

Matrix - Praktikum

Started on	Thursday, 24 October 2024, 9:21 PM
State	Finished
Completed on	Thursday, 24 October 2024, 9:34 PM
Time taken	13 mins 11 secs
Grade	300.00 out of 300.00 (100%)

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Kekacauan

Deskripsi

Tuan Neb sedang bermain sebuah *board game* yang dapat digambarkan sebagai sebuah papan berukuran $N \times N$. Di dalam papan tersebut, ada beberapa benteng. Pada baris r dan kolom c , apabila papan[r][c] bernilai 0 maka artinya **tidak ada benteng** dalam titik tersebut. Sebaliknya, apabila papan[r][c] bernilai lebih dari 0, maka pada titik itu **terdapat benteng** dengan **kekuatan papan[r][c]**.

Benteng dalam permainan ini memiliki kemampuan mirip seperti benteng pada catur namun lebih kuat, benteng bisa menyerang semua benteng yang memiliki **kolom sama** atau **baris sama** dengannya.

Nilai **kekacauan** suatu benteng adalah total kekuatan benteng yang dapat menyerangnya. Tugas anda adalah membantu Tuan Neb menghitung **total** nilai kekacauan dalam papan tersebut.

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($2 \leq N \leq 100$).
- Baris kedua berisi $N \times N$ kotak. Tiap kotak dapat berupa angka 0 hingga 100000.

Format Keluaran

Total nilai kekacauan modulo $10^9 + 7$.

Contoh

No	Masukan	Keluaran
1.	2 1 0 0 1	0
2	2 1 1 0 0	2
3	3 1 2 0 0 0 3 4 0 0	8

Penjelasan

Pada contoh 1, tidak ada benteng yang dapat diserang oleh benteng lain sehingga kekacauan bernilai 0.

Pada contoh 2, kedua benteng memiliki kekacauan 1 sehingga total kekacauan bernilai 2.

Pada contoh 3, benteng dengan kekuatan 1 memiliki kekacauan 6, benteng dengan kekuatan 2 memiliki kekacauan 1, benteng dengan kekuatan 3 memiliki kekacauan 0, dan benteng dengan kekuatan 4 memiliki kekacauan 1. Sehingga, total kekacauan adalah $6 + 1 + 0 + 1 = 8$.

Catatan

- Keluaran diakhiri dengan newline (\n)
- Download dan gunakan file [boolean.h](#), [matrix.h](#) & [matrix.c](#) yang telah dibuat sebelumnya
- Download dan kumpulkan jawaban dalam file bernama **kekacauan.c**

C

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.66 MB
2	10	Accepted	0.00 sec, 1.72 MB
3	10	Accepted	0.00 sec, 1.61 MB
4	10	Accepted	0.00 sec, 1.61 MB
5	10	Accepted	0.00 sec, 1.70 MB
6	10	Accepted	0.00 sec, 1.68 MB
7	10	Accepted	0.00 sec, 1.68 MB
8	10	Accepted	0.00 sec, 1.54 MB
9	10	Accepted	0.00 sec, 1.64 MB
10	10	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

Grid

Deskripsi

Bayangkan terdapat sebuah **Grid** berukuran $N \times N$ yang kotak-kotaknya mungkin memiliki jebakan. Tidak diperbolehkan untuk berpindah ke kotak yang memiliki jebakan.

Tugas Anda adalah menghitung jumlah jalur dari kotak kiri atas ke kotak kanan bawah. Anda hanya bisa bergerak ke kanan atau ke bawah.

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($1 \leq N \leq 1000$).
- Baris kedua berisi $N \times N$ kotak. Tiap kotak dapat berupa angka 0 (aman) dan 1 (jebakan).

Format Keluaran

Jumlah jalur dengan modulo $10^9 + 7$.

Contoh

No	Masukan	Keluaran
1.	3 0 0 1 1 0 0 0 0 0	2

Penjelasan

Jumlah jalur yang mungkin adalah 2, yaitu Kanan-Bawah-Kanan-Bawah & Kanan-Bawah-Bawah-Kanan.

