

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  const int MAX = 12;
6
7  struct Buku {
8      string judul;
9      string penulis;
10 };
11
12 struct RakBuku {
13     Buku T[MAX];
14     int Top;
15 };
16
17 void createStack(RakBuku &S) {
18     S.Top = 0;
19 }
20
21
22 void Push(RakBuku &S, string judul, string penulis) {
23     if (isFull(S)) {
24         cout << "Rak penuh!" << endl;
25     } else {
26         S.Top++;
27         S.T[S.Top].judul = judul;
28         S.T[S.Top].penulis = penulis;
29     }
30 }
31
32 void Pop(RakBuku &S, string &judul, string &penulis) {
33     if (isEmpty(S)) {
34         cout << "Rak kosong!" << endl;
35     } else {
36         judul = S.T[S.Top].judul;
37         penulis = S.T[S.Top].penulis;
38         S.Top--;
39     }
40 }
41
42 void Get(RakBuku &S, string judulCari, string &namaPenulis) {
43     bool ditemukan = false;
44
45     if (isEmpty(S)) {
46         cout << "Rak kosong!" << endl;
47         namaPenulis = "";
48     } else {
49         while (!isEmpty(S) && !ditemukan) {
50             if (S.T[S.Top].judul == judulCari) {
51                 namaPenulis = S.T[S.Top].penulis;
52                 S.Top--;
53                 ditemukan = true;
54             } else {
55                 S.Top--;
56             }
57         }
58
59         if (!ditemukan) {
60             cout << "Buku dengan judul '" << judulCari << "' tidak ditemukan." << endl;
61             namaPenulis = "";
62         }
63     }
64 }
65

```

```

66 void tampilRak(RakBuku S) {
67     if (isEmpty(S)) {
68         cout << "Rak kosong." << endl;
69     } else {
70         cout << "\nIsi rak buku (dari atas ke bawah):" << endl;
71         for (int i = S.Top; i >= 1; i--) {
72             cout << i << ". " << S.T[i].judul << " - " << S.T[i].penulis << endl;
73         }
74     }
75 }
76

77 int main() {
78     RakBuku rak;
79     createStack(rak);
80
81     Push(rak, "Dilan", "Pidi Baiq");
82     Push(rak, "Persuasion", "Jane Austen");
83     Push(rak, "The Alchemist", "Paulo Coelho");
84     Push(rak, "Supernova: Kesatria, Putri, & Bintang Jatuh", "Dee Lestari");
85     Push(rak, "5cm", "Donny Dhigantoro");
86
87     tampilkanRak(rak);
88
89     string judul, penulis;
90     Pop(rak, judul, penulis);
91     cout << "\nJudul buku yang diambil: " << judul << endl
92     << "Penulis: " << penulis << endl;
93
94     tampilkanRak(rak);
95
96     string cari = "The Alchemist";
97     string namaPenulis;
98     Get(rak, cari, namaPenulis);
99     if (!namaPenulis.empty()) {
100         cout << "\nPenulis buku '" << dicari << "' yang di get adalah: "
101         << namaPenulis << endl;
102     }
103
104     tampilkanRak(rak);
105
106     return 0;
107 }

```

