

[Dashboard](#) / My courses / [ITB\\_IF2110\\_1\\_2526](#) / [Praktikum 7](#) / Post Praktikum 7

**Started on** Thursday, 6 November 2025, 12:09 PM

**State** Finished

**Completed on** Thursday, 6 November 2025, 12:23 PM

**Time taken** 14 mins 52 secs

**Marks** 400.00/400.00

**Grade** **10.00** out of 10.00 (**100%**)

**Question 1**

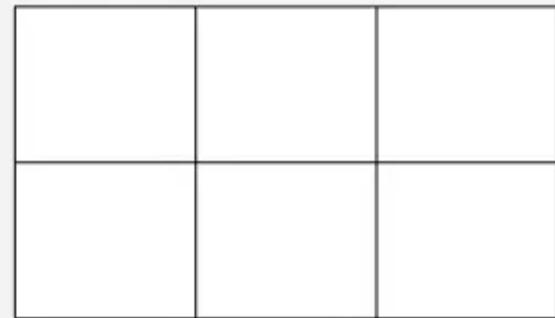
Correct

Mark 100.00 out  
of 100.00

Time limit	1 s
Memory limit	64 MB

**Nama file:** lantai.c

Diberikan sebuah ruangan berukuran  $2 \times N$  ( $N > 0$ ). dan ubin berukuran  $2 \times 1$ . Hitunglah berapa banyak cara menyusun ruangan tersebut menggunakan ubin berukuran  $2 \times 1$ . Ubin dapat diletakkan secara horizontal ( $1 \times 2$ ) atau vertikal ( $2 \times 1$ ). Soal ini harus diselesaikan dengan menggunakan rekursi. Jangan lupa newline di akhir output.

**Board****Tiles****Format Masukan**

- Satu bilangan bulat  $n$  dengan  $1 \leq n \leq 30$ .

**Format Keluaran**

- Satu bilangan bulat yang menyatakan banyak cara menyusun ubin pada lantai.
- Keluaran diakhiri dengan **newline**.

**Contoh**

Input (N)	Output	Penjelasan
3	3	Cara yang mungkin : - tiga ubin diletakkan secara vertikal. - satu ubin secara vertikal dan dua ubin secara horizontal (2 cara)
1	1	Hanya terdapat satu cara

C

 [lantai.c](#)**Score: 100****Blackbox****Score: 100****Verdict: Accepted**

**Evaluator: Exact**

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.50 MB
2	10	Accepted	0.00 sec, 1.50 MB
3	10	Accepted	0.00 sec, 1.49 MB
4	10	Accepted	0.00 sec, 1.59 MB
5	10	Accepted	0.00 sec, 1.66 MB
6	10	Accepted	0.00 sec, 1.63 MB
7	10	Accepted	0.00 sec, 1.62 MB
8	10	Accepted	0.00 sec, 1.59 MB
9	10	Accepted	0.00 sec, 1.59 MB
10	10	Accepted	0.01 sec, 1.59 MB

**Question 2**

Correct

Mark 100.00 out  
of 100.00

Time limit	1 s
Memory limit	64 MB

**Nama file:** parentheses.c

Diberikan sebuah bilangan bulat  $n$  (jumlah pasangan kurung), hitung **total banyaknya string kurung seimbang** yang dapat dibentuk menggunakan tepat  $n$  buah '(' dan  $n$  buah ')'. Sebuah string disebut *seimbang* bila jumlah ')' dan '(' pada akhir string sama dan memiliki sekuenks string kurung yang valid. Dengan kata lain, setiap kurung buka '(' memiliki pasangan kurung tutup ')' yang tepat.

**Format Masukan**

- Satu bilangan bulat  $n$  dengan batasan:  $0 \leq n \leq 20$ .

**Format Keluaran**

- Satu bilangan bulat: **total jumlah** kombinasi kurung seimbang untuk  $n$ .
- Keluaran diakhiri dengan **newline**.

**Contoh**

Input	Output	Penjelasan Singkat
0	0	
1	1	"()".
2	2	"((0))", "((0))", "(0)0)".
3	5	"((0))", "((0))", "(0)0)", "0(0)", "000)".