Catatan

- Keluaran diakhiri dengan newline (`\n`)
- Download dan gunakan file [boolean.h](#), [matrix.h](#) (terdapat perubahan pada ROW_CAP & COL_CAP) & **matrix.c** yang telah dibuat sebelumnya
- Download dan kumpulkan jawaban dalam file bernama [grid.c](#)

C

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 3.55 MB
2	10	Accepted	0.00 sec, 3.59 MB
3	10	Accepted	0.00 sec, 3.55 MB
4	10	Accepted	0.00 sec, 3.60 MB
5	10	Accepted	0.00 sec, 3.62 MB
6	10	Accepted	0.00 sec, 3.66 MB

No	Score	Verdict	Description
7	10	Accepted	0.00 sec, 3.62 MB
8	10	Accepted	0.00 sec, 3.57 MB
9	10	Accepted	0.00 sec, 3.67 MB
10	10	Accepted	0.06 sec, 9.34 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Matriks Hanoi

Deskripsi

Tower of Hanoi merupakan sebuah permainan dimana kamu harus memindahkan banyak piringan dari lokasi awal ke akhir. Namun, sebuah piringan tidak bisa berada di atas piringan yang lebih kecil darinya.

Saat ini, Tuan Farr sedang memainkan permainan Matriks Hanoi. Permainan ini dilakukan pada sebuah papan $N \times N$ dan setiap 4 piringan saling tergabung menjadi sebuah gabungan piringan yang berukuran 2×2 . Tuan Farr hanya bisa menaruh sebuah gabungan piringan jika dan hanya jika seluruh piringan berukuran **lebih kecil** dari piringan yang di bawahnya. Gabungan piringan ini **bisa diputar** namun **tidak bisa dibalik**.

Karena permainan ini cukup sulit, Tuan Farr hanya meminta bantuanmu untuk menghitung berapa **banyak kemungkinan lokasi** Tuan Farr dapat menaruh suatu gabungan piringan pada suatu papan.

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($2 \leq N \leq 100$).
- Baris selanjutnya berisi $N \times N$ kotak yang menyatakan kondisi papan. Tiap kotak dapat berupa angka 1 hingga 100000.
- Baris terakhir berisi 2×2 kotak yang menyatakan gabungan piringan yang dimiliki Tuan Farr. Tiap kotak dapat berupa angka 1 hingga 100000.

Format Keluaran

Banyak kemungkinan lokasi Tuan Farr menaruh gabungan piringan yang dimilikinya.

Contoh

No	Masukan	Keluaran
1.	2 2 3 2 2 1 2 1 1	1
2	2 2 3 2 2 2 1 1 1	1
3	3 2 2 2 2 2 2 2 2 2 1 1 1 1	4
4	4 30 20 20 30 20 10 10 20 20 10 10 20 30 20 20 30 5 15 15 25	4

Penjelasan

Pada contoh 1, Tuan Farr dapat menaruh gabungan piringan di titik 1,1 tanpa memutarnya. Sedangkan, pada contoh 2, Tuan Farr juga hanya dapat menaruhnya di posisi 1,1 namun harus memutarnya terlebih dahulu sebesar 90 derajat.

Pada contoh 3, keempat titik 1,1, 1,2, 2,1, dan 2,2 merupakan lokasi valid sehingga keluarannya bernilai 4. (Perhatikan meskipun semua rotasi valid namun yang dihitung hanya jumlah lokasinya)

Pada contoh 4, setiap 4 putaran dari matriks

5 15

15 25

masing-masing dapat ditaruh di salah satu pojok matriks. Sehingga, jawabannya adalah 4.

Catatan

- Keluaran diakhiri dengan newline (\n)
- Download dan gunakan file [boolean.h](#), [matrix.h](#) & [matrix.c](#) yang telah dibuat sebelumnya
- Download dan kumpulkan jawaban dalam file bernama **mhanoi.c**

C

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.61 MB
2	10	Accepted	0.00 sec, 1.55 MB
3	10	Accepted	0.00 sec, 1.71 MB
4	10	Accepted	0.00 sec, 1.78 MB
5	10	Accepted	0.01 sec, 1.58 MB
6	10	Accepted	0.01 sec, 1.79 MB
7	10	Accepted	0.01 sec, 1.80 MB
8	10	Accepted	0.01 sec, 1.58 MB
9	10	Accepted	0.01 sec, 1.76 MB
10	10	Accepted	0.01 sec, 1.73 MB

Jump to...