```

PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack> cd "c:\shellyn\kuliah\semester3\struktur
rData\teori\tugasStack\" ; if ($?) { g++ stack.cpp -o stack } ; if ($?) { .\stack }
stack.cpp: In function 'void Push(RakBuku&, std::__cxx11::string, std::__cxx11::string)':
stack.cpp:23:17: error: 'isFull' was not declared in this scope
    if (isFull(S)) {
        ^
stack.cpp: In function 'void Pop(RakBuku&, std::__cxx11::string&, std::__cxx11::string&)':
stack.cpp:33:18: error: 'isEmpty' was not declared in this scope
    if (isEmpty(S)) {
        ^
stack.cpp: In function 'void Get(RakBuku&, std::__cxx11::string, std::__cxx11::string&)':
stack.cpp:45:18: error: 'isEmpty' was not declared in this scope
    if (isEmpty(S)) {
        ^
stack.cpp: In function 'void tampilRak(RakBuku)':
stack.cpp:67:18: error: 'isEmpty' was not declared in this scope
    if (isEmpty(S)) {
        ^
stack.cpp: In function 'int main()':
stack.cpp:87:21: error: 'tampilkanRak' was not declared in this scope
    tampilkanRak(rak);
    ^
stack.cpp:100:39: error: 'dicari' was not declared in this scope
    cout << "\nPenulis buku '" << dicari << "' yang di get adalah: "
                                ^~~~~~

```

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  const int MAX = 12;
6
7  struct Buku {
8      string judul;
9      string penulis;
10 };
11
12 struct RakBuku {
13     Buku T[MAX];
14     int Top;
15 };
16
17 void createStack(RakBuku &S) {
18     S.Top = 0;
19 }
20
21 bool isEmpty(RakBuku S) {
22     return S.Top == 0;
23 }
24
25 bool isFull(RakBuku S) {
26     return S.Top == MAX;
27 }
28
29 void Push(RakBuku &S, string judul, string penulis) {
30     if (isFull(S)) {
31         cout << "Rak penuh!" << endl;
32     } else {
33         S.Top++;
34         S.T[S.Top].judul = judul;
35         S.T[S.Top].penulis = penulis;
36     }
37 }
38
39 void Pop(RakBuku &S, string &judul, string &penulis) {
40     if (isEmpty(S)) {
41         cout << "Rak kosong!" << endl;
42     } else {
43         judul = S.T[S.Top].judul;
44         penulis = S.T[S.Top].penulis;
45         S.Top--;
46     }
47 }
48
49 void Get(RakBuku &S, string judulCari, string &namaPenulis) {
50     bool ditemukan = false;
51
52     if (isEmpty(S)) {
53         cout << "Rak kosong!" << endl;
54         namaPenulis = "";
55     } else {
56         while (!isEmpty(S) && !ditemukan) {
57             if (S.T[S.Top].judul == judulCari) {
58                 namaPenulis = S.T[S.Top].penulis;
59                 S.Top--;
60                 ditemukan = true;
61             } else {
62                 S.Top--;
63             }
64         }
65
66         if (!ditemukan) {
67             cout << "Buku dengan judul '" << judulCari << "' tidak ditemukan." << endl;
68             namaPenulis = "";
69         }
70     }
71 }
72

```

```

73 void tampilRak(RakBuku S) {
74     if (isEmpty(S)) {
75         cout << "Rak kosong." << endl;
76     } else {
77         cout << "\nIsi rak buku (dari atas ke bawah):" << endl;
78         for (int i = S.Top; i >= 1; i--) {
79             cout << i << ". " << S.T[i].judul << " - " << S.T[i].penulis << endl;
80         }
81     }
82 }
83
84 int main() {
85     RakBuku rak;
86     createStack(rak);
87
88     Push(rak, "Dilan", "Pidi Baiq");
89     Push(rak, "Persuasion", "Jane Austen");
90     Push(rak, "The Alchemist", "Paulo Coelho");
91     Push(rak, "Supernova: Kesatria, Putri, & Bintang Jatuh", "Dee Lestari");
92     Push(rak, "5cm", "Donny Dhingantoro");
93
94     tampilRak(rak);
95
96     string judul, penulis;
97     Pop(rak, judul, penulis);
98     cout << "\nJudul buku yang diambil: " << judul << endl;
99     << "Penulis: " << penulis << endl;
100
101     tampilRak(rak);
102
103     string dicari = "The Alchemist";
104     string namaPenulis;
105     Get(rak, dicari, namaPenulis);
106     if (!namaPenulis.empty()) {
107         cout << "\nPenulis buku '" << dicari << "' yang di get adalah: "
108         << namaPenulis << endl;
109     }
110
111     tampilRak(rak);
112
113     return 0;
114 }
115

```

PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack> cd "c:\shellyn\kuliah\semester3\strukturData\teori\tugasStack\" ; if (\$?) { g++ stack.cpp -o stack } ; if (\$?) { .\stack }

```

Isi rak buku (dari atas ke bawah):
5. 5cm - Donny Dhingantoro
4. Supernova: Kesatria, Putri, & Bintang Jatuh - Dee Lestari
3. The Alchemist - Paulo Coelho
2. Persuasion - Jane Austen
1. Dilan - Pidi Baiq

