

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

# Penyontek (Kurang) Andal

Buat program dengan nama **awasketahuan.c** untuk mengimplementasikan [awasketahuan.h](#)

Gunakan [stack.h](#), [stack.c](#), dan [boolean.h](#)

## Deskripsi

Seorang peserta mata kuliah IF4222 bernama Bobi Bola ditugaskan untuk membuat sebuah *stack* beserta isinya. Namun, karena beliau malas, beliau memilih untuk menyalin *stack* milik temannya! Sejauh ini Bobi meminta temannya mengirim *stack* lewat surel, kemudian disalin satu per satu.

Akan tetapi, Bobi mendapat petuah alam, kemudian memutuskan lebih cepat bila disalin menggunakan program saja. Selain itu, Bobi juga ingin memodifikasi *stack* milik temannya agar tidak mencurigakan. Apabila suatu elemen pada *stack* bernilai genap, maka bagi elemen tersebut dengan dua. Elemen bernilai ganjil disalin mentah-mentah saja. Bantu Bobi membuat program tersebut!

## Format Masukan

Sebuah *stack* berisi *integer*. **Tidak perlu scanf(), gunakan argumen stack pada fungsi!**

## Format Keluaran

Sebuah *stack* berisi *integer*. **Tidak perlu printf(), lihat fungsi itu akan meminta return!**

## Batasan

*Stack* tidak kosong dan jumlah elemen maksimum *stack* adalah 100.

## Contoh Input 1

Masukan:

8

7

6

5

4

3

Keluaran:

4

7

3

5

2

3

## Penjelasan

Elemen pertama (8), ketiga (6), dan kelima (4) bernilai genap sehingga dibagi dua, elemen lainnya disalin biasa saja.

## Contoh Input 2

Masukan:

12

7

3

3

6

Keluaran:

6  
7  
3  
3  
3

## Penjelasan

Elemen pertama (12) kelima (6) bernilai genap sehingga dibagi dua, elemen lainnya disalin biasa saja.



Score: 100

## Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 5     | Accepted | 0.00 sec, 1.58 MB |
| 2  | 5     | Accepted | 0.00 sec, 1.65 MB |
| 3  | 5     | Accepted | 0.00 sec, 1.72 MB |
| 4  | 5     | Accepted | 0.00 sec, 1.50 MB |
| 5  | 5     | Accepted | 0.00 sec, 1.64 MB |
| 6  | 5     | Accepted | 0.01 sec, 1.67 MB |
| 7  | 5     | Accepted | 0.00 sec, 1.65 MB |
| 8  | 5     | Accepted | 0.00 sec, 1.72 MB |
| 9  | 5     | Accepted | 0.00 sec, 1.65 MB |
| 10 | 5     | Accepted | 0.00 sec, 1.65 MB |
| 11 | 5     | Accepted | 0.00 sec, 1.65 MB |
| 12 | 5     | Accepted | 0.00 sec, 1.67 MB |
| 13 | 5     | Accepted | 0.00 sec, 1.66 MB |
| 14 | 5     | Accepted | 0.00 sec, 1.66 MB |
| 15 | 5     | Accepted | 0.00 sec, 1.61 MB |
| 16 | 5     | Accepted | 0.00 sec, 1.51 MB |
| 17 | 5     | Accepted | 0.00 sec, 1.68 MB |
| 18 | 5     | Accepted | 0.00 sec, 1.72 MB |
| 19 | 5     | Accepted | 0.00 sec, 1.51 MB |
| 20 | 5     | Accepted | 0.00 sec, 1.68 MB |

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

# Fix Tags

Buat program dengan nama **fixTags.c**

## Deskripsi

Anda adalah seorang programmer senior yang diberikan tugas memperbaiki pekerjaan anak unpaid intern. Anak intern tersebut tidak tahu cara menggunakan tags di HTML, sehingga hasil pekerjaan miliknya memiliki tags yang berantakan.

Tugas kamu adalah membuat suatu program yang mengubah string yang memiliki tags yang berantakan menjadi string dengan tags yang benar.

### DEFINISI TAGS YANG BENAR UNTUK SOAL INI :

- 1. Setiap tag **<** harus diakhiri dengan **>**, tags yang tidak memiliki awalan atau akhiran akan dihapuskan dan tidak ditulis di hasil akhir
- 2. Tag yang ditulis dengan aturan **<<>>** harus dituliskan dengan format **<><>**

## Format Masukan

- 1. Program akan menerima 2 input, yaitu suatu integer N yang merupakan panjang string yang akan dimasukkan dan string input yang akan diproses. Kedua masukan dimasukkan di 2 baris yang berbeda.
- 2. Asumsikan bahwa N selalu lebih besar daripada 0 dan input tidak pernah kosong
- 3. Asumsikan bahwa string yang dimasukan hanya terdiri dari simbol **<** dan **>** saja
- 4. Asumsikan bahwa panjang input selalu sesuai dengan N

## Format Keluaran

Sebuah string yang sudah memiliki tag yang mematuhi aturan tag yang benar. Diakhiri dengan NEWLINE.

## Batasan

Untuk membantu pengerjaan soal ini, HARAP GUNAKAN [boolean.h](#), [stack.h](#) dan [stack.c](#) yang disediakan. **ADT stack ini memiliki tipe elemen char jadi HARAP menggunakan ADT yang disediakan.**

## Contoh Input 1

Masukan:

5<><>>

Keluaran:

<><>

## Penjelasan

Program akan menghilangkan **>** yang ada di akhir karena tidak memiliki tag awalan.

## Contoh Input 2

Masukan:

6<<>><>

Keluaran:

<><><>

## Penjelasan

Program akan akan mengubah brackets **<<>>** dan menuliskannya menjadi **<><>** sehingga hasil akhir yang akan dikeluarkan adalah **<><><>**

## Contoh Input 3

Masukan:

4

<<<<

Keluaran:

## Penjelasan

Program akan akan menghapus semua bracket yang tidak memenuhi aturan sehingga tidak menuliskan apapun selain newline.

C

⬆

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 10    | Accepted | 0.00 sec, 1.55 MB |
| 2  | 10    | Accepted | 0.00 sec, 1.64 MB |
| 3  | 10    | Accepted | 0.00 sec, 1.60 MB |
| 4  | 10    | Accepted | 0.00 sec, 1.60 MB |
| 5  | 10    | Accepted | 0.00 sec, 1.49 MB |
| 6  | 10    | Accepted | 0.00 sec, 1.66 MB |
| 7  | 10    | Accepted | 0.00 sec, 1.55 MB |
| 8  | 10    | Accepted | 0.00 sec, 1.48 MB |
| 9  | 10    | Accepted | 0.00 sec, 1.60 MB |
| 10 | 10    | Accepted | 0.00 sec, 1.54 MB |

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

Submit file dividestack.c yang merupakan hasil implementasi [dividestack.h](#)  
Gunakan [stack.h](#), [stack.c](#), [boolean.h](#)

C ▾

 [dividestack.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, 1.61 MB |
| 3  | 10    | Accepted | 0.00 sec, 1.64 MB |
| 4  | 10    | Accepted | 0.00 sec, 1.64 MB |
| 5  | 10    | Accepted | 0.00 sec, 1.64 MB |
| 6  | 10    | Accepted | 0.00 sec, 1.55 MB |
| 7  | 10    | Accepted | 0.00 sec, 1.60 MB |
| 8  | 10    | Accepted | 0.00 sec, 1.58 MB |
| 9  | 10    | Accepted | 0.00 sec, 1.60 MB |
| 10 | 10    | Accepted | 0.00 sec, 1.67 MB |