

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Nama File: set.c

Pada minggu kemarin, nimons ditugaskan oleh Gro untuk mengamankan taman loji di ITB Jatinangor. Untuk menjaga taman tersebut, Gro memutuskan untuk memberikan akses khusus kepada para nimons terpilih. Karena ingin memastikan tidak ada nimon yang masuk lebih dari sekali dan nomor akses mereka unik, Gro meminta bantuan Dr Neroifa untuk membuat sebuah sistem pencatatan.

Awalnya, Dr Neroifa berencana menggunakan array biasa, tetapi ia sadar bahwa akan lebih efektif menggunakan struktur data **Set**, karena Set tidak mengizinkan elemen duplikat dan bisa membantu mengelola data dengan lebih rapi. Dalam Set ini, setiap elemen merepresentasikan **nomor akses unik** yang dimiliki nimon untuk masuk ke taman.

Bantulah Dr Neroifa untuk membuat **ADT Set** dengan melengkapi [set.h](#). Kumpulkan hanya `set.c`. Berikut file [boolean.h](#)

Gunakan [main.c](#) untuk menguji program yang dibuat.

Keluaran yang diharapkan:

Set is empty

Set elements: 3 5 8

Is 3 a member? Yes

Is 7 a member? No

Set elements after deleting 3: 5 8

Set is not full

Pastikan setiap output diakhiri oleh **endline** ("`\n`")!

C

 [set.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.00 sec, 1.50 MB
2	20	Accepted	0.00 sec, 1.67 MB
3	20	Accepted	0.00 sec, 1.59 MB
4	20	Accepted	0.00 sec, 1.50 MB
5	20	Accepted	0.00 sec, 1.57 MB

Question **2**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Nama File: hashmap.zip



open address

Dr. Neroifa sedang kebingungan, karena ia tidak sengaja kembali ke jaman batu. Ia akhirnya terpaksa menciptakan komputer dengan ukuran penyimpanan yang relatif kecil untuk melakukan pendataan nimons. Namun, sayangnya jumlah nimons terlalu banyak, sehingga perlu dilakukan **hashing**.

Silakan buat ADT Hash Map untuk membantu Dr. Neroifa membangun komputernya. Untungnya ada dokumen yang tidak sengaja terbawa oleh Dr. Neroifa. Berikut dokumen yang dapat membantu pengerjaan anda [hashmap.h](#).

Dr. Neroifa juga memiliki request supaya outputnya disesuaikan sesuai dengan ketentuan berikut:

- Diberi [main.c](#) anda diminta untuk menyesuaikan outputnya seperti file [berikut](#).

Notes:

- Silakan gunakan [Boolean.h](#)
- Kumpulkan **hashmap.c** dan **hashmap.h** dalam 1 file **zip**

C

[hashmap.zip](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	14	Accepted	0.00 sec, 1.51 MB
2	14	Accepted	0.00 sec, 1.63 MB
3	14	Accepted	0.00 sec, 1.60 MB
4	14	Accepted	0.00 sec, 1.50 MB
5	14	Accepted	0.00 sec, 1.73 MB
6	14	Accepted	0.00 sec, 1.67 MB
7	16	Accepted	0.00 sec, 1.63 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	2 s
Memory limit	8 MB

Nama File: map.c

Pada minggu kemarin, nimons ditugaskan Gro untuk mencuri patung Optimus Prime di ITB Jatinangor. Misi berjalan sukses dan Gro berhasil mendapatkan patung Optimus Prime, walaupun misi berjalan dengan cukup sulit karena keamanan ITB Jatinangor sangat kuat terutama karena adanya satpam ITB Jatinangor. Atas keberhasilan misi ini, Gro berkeinginan untuk memberikan hadiah kepada nimons-nya. Hadiah yang diberikan adalah sebuah ruangan relaksasi untuk para nimons. Ruangan ini memiliki bilik - bilik yang setiap biliknya bisa diisi oleh 1 nimon. Karena jumlah nimon sangat banyak dan jumlah bilik terbatas, Dr Neroifa kemudian membuat sebuah program untuk melakukan tracking bilik mana saja yang sedang dipakai. Proses tracking awalnya ingin dilakukan menggunakan array tapi Dr Neroifa baru sadar bahwa nomor - nomor bilik random dan tidak berurutan. Dr Neroifa lalu sadar bahwa proses tracking bisa dilakukan dengan menggunakan ADT Map dengan keynya adalah nomor bilik dan value-nya adalah nama nimon yang sedang memakai bilik. Bantulah Dr Neroifa untuk membuat ADT map dengan melengkapi [map.h](#). Kumpulkan hanya map.c.

Gunakan kode main.c berikut untuk melakukan testing terhadap kode yang telah di buat.

```
#include "map.h"
#include <stdio.h>

int main() {
    Map *map = create_map(5);
    char *nimons[] = {"Kebin", "Stuart", "Dave", "Pop", "Larry", "Mel"};
    for (int i = 0; i <= 5; i++) {
        if (!map_insert(map, i, nimons[i])) {
            printf("Terjadi error ketika memasukkan %s ke bilik %d\n",
                nimons[i], i);
        }
    }
    for (int i = 0; i <= 5; i++) {
        const char *nimon = map_get(map, i);
        if (nimon != NULL) {
            printf("Bilik ke-%d diisi oleh %s\n", i, nimon);
        } else {
            printf("Terjadi error ketika melakukan pengecekan ke bilik %d\n",
                i);
        }
    }
    if (map_delete(map, 1)) {
        printf("Bilik ke-1 berhasil dikosongkan\n");
    }
    // Mencoba melakukan delete ke key yang sudah dihapus
    if (!map_delete(map, 1)) {
        printf("Terjadi error ketika ingin melakukan pengosongan bilik ke 1\n");
    }
    printf("Ukuran map saat ini %d\n", map_size(map));
    return 0;
}
```

Output yang diharapkan:

Terjadi error ketika memasukkan Mel ke bilik 5

Bilik ke-0 diisi oleh Kebin

Bilik ke-1 diisi oleh Stuart

Bilik ke-2 diisi oleh Dave

Bilik ke-3 diisi oleh Pop

Bilik ke-4 diisi oleh Larry

Terjadi error ketika melakukan pengecekan ke bilik 5

Bilik ke-1 berhasil dikosongkan

Terjadi error ketika ingin melakukan pengosongan bilik ke 1

Ukuran map saat ini 4

Tips:

Jika terkena memory limit di test case 5, silahkan evaluasi cara **alokasi** dan **dealokasi** memori di kode kalian.

C

 [map.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.00 sec, 1.65 MB
2	20	Accepted	0.00 sec, 1.73 MB
3	20	Accepted	0.00 sec, 1.67 MB
4	20	Accepted	0.00 sec, 1.55 MB
5	20	Accepted	0.04 sec, 1.65 MB

[◀ Post Praktikum 5 - K3 & K4](#)

Jump to...