```

```

Judul buku yang diambil: 5cm
Penulis: Donny Dhingantoro

```

```

Isi rak buku (dari atas ke bawah):
4. Supernova: Kesatria, Putri, & Bintang Jatuh - Dee Lestari
3. The Alchemist - Paulo Coelho
2. Persuasion - Jane Austen
1. Dilan - Pidi Baiq

```

```

Penulis buku 'The Alchemist' yang di get adalah: Paulo Coelho

```

```

Isi rak buku (dari atas ke bawah):
2. Persuasion - Jane Austen
1. Dilan - Pidi Baiq
PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack>

```

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  const int MAX = 12;
6
7  struct Buku {
8      string judul;
9      string penulis;
10 };
11
12 struct RakBuku {
13     Buku T[MAX];
14     int head;
15     int tail;
16 };
17
18 void createQueue(RakBuku &Q) {
19     Q.head = 0;
20     Q.tail = 0;
21 }
22
23 bool isEmpty(RakBuku Q) {
24     return (Q.head == 0 && Q.tail == 0);
25 }
26
27 bool isFull(RakBuku Q) {
28     return (Q.tail == MAX);
29 }
30
31 void Enqueue(RakBuku &Q, string judul, string penulis) {
32     if (isFull(Q)) {
33         cout << "Rak penuh!" << endl;
34     } else if (isEmpty(Q)) {
35         Q.head = 1;
36         Q.tail = 1;
37         Q.T[Q.tail - 1].judul = judul;
38         Q.T[Q.tail - 1].penulis = penulis;
39     } else {
40         Q.tail++;
41         Q.T[Q.tail - 1].judul = judul;
42         Q.T[Q.tail - 1].penulis = penulis;
43     }
44 }
45
46 void Dequeue(RakBuku &Q, string &judul, string &penulis) {
47     if (isEmpty(Q)) {
48         cout << "Rak kosong!" << endl;
49     } else {
50         judul = Q.T[0].judul;
51         penulis = Q.T[0].penulis;
52
53         for (int i = 1; i < Q.tail; i++) {
54             Q.T[i - 1] = Q.T[i];
55         }
56
57         if (Q.tail == 0) Q.head = 0;
58     }
59 }
60

```