C

[parentheses.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, 1.62 MB
3	10	Accepted	0.00 sec, 1.49 MB
4	10	Accepted	0.00 sec, 1.59 MB
5	10	Accepted	0.00 sec, 1.51 MB
6	10	Accepted	0.00 sec, 1.63 MB
7	10	Accepted	0.00 sec, 1.71 MB
8	10	Accepted	0.00 sec, 1.59 MB

No	Score	Verdict	Description
9	10	Accepted	0.00 sec, 1.49 MB
10	10	Accepted	0.00 sec, 1.57 MB

**Question 3**

Correct

Mark 100.00 out  
of 100.00

Time limit	1 s
Memory limit	64 MB

**Nama file:** subsets.c

Diberikan sebuah bilangan bulat **n** (menyatakan panjang array) dan sebuah array bilangan bulat terurut dengan elemen-elemen unik. Kembalikan semua kemungkinan **subset** dari array tersebut.

Set solusi tidak boleh mengandung **subset** yang duplikat. Dua subset dianggap duplikat jika memiliki elemen yang sama, meskipun urutannya berbeda (misalnya [1, 2] dan [2, 1]).

Semua subset harus ditampilkan dalam **urutan yang benar**, yaitu:

1. **Diurutkan berdasarkan panjang subset (jumlah elemen)** dari yang paling sedikit ke paling banyak.
  - Contoh: subset kosong [] ditampilkan terlebih dahulu.
2. **Untuk subset dengan panjang yang sama**, urutan ditentukan secara **leksikografis berdasarkan posisi elemen dalam array input**.

Contoh urutan yang benar dapat dilihat pada contoh output dibawah.

**Catatan:** Solusi wajib menggunakan **prosedur rekursif** dalam penyelesaiannya.

**Format Masukan**

- Satu bilangan bulat **n** dengan batasan: **0 ≤ n ≤ 10**.
- Baris berikutnya adalah bilangan bulat unik yang terurut membesar.

**Format Keluaran**

- Semua subset dari array, ditampilkan dalam urutan yang benar.
- Setiap subset ditulis dalam format daftar (contoh: [3, 6]).
- Keluaran diakhiri dengan **newline**.

**Contoh****Input**

4

3 6 8 9

**Output**

C
---

 [subsets.c](#)
**Score: 100****Blackbox****Score: 100****Verdict: Accepted****Evaluator: Exact**

No	Score	Verdict	Description
1	16	Accepted	0.00 sec, 1.64 MB
2	16	Accepted	0.00 sec, 1.66 MB
3	16	Accepted	0.00 sec, 1.64 MB
4	16	Accepted	0.00 sec, 1.60 MB
5	16	Accepted	0.00 sec, 1.60 MB
6	20	Accepted	0.00 sec, 1.62 MB

**Question 4**

Correct

Mark 100.00 out  
of 100.00

Time limit	1 s
Memory limit	64 MB

**Nama file:** password.c

Diberikan sebuah bilangan bulat  $n$ . Tampilkan seluruh kombinasi bilangan tersebut. Gunakan file header [berikut](#) sebagai panduan, jangan lupa untuk melakukan import file header tersebut.

**Format Masukan**

- Satu bilangan bulat  $n$  dengan batasan:  $0 \leq n \leq 10e9$ .

**Format Keluaran**

- Seluruh kombinasi bilangan  $n$

**Contoh**

Input	Output
12	21 12
123	321 312 231 213 132 123
4321	1234 1243 1324 1342 1423 1432 2134 2143 2314 2341 2413 2431 3124 3142 3214 3241 3412 3421 4123 4132 4213 4231 4312 4321

C

[password.c](#)**Score: 100**

## Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.71 MB
2	10	Accepted	0.00 sec, 1.96 MB
3	10	Accepted	0.14 sec, 1.80 MB
4	10	Accepted	0.00 sec, 1.71 MB
5	10	Accepted	0.00 sec, 1.94 MB
6	10	Accepted	0.00 sec, 1.82 MB
7	10	Accepted	0.00 sec, 1.82 MB
8	10	Accepted	0.00 sec, 1.83 MB
9	10	Accepted	0.00 sec, 1.83 MB
10	10	Accepted	0.00 sec, 1.79 MB

◀ Praktikum 7

Jump to...

boolean.h ►