

Started on	Saturday, 9 November 2024, 9:33 PM
State	Finished
Completed on	Tuesday, 12 November 2024, 11:40 AM
Time taken	2 days 14 hours
Grade	400.00 out of 400.00 (100%)

Time limit	1 s
Memory limit	64 MB

Inventory

Deskripsi

Anda diminta untuk mengelola sistem inventaris barang dengan aturan yang sederhana. Pada awalnya, sistem catatan barang kosong.

Anda diberikan sebuah daftar catatan sepanjang N . Setiap operasi dapat berupa salah satu dari berikut ini:

- **Sebuah bilangan bulat positif (satuan) x** : Tambahkan jumlah barang baru sebesar x ke dalam catatan.
- **'+'**: Tambahkan jumlah barang baru yang merupakan penjumlahan dari dua barang sebelumnya.
- **'X'**: Batalkan jumlah barang terakhir, menghapusnya dari catatan.
- **'S'**: Tambahkan jumlah barang baru yang merupakan total dari semua jumlah barang yang ada, kemudian hapus jumlah barang yang digunakan untuk menghitung total tersebut.
- **'A'**: Tambahkan jumlah barang baru yang merupakan rata-rata dari semua jumlah barang yang ada (dibulatkan ke bawah jika hasilnya pecahan), lalu hapus jumlah barang yang digunakan untuk menghitung rata-rata tersebut.

Tugas Anda adalah mengembalikan total jumlah barang dalam catatan setelah semua operasi selesai diterapkan.

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($1 \leq N \leq 100$).
- Baris kedua berisi N karakter yang merupakan urutan operasi dan hanya terdiri dari (bilangan bulat positif, +, X, S, A) sebagai karakter.
Masukan dijamin valid (+ dan X tidak mungkin ada jika catatan kosong).

Format Keluaran

Keluarkan jumlah total barang.

Contoh

No	Masukan	Keluaran
1.	5 52+X8	15
2.	7 241S8A2	9

Penjelasan

1. input = 52+X8, output = 15

"5" - Menambah 5 ke catatan, catatan = [5]
"2" - Menambah 2 ke catatan, catatan = [5, 2]
"+" - Menambah (5 + 2) ke catatan, catatan = [5, 2, 7]
"X" - Menghapus catatan sebelumnya, catatan = [5, 2]
"8" - Menambah 8 ke catatan, catatan = [5, 2, 8]

Total = 5 + 2 + 8 = 15

2. input = 241S8A2, output = 9

"2" - Menambah 2 ke catatan, catatan = [2]
"4" - Menambah 4 ke catatan, catatan = [2, 4]
"1" - Menambah 1 ke catatan, catatan = [2, 4, 1]
"S" - Menambah jumlah total ke catatan, hapus elemen sebelumnya, catatan = [7]
"8" - Menambah 8 ke catatan, catatan = [7, 8]
"A" - Menambah avg ke catatan, hapus elemen sebelumnya, catatan = [7]
"2" - Menambah 2 ke catatan, catatan = [7, 2]

Total = 9

Catatan

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

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.59 MB
2	10	Accepted	0.00 sec, 1.51 MB
3	10	Accepted	0.00 sec, 1.57 MB
4	10	Accepted	0.00 sec, 1.71 MB
5	10	Accepted	0.00 sec, 1.66 MB
6	10	Accepted	0.00 sec, 1.65 MB
7	10	Accepted	0.00 sec, 1.64 MB
8	10	Accepted	0.00 sec, 1.64 MB
9	10	Accepted	0.00 sec, 1.71 MB
10	10	Accepted	0.00 sec, 1.66 MB

Time limit	1 s
Memory limit	64 MB

Kurung

Deskripsi

Diberikan urutan **Kurung** dengan panjang N , Anda harus menentukan apakah urutan tersebut valid!

Urutan kurung yang valid didefinisikan secara rekursif sebagai:

- ""
- (x) dimana x adalah urutan kurung yang valid
- $[x]$ dimana x adalah urutan kurung yang valid
- $\{x\}$ dimana x adalah urutan kurung yang valid
- xy , dimana x dan y adalah urutan kurung yang valid

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($1 \leq N \leq 1000$).
- Baris kedua berisi N karakter yang merupakan urutan kurung dan hanya terdiri dari $()\{\}$ sebagai karakter.

Format Keluaran

Keluarkan "Valid" jika urutan valid, sebaliknya keluarkan "Invalid".

Contoh

No	Masukan	Keluaran
1.	4 ({})	Valid
2.	4 (}{)	Invalid

Penjelasan

Contoh kedua Invalid dikarenakan "}" bukan merupakan urutan kurung yang valid.

Catatan

- Keluaran diakhiri dengan newline ($\backslash n$)
- Download dan gunakan file [boolean.h](#), [stack.h](#) (terdapat perubahan pada infotype & MaxEl) & **stack.c** yang telah dibuat sebelumnya
- Download dan kumpulkan jawaban dalam file bernama [kurung.c](#)

C

 [kurung.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.54 MB
2	10	Accepted	0.00 sec, 1.50 MB
3	10	Accepted	0.00 sec, 1.51 MB

No	Score	Verdict	Description
4	10	Accepted	0.00 sec, 1.58 MB
5	10	Accepted	0.00 sec, 1.61 MB
6	10	Accepted	0.00 sec, 1.57 MB
7	10	Accepted	0.00 sec, 1.58 MB
8	10	Accepted	0.00 sec, 1.61 MB
9	10	Accepted	0.00 sec, 1.46 MB
10	10	Accepted	0.00 sec, 1.65 MB

Time limit	1 s
Memory limit	64 MB

Decode

Deskripsi

Diberikan string *encoded*, Anda harus mengembalikan hasil string yang *decoded*.

