

Started on	Wednesday, 13 November 2024, 3:11 PM
State	Finished
Completed on	Thursday, 14 November 2024, 2:39 PM
Time taken	23 hours 27 mins
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

Submit file [stack.c](#) yang merupakan hasil implementasi [stack.h](#)

C

 [stack.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, 1.64 MB
3	10	Accepted	0.00 sec, 1.61 MB
4	10	Accepted	0.00 sec, 1.50 MB
5	10	Accepted	0.00 sec, 1.59 MB
6	10	Accepted	0.00 sec, 1.70 MB
7	10	Accepted	0.00 sec, 1.64 MB
8	10	Accepted	0.00 sec, 1.49 MB
9	10	Accepted	0.00 sec, 1.64 MB
10	10	Accepted	0.00 sec, 1.63 MB

Time limit	1 s
Memory limit	64 MB

Buat program dengan nama **kalkulator.c** yang memanfaatkan [stack.h](#), [stackchar.h](#), dan [boolean.h](#).

## Deskripsi

Trump D. Mask adalah seorang matematikawan senior di United Colonies yang sedang mengembangkan kalkulator sederhana dengan fitur penyimpanan history. Keunikan kalkulator ini terletak pada kemampuannya untuk menyimpan setiap hasil perhitungan dan memiliki fitur "undo" untuk kembali ke hasil sebelumnya.

Kalkulator dapat membaca dan memproses operasi berikut:

- '+' : Melakukan operasi penjumlahan
- '-' : Melakukan operasi pengurangan
- 'u' : Melakukan undo (membatalkan operasi terakhir)
- '=' : Menampilkan hasil akhir perhitungan

## Batasan

- Jumlah operasi (n) <= 10^6
- Nilai operand berupa integer < 10^6

## Format Input

1. Setiap baris berisi satu operasi
2. Untuk operasi '+' dan '-', diikuti spasi dan sebuah bilangan
3. Operasi 'u' dan '=' tidak diikuti bilangan
4. Input diakhiri dengan karakter '='

## Format Output

Satu baris berisi hasil akhir perhitungan

## Contoh Input

Masukan:

```
+ 5
- 2
u
+ 5
- 3
=
```

Keluaran:

```
7
```



Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	5	Accepted	0.00 sec, 2.64 MB
2	5	Accepted	0.00 sec, 3.57 MB
3	5	Accepted	0.00 sec, 2.45 MB
4	5	Accepted	0.00 sec, 3.62 MB
5	5	Accepted	0.00 sec, 3.61 MB
6	5	Accepted	0.00 sec, 1.72 MB
7	5	Accepted	0.00 sec, 1.57 MB
8	5	Accepted	0.00 sec, 2.63 MB
9	5	Accepted	0.00 sec, 2.68 MB
10	5	Accepted	0.00 sec, 2.66 MB
11	5	Accepted	0.00 sec, 4.61 MB
12	5	Accepted	0.00 sec, 4.51 MB
13	5	Accepted	0.01 sec, 2.73 MB
14	5	Accepted	0.61 sec, 6.33 MB
15	5	Accepted	0.00 sec, 2.62 MB
16	5	Accepted	0.00 sec, 4.60 MB
17	5	Accepted	0.00 sec, 4.68 MB
18	5	Accepted	0.00 sec, 2.51 MB
19	5	Accepted	0.00 sec, 4.46 MB
20	5	Accepted	0.00 sec, 2.66 MB

Time limit	1 s
Memory limit	64 MB

# Minimum MinStack

Buat program dengan nama **MinStack.c** dengan bantuan [stack.c](#), [stack.h](#), dan [boolean.h](#)

## Deskripsi

Kamu adalah seorang kepala desa. Berbeda dengan desa pada umumnya, penduduk pada desa kali ini tinggal pada sebuah gedung. Setiap lantainya dihuni oleh seorang warga lainnya. Setiap warga memiliki nilai  $X$  yang menyatakan jumlah uang dari warga tersebut. Dikarenakan desamu yang unik, kamu menerima perintah khusus oleh [POI](#). Kamu menerima 3 tipe perintah oleh beliau. Berikut adalah detail perintah tersebut :

- Menambahkan warga baru. Warga baru yang datang akan tinggal pada lantai tertinggi dari gedung tersebut.
- Mengusir warga paling baru. Warga yang diusir adalah warga yang tinggal pada lantai atas. Apabila awalnya terdapat  $N$  orang warga yang tinggal di desa tersebut, maka hanya akan ada  $N - 1$  warga yang tinggal pada desa tersebut setelah warga pada lantai paling atas diusir.
- Mencari warga yang memiliki uang terbanyak pada desa tersebut. Kamu diminta untuk memberi tahu pada lantai berapa warga tersebut tinggal serta berapa banyak jumlah uang yang dimiliki.

Kamu diminta untuk mengeksekusikan semua perintah tersebut tanpa tanya. Lakukan perintah tersebut!.

## Format Masukan

Baris pertama diisi bilangan  $Q$  yang menyatakan berapa banyak perintah yang harus dijalankan.  $Q$  baris selanjutnya diisi oleh perintah dari [POI](#) dengan ketentuan sebagai berikut.

- 1  $X$  yang menyatakan untuk menambah warga baru dengan uang yang dimiliki sebesar  $X$ .
- 2 yang menyatakan untuk mengusir warga yang paling baru.
- 3 yang meminta untuk mencetak lantai dari warga yang memiliki uang terbanyak serta nilai dari uang tersebut.

## Format Keluaran

Setiap perintah nomor 3 akan dicetak dengan format sebagai berikut  $X \ Y \backslash n$  dimana  $X$  menyatakan lantai dari warga tersebut serta  $Y$  menyatakan nilai dari uang yang dimiliki warga tersebut.

## Batasan

- $1 \leq Q \leq 10^5$
- $-10^9 \leq X \leq 10^9$
- Dipastikan tidak ada perintah nomor 2 ataupun 3 ketika desa tersebut kosong.

## Contoh Input 1

Masukan:

6  
1 4  
1 2  
1 5  
3  
2  
3

Keluaran:

3 5  
1 4



# Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 2.03 MB
2	10	Accepted	0.73 sec, 1.88 MB
3	10	Accepted	0.03 sec, 2.09 MB
4	10	Accepted	0.64 sec, 1.88 MB
5	10	Accepted	0.36 sec, 2.04 MB
6	10	Accepted	0.27 sec, 2.09 MB
7	10	Accepted	0.35 sec, 2.06 MB
8	10	Accepted	0.82 sec, 2.10 MB
9	10	Accepted	0.66 sec, 2.05 MB
10	10	Accepted	0.70 sec, 2.09 MB

[◀ Feedback Form Praktikum 6](#)

Jump to...

◆