.cpp C/C++ Windows (CR+LF) WINDOWS-1252 Line 6, Col 1, Pos 67 Insert Read/Write default
```

The screenshot shows a C++ IDE with a project named 'TP\_MOD11\_1301213126'. The main.cpp file contains a queue implementation. The queue is initialized with 15 elements, each representing an artist and their album. The elements are: Kanye West (Heartless), Joji (Sanctuary), Arctic Monkeys (R U Mine?), Cigarettes After Sex (Sweet), Khalid (American Teen), and The Smiths (Please, Please, Please, Let Me Get What I Want). The queue is implemented using an array and a pointer to the front element. The main function calls enqueue, createElement, dequeue, and showsemualagu functions to demonstrate the queue operations.

```
65 enqueue_1301213126(Q,P);
66
67 x.artis = "Kanye West";
68 x.judul = "Heartless";
69 createElement_1301213126(x, P);
70 enqueue_1301213126(Q,P);
71
72 x.artis = "Joji";
73 x.judul = "Sanctuary";
74 createElement_1301213126(x, P);
75 enqueue_1301213126(Q,P);
76
77 x.artis = "Arctic Monkeys";
78 x.judul = "R U Mine?";
79 createElement_1301213126(x, P);
80 enqueue_1301213126(Q,P);
81
82 x.artis = "Cigarettes After Sex";
83 x.judul = "Sweet";
84 createElement_1301213126(x, P);
85 enqueue_1301213126(Q,P);
86
87 x.artis = "Khalid";
88 x.judul = "American Teen";
89 createElement_1301213126(x, P);
90 enqueue_1301213126(Q,P);
91
92 x.artis = "The Smiths";
93 x.judul = "Please, Please, Please, Let Me Get What I Want";
94 createElement_1301213126(x, P);
95 enqueue_1301213126(Q,P);
96 showsemualagu(Q);
97 dequeue_1301213126(Q,K);
98 showsemualagu(Q);
99
100 return 0;
101 }
102
```

## Output

The terminal window shows the output of the C++ program. It displays the artist names and album titles for each element in the queue, followed by the execution time and a prompt to press any key to continue.

```
Judul : Pluto Projector
Artis : Simple Plan
Judul : i'm just a kid
Artis : Bring Me The Horizon
Judul : Throne
Artis : The Weeknd
Judul : The Hills
Artis : Frank Ocean
Judul : Lost
Artis : Reality Club
Judul : Telenovia
Artis : Boy Pablo
Judul : Sick Feeling
Artis : Kanye West
Judul : Heartless
Artis : Joji
Judul : Sanctuary
Artis : Arctic Monkeys
Judul : R U Mine?
Artis : Cigarettes After Sex
Judul : Sweet
Artis : Khalid
Judul : American Teen
Artis : The Smiths
Judul : Please, Please, Please, Let Me Get What I Want

Process returned 0 (0x0)   execution time : 0.033 s
Press any key to continue.
```