Aturan pengkodean adalah sebagai berikut:

- $k[\text{encoded_string}]$, dimana *encoded_string* yang ada di dalam tanda kurung siku diulang tepat k kali, $0 \leq k \leq 150$.

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($1 \leq N \leq 100$).
- Baris kedua berisi N karakter yang merupakan *encoded string*.

Format Keluaran

Satu baris *decoded* string, diakhiri newline.

Contoh

No	Masukan	Keluaran
1.	11 2[hmif]cool	hmifhmifcool
2.	8 2[a]3[b]	aabbbb
3.	8 3[a2[c]]	accaccacc

Catatan

- Download dan gunakan file [boolean.h](#), [stack.h](#) (terdapat perubahan pada infotype & MaxEl) & **stack.c** yang telah dibuat sebelumnya
- Download dan kumpulkan jawaban dalam file bernama [decode.c](#).
- Keluaran $0 \leq \text{length}(\text{result}) \leq 10000$
- Jika menggunakan array of char, gunakan '\0' untuk terminasi string. Misal ingin membuat string hmif, maka char s[5] = ['h', 'm', 'i', 'f', '\0']. Perhatikan 4 huruf perlu array of char sepanjang 5 karena harus ada terminator di akhir.



[decode.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	6	Accepted	0.00 sec, 1.57 MB
2	6	Accepted	0.02 sec, 1.54 MB
3	6	Accepted	0.00 sec, 1.52 MB

No	Score	Verdict	Description
4	6	Accepted	0.00 sec, 1.57 MB
5	6	Accepted	0.00 sec, 1.62 MB
6	6	Accepted	0.00 sec, 1.60 MB
7	6	Accepted	0.00 sec, 1.49 MB
8	6	Accepted	0.00 sec, 1.50 MB
9	6	Accepted	0.00 sec, 1.57 MB
10	6	Accepted	0.00 sec, 1.58 MB
11	10	Accepted	0.00 sec, 1.55 MB
12	10	Accepted	0.00 sec, 1.49 MB
13	10	Accepted	0.00 sec, 1.57 MB
14	10	Accepted	0.00 sec, 1.61 MB

Time limit	1 s
Memory limit	64 MB

Maxxing

[BONUS!!! (KERJAKAN YANG LAIN TERLEBIH DAHULU!)]

Deskripsi

Asep Spakbor memiliki N orang prajurit. Prajurit-prajurit ini diberi nomor dari 1 hingga N dari kiri ke kanan, dan masing-masing prajurit memiliki tinggi yang berbeda-beda. Tinggi prajurit ke- i adalah $a[i]$.

Sebuah kelompok prajurit adalah segmen berurutan dari barisan prajurit tersebut. Ukuran dari sebuah kelompok adalah jumlah prajurit di dalam kelompok itu. Kekuatan dari sebuah kelompok adalah tinggi prajurit paling rendah di kelompok tersebut.

Asep Spakbor ingin mengetahui, untuk setiap ukuran kelompok X (1 hingga N), berapa kekuatan maksimal di antara semua kelompok yang memiliki ukuran X .

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N ($1 \leq N \leq 10.000$).
- Baris kedua berisi N bilangan bulat (1 - 1.000.000.000) yang merepresentasikan list tinggi prajurit.

Format Keluaran

Keluarkan N bilangan bulat. Bilangan ke- X menunjukkan kekuatan maksimal untuk kelompok yang memiliki ukuran X .

Contoh

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

Penjelasan

Terdapat 4 orang prajurit yang dapat membentuk kelompok yang berukuran 1 - 4:

- Kelompok berukuran 1: $[[1], [2], [3], [1]] \rightarrow$ kekuatan = $[1, 2, 3, 1] \rightarrow$ kekuatan maksimal = 3.
- Kelompok berukuran 2: $[[1, 2], [2, 3], [3, 1]] \rightarrow$ kekuatan = $[1, 2, 1] \rightarrow$ kekuatan maksimal = 2.
- Kelompok berukuran 3: $[[1, 2, 3], [2, 3, 1]] \rightarrow$ kekuatan = $[1, 1] \rightarrow$ kekuatan maksimal = 1.
- Kelompok berukuran 4: $[[1, 2, 3, 4]] \rightarrow$ kekuatan = $[1] \rightarrow$ kekuatan maksimal = 1.

Catatan

- Keluaran diakhiri dengan newline ($\backslash n$)
- Download dan gunakan file [boolean.h](#), [stack.h](#) (terdapat perubahan pada MaxEl) & **stack.c** yang telah dibuat sebelumnya
- Download dan kumpulkan jawaban dalam file bernama [maxxing.c](#)



[maxxing.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.05 sec, 1.92 MB

No	Score	Verdict	Description
2	20	Accepted	0.05 sec, 1.89 MB
3	20	Accepted	0.05 sec, 1.88 MB
4	20	Accepted	0.04 sec, 1.75 MB
5	20	Accepted	0.05 sec, 1.74 MB

[◀ Stack - Praktikum](#)

Jump to...

⌵

[Queue - Pra Praktikum ▶](#)