

Started on	Monday, 17 October 2022, 12:26 PM
State	Finished
Completed on	Monday, 17 October 2022, 12:46 PM
Time taken	20 mins 29 secs
Grade	600.00 out of 600.00 (100%)

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Implementasikanlah ADT Queue dengan representasi array secara eksplisit dan alokasi statik.

Diberikan file berikut:

- [boolean.h](#)
- [queue.h](#)

Submit file hasil implementasi **queue.h**

C ▾

 [queue.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	4	Accepted	0.00 sec, 1.66 MB
2	4	Accepted	0.00 sec, 1.61 MB
3	4	Accepted	0.00 sec, 1.64 MB
4	4	Accepted	0.00 sec, 1.56 MB
5	4	Accepted	0.00 sec, 1.60 MB
6	4	Accepted	0.00 sec, 1.64 MB
7	4	Accepted	0.00 sec, 1.64 MB
8	4	Accepted	0.00 sec, 1.60 MB
9	4	Accepted	0.00 sec, 1.52 MB
10	4	Accepted	0.00 sec, 1.64 MB
11	4	Accepted	0.00 sec, 1.64 MB
12	4	Accepted	0.00 sec, 1.50 MB
13	4	Accepted	0.00 sec, 1.61 MB
14	4	Accepted	0.00 sec, 1.63 MB
15	4	Accepted	0.00 sec, 1.55 MB
16	4	Accepted	0.00 sec, 1.57 MB
17	4	Accepted	0.00 sec, 1.64 MB
18	4	Accepted	0.00 sec, 1.64 MB
19	4	Accepted	0.00 sec, 1.50 MB
20	4	Accepted	0.00 sec, 1.63 MB
21	4	Accepted	0.00 sec, 1.57 MB
22	4	Accepted	0.00 sec, 1.64 MB
23	4	Accepted	0.00 sec, 1.50 MB

23	4	Accepted	0.00 sec, 1.50 MB
No	Score	Verdict	Description

24	8	Accepted	0.00 sec, 1.64 MB
----	---	----------	-------------------

Question **2**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Implementasikanlah ADT Circular Queue dengan membuat file **circular_queue.c**!


Gunakanlah file [circular_queue.h](#) dan [boolean.h](#)

C

⬆

⬇

⬆

 [circular_queue.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	14	Accepted	0.00 sec, 1.66 MB
2	14	Accepted	0.00 sec, 1.59 MB
3	14	Accepted	0.00 sec, 1.60 MB
4	14	Accepted	0.00 sec, 1.54 MB
5	14	Accepted	0.00 sec, 1.66 MB
6	14	Accepted	0.00 sec, 1.64 MB
7	16	Accepted	0.00 sec, 1.71 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Implementasikan file header [reverse.h](#) yang berisi sebuah fungsi transferReverse yang berfungsi memindahkan isi dari satu queue ke queue lain dengan urutan yang terbalik.

Tidak diperbolehkan membuat array baru dalam pengerjaan soal ini. Gunakanlah fungsi-fungsi yang tersedia pada file [circular_queue.h](#) yang sudah kalian buat pada soal sebelumnya.

Contoh:

q1 (Kondisi awal)	q2 (Kondisi akhir)
[1, 2, 3]	[3, 2, 1]
[1]	[1]

Submit file **reverse.c**

C ▾

 [reverse.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.00 sec, 1.65 MB
2	25	Accepted	0.00 sec, 1.55 MB
3	25	Accepted	0.00 sec, 1.64 MB
4	25	Accepted	0.00 sec, 1.64 MB

Question **4**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Implementasikan [tanpaBomb.h](#) kedalam file tanpaBomb.c

Gunakan ADT queue yang telah dibuat pada pra-praktikum sebagai bantuan dalam menyelesaikan soal.



[tanpaBomb.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.57 MB
2	10	Accepted	0.00 sec, 1.65 MB
3	10	Accepted	0.00 sec, 1.64 MB
4	10	Accepted	0.00 sec, 1.63 MB
5	10	Accepted	0.00 sec, 1.71 MB
6	10	Accepted	0.00 sec, 1.64 MB
7	10	Accepted	0.00 sec, 1.57 MB
8	10	Accepted	0.00 sec, 1.50 MB
9	10	Accepted	0.00 sec, 1.61 MB
10	10	Accepted	0.00 sec, 1.63 MB

Question **5**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Submit file **copyqueue.c** yang merupakan implementasi dari **copyqueue.h**. Gunakan bantuan fungsi yang sudah terimplementasi di **queue.c**. Implementasi hanya boleh menggunakan fungsi yang sudah disediakan (CreateQueue, isEmpty, isFull, enqueue, dll) dan makro(IDX_HEAD, IDX_TAIL, HEAD, TAIL). Tidak boleh mengubah atau mengakses Buffer, idxHead dan idxTail.

[copyqueue.h](#)
[queue.h](#)
[queue.c](#)
[boolean.h](#)

C ▾

 [copyqueue.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.00 sec, 1.71 MB
2	25	Accepted	0.00 sec, 1.64 MB
3	25	Accepted	0.00 sec, 1.50 MB
4	25	Accepted	0.00 sec, 1.65 MB

Question **6**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Ibu Gra adalah pemilik dari restoran bintang 5, Aldat, dan terkadang datang ke restoran tersebut untuk melihat kondisi di lapangan dan membantu memasak. Tidak seperti para koki yang lainnya, Ibu Gra hanya dapat memasak 1 masakan untuk 1 waktunya, dengan setiap masakannya membutuhkan waktu yang berbeda-beda, dan tamu-tamu lainnya dapat melakukan permintaan untuk mendapat masakan dari Ibu Gra, karena masakannya yang sangat populer dan juga viral di social media. Suatu hari, Ibu Gra ingin mengetahui berapa jumlah masakan yang telah disajikan, waktu minimal yang dibutuhkan, dan waktu maksimal yang dibutuhkan pada 1 hari tertentu. Menurut Ibu Gra, masalah ini dapat diselesaikan dengan membuat kode dalam bahasa C.

Spesifikasi program:

1. Input berupa angka, yaitu 0, 1, dan 2. 0 berarti program berhenti menerima masukan. 1 berarti menerima permintaan (dengan tambahan parameter adalah waktu yang dibutuhkan dan waktu yang tidak valid akan diabaikan). 2 berarti menyajikan masakan.\
2. Output berupa jumlah masakan yang disaji, waktu minimal yang dibutuhkan, dan waktu maksimal yang dibutuhkan (diikuti dengan newline). Default value adalah 0.
3. Nama file adalah **restoran.c**, dengan menggunakan header dari **queue.h**.

Berikut adalah contoh interaksi input/output program.

Input	Output	Keterangan
1 2 1 5 2 0	1 2 2	Karena hanya menyajikan 1 masakan, maka total adalah 1, waktu minimal dan waktu maksimal yang dibuutuhkan adalah 2
1 2 2 1 5 2 0	2 2 5	Karena menyajikan 2 masakan, maka total adalah 2, dan waktu minimal yang dibutuhkan adalah 2 (pada pesanan pertama) dan waktu maksimal yang dibutuhkan adalh 5 (pada pesanan kedua).



[restoran.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.00 sec, 1.59 MB
2	20	Accepted	0.00 sec, 1.60 MB
3	20	Accepted	0.00 sec, 1.66 MB
4	20	Accepted	0.00 sec, 1.71 MB
5	20	Accepted	0.00 sec, 1.71 MB

