
Started on Thursday, 3 February 2022, 11:45 AM

State Finished

Completed on Thursday, 3 February 2022, 1:30 PM

Time taken 1 hour 44 mins

Marks 500.00/500.00

Grade **100.00** out of 100.00

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB



Buatlah implementasi kelas **BunchOfKeys** dalam bahasa pemrograman C++ yang memiliki sebuah atribut **n_keys** bertipe integer yang menyimpan jumlah kunci yang ada. Kelas juga **hanya** memiliki 2 buah *member function*, yaitu:

1. **add**: menambah kunci (hanya melakukan increment pada atribut **n_keys**)
2. **shake**: mengeluarkan bunyi "kringing" sejumlah **n_keys** kali (*setiap "kringing" diakhiri end-of-line*) apabila jumlah kunci > 1, atau mengeluarkan pesan "Tidak terjadi apa-apa" (*diakhiri end-of-line*) apabila jumlah kunci ≤ 1.

Pada saat konstruksi obyek, jumlah kunci semula adalah 0.

Diberikan header C++ sebagai berikut, Anda hanya diminta mengumpulkan file implementasi (**BunchOfKeys.cpp**) nya saja.

```
// BunchOfKeys.hpp
#ifndef __BUNCH_OF_KEYS_HPP__
#define __BUNCH_OF_KEYS_HPP__

#include <iostream>
using namespace std;

class BunchOfKeys {
public:
    // ctor
    BunchOfKeys();

    // member function
    void add();
    void shake();

private:
    int n_keys; // jumlah kunci yg ada
};

#endif
```

C++14

 [BunchOfKeys.cpp](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.00 sec, 2.80 MB
2	25	Accepted	0.00 sec, 2.89 MB
3	25	Accepted	0.02 sec, 2.95 MB
4	25	Accepted	0.00 sec, 2.93 MB

Question **2**

Correct

Mark 200.00 out of 200.00

Time limit

1 s

Memory limit

64 MB

Diberikan kelas **Bottle** yang merupakan botol dengan radius alas botol (**radius**) dan tinggi botol (**height**). Kelas Bottle menyimpan berapa banyak botol yang telah diciptakan (**numOfBottle**). Botol yang diciptakan sedang menampung cairan setinggi **waterHeight** dan menyimpan nomor urutan diciptakannya botol ini (**id**).

Keterangan: Nilai **id** didapat dari jumlah botol yang telah diciptakan (**numOfBottle + 1**)

[Bottle.hpp](#)

Buatlah definisi dari **Bottle.cpp**

C++14

 [Bottle.cpp](#)

Score: 200

Blackbox

Score: 200

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 3.07 MB
2	10	Accepted	0.00 sec, 2.94 MB
3	10	Accepted	0.02 sec, 3.07 MB
4	10	Accepted	0.00 sec, 2.88 MB
5	10	Accepted	0.02 sec, 3.09 MB
6	10	Accepted	0.00 sec, 3.16 MB
7	10	Accepted	0.01 sec, 2.86 MB
8	10	Accepted	0.00 sec, 2.97 MB
9	10	Accepted	0.00 sec, 3.02 MB
10	10	Accepted	0.00 sec, 3.00 MB
11	10	Accepted	0.00 sec, 2.98 MB
12	10	Accepted	0.01 sec, 2.98 MB
13	10	Accepted	0.01 sec, 3.07 MB
14	10	Accepted	0.00 sec, 3.07 MB
15	10	Accepted	0.00 sec, 3.16 MB
16	10	Accepted	0.00 sec, 2.86 MB
17	10	Accepted	0.02 sec, 3.08 MB
18	10	Accepted	0.00 sec, 3.07 MB
19	10	Accepted	0.00 sec, 3.07 MB
20	10	Accepted	0.00 sec, 3.07 MB

20	10	Accepted	0.02 sec, 3.02 MB
----	----	----------	-------------------

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Berikut adalah deklarasi dan definisi dari kelas Box. Kelas Box menghasilkan keluaran khusus jika constructor, copy constructor, copy assignment operator, destructor, dan fungsi peek dipanggil.

[Box.hpp](#)

[Box.cpp](#)

Tugas anda: Unggah file **main.cpp** yang isinya adalah main program, yang menggunakan kelas A untuk menghasilkan keluaran berikut:

```
new empty box 2
new empty box with default id 0
new empty box 1
assign box 0 <- 2
copy box 1
box 2
box 1
destroy box 2
destroy box 1
destroy box 1
destroy box 2
```

C++14

 [main.cpp](#)

Score: 10

Blackbox

Score: 10

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 2.86 MB

Question **4**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

PrioQueue merupakan sebuah struktur data yang merepresentasikan antrian dengan prioritas. Elemen dari PrioQueue merupakan kelas PQElmt yang memiliki value dan prio. PrioQueue selalu terurut dari yang **terbesar**.

Selain itu, dapat dilakukan penggabungan 2 PrioQueue dengan operator penjumlahan +.

Contoh: (format queue dalam bentuk [(value, prio)])

A = [(3, 3), (5, 2), (1, 1)]

B = [(1, 3), (2, 2), (,)]

Maka

A + B = [(1, 3), (3, 3), (2, 2), (5, 2), (1, 1)]

PrioQueue juga punya aksesor [k] untuk mengakses elemen ke k.

[PQElmt.hpp](#), [PQElmt.cpp](#), [PrioQueue.hpp](#), [PrioQueue.cpp](#)

Diberikan definisi dan sebagian realisasi PQElmt dan PrioQueue, lengkapi bagian yang masih kosong!

Upload PQElmt.cpp dan PrioQueue.cpp yang sudah dilengkapi dan di-zip menjadi **PrioQueue.zip**

C++14

 [PrioQueue.zip](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 2.91 MB
2	10	Accepted	0.00 sec, 2.80 MB
3	10	Accepted	0.01 sec, 2.89 MB
4	10	Accepted	0.00 sec, 2.89 MB
5	10	Accepted	0.00 sec, 2.88 MB
6	10	Accepted	0.00 sec, 2.89 MB
7	10	Accepted	0.00 sec, 2.98 MB
8	10	Accepted	0.00 sec, 2.87 MB
9	10	Accepted	0.00 sec, 2.92 MB
10	10	Accepted	0.00 sec, 2.93 MB

◀ Slide Minggu 3

Jump to...