```

61 void tampilRak(RakBuku Q) {
62     if (isEmpty(Q)) {
63         cout << "Rak kosong." << endl;
64     } else {
65         cout << endl << "Isi rak buku:" << endl;
66         for (int i = Q.head - 1; i < Q.tail; i++) {
67             cout << (i - Q.head + 2) << ". " << Q.T[i].judul
68                 << " - " << Q.T[i].penulis << endl;
69         }
70     }
71 }
72
73 int main() {
74     RakBuku rak;
75     createQueue(rak);
76
77     Enqueue(rak, "Dilan", "Pidi Baiq");
78     Enqueue(rak, "Persuasion", "Jane Austen");
79     Enqueue(rak, "The Alchemist", "Paulo Coelho");
80     Enqueue(rak, "Supernova: Kesatria, Putri, & Bintang Jatuh", "Dee Lestari");
81     Enqueue(rak, "5cm", "Donny Dhirgantoro");
82
83     tampilRak(rak);
84
85     string judul, penulis;
86     Dequeue(rak, judul, penulis);
87     cout << "\nJudul buku yang diambil: " << judul << endl
88         << "Penulis: " << penulis << endl;
89
90     tampilRak(rak);
91
92     return 0;
93 }
94

```

- PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack> cd "c:\shellyn\kuliah\semester3\strukturData\teori\tugasStack\" ; if (\$?) { g++ queue.cpp -o queue } ; if (\$?) { .\queue }

Isi rak buku:

1. Dilan - Pidi Baiq
2. Persuasion - Jane Austen
3. The Alchemist - Paulo Coelho
4. Supernova: Kesatria, Putri, & Bintang Jatuh - Dee Lestari
5. 5cm - Donny Dhirgantoro

Judul buku yang diambil: Dilan

Penulis: Pidi Baiq

Isi rak buku:

1. Persuasion - Jane Austen
2. The Alchemist - Paulo Coelho
3. Supernova: Kesatria, Putri, & Bintang Jatuh - Dee Lestari
4. 5cm - Donny Dhirgantoro
5. 5cm - Donny Dhirgantoro

- PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack> █

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  const int MAX = 12;
6
7  struct Buku {
8      string judul;
9      string penulis;
10 };
11
12 struct RakBuku {
13     Buku T[MAX];
14     int head;
15     int tail;
16 };
17
18 void createQueue(RakBuku &Q) {
19     Q.head = 0;
20     Q.tail = 0;
21 }
22
23 bool isEmpty(RakBuku Q) {
24     return (Q.head == 0 && Q.tail == 0);
25 }
26
27 bool isFull(RakBuku Q) {
28     return (Q.tail == MAX);
29 }
30
31 void Enqueue(RakBuku &Q, string judul, string penulis) {
32     if (isFull(Q)) {
33         cout << "Rak penuh!" << endl;
34     } else if (isEmpty(Q)) {
35         Q.head = 1;
36         Q.tail = 1;
37         Q.T[Q.tail - 1].judul = judul;
38         Q.T[Q.tail - 1].penulis = penulis;
39     } else {
40         Q.tail++;
41         Q.T[Q.tail - 1].judul = judul;
42         Q.T[Q.tail - 1].penulis = penulis;
43     }
44 }
45
46 void Dequeue(RakBuku &Q, string &judul, string &penulis) {
47     if (isEmpty(Q)) {
48         cout << "Rak kosong!" << endl;
49     } else {
50         judul = Q.T[0].judul;
51         penulis = Q.T[0].penulis;
52
53         for (int i = 1; i < Q.tail; i++) {
54             Q.T[i - 1] = Q.T[i];
55         }
56
57         Q.tail--;
58         if (Q.tail == 0) Q.head = 0;
59     }
60 }
61

```

```

61 void tampilRak(RakBuku Q) {
62     if (isEmpty(Q)) {
63         cout << "Rak kosong." << endl;
64     } else {
65         cout << endl << "Isi rak buku:" << endl;
66         for (int i = Q.head - 1; i < Q.tail; i++) {
67             cout << (i - Q.head + 2) << ". " << Q.T[i].judul
68                 << " - " << Q.T[i].penulis << endl;
69         }
70     }
71 }
72
73 int main() {
74     RakBuku rak;
75     createQueue(rak);
76
77     Enqueue(rak, "Dilan", "Pidi Baiq");
78     Enqueue(rak, "Persuasion", "Jane Austen");
79     Enqueue(rak, "The Alchemist", "Paulo Coelho");
80     Enqueue(rak, "Supernova: Kesatria, Putri, & Bintang Jatuh", "Dee Lestari");
81     Enqueue(rak, "5cm", "Donny Dhirgantoro");
82
83     tampilRak(rak);
84
85     string judul, penulis;
86     Dequeue(rak, judul, penulis);
87     cout << "\nJudul buku yang diambil: " << judul << endl
88         << "Penulis: " << penulis << endl;
89
90     tampilRak(rak);
91
92     return 0;
93 }
94

```

- PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack> cd "c:\shellyn\kuliah\semester3\strukturData\teori\tugasStack\" ; if (\$?) { g++ queue.cpp -o queue } ; if (\$?) { .\queue }

Isi rak buku:

1. Dilan - Pidi Baiq
2. Persuasion - Jane Austen
3. The Alchemist - Paulo Coelho
4. Supernova: Kesatria, Putri, & Bintang Jatuh - Dee Lestari
5. 5cm - Donny Dhirgantoro

Judul buku yang diambil: Dilan

Penulis: Pidi Baiq

Isi rak buku:

1. Persuasion - Jane Austen
2. The Alchemist - Paulo Coelho
3. Supernova: Kesatria, Putri, & Bintang Jatuh - Dee Lestari
4. 5cm - Donny Dhirgantoro

- PS C:\shellyn\kuliah\semester3\strukturData\teori\tugasStack> █