

Задача НОД 2

⌚ 4.5 sec. 💾 512 MB

Кой не обича машинен код? 😊

Даден е масив $a_1, a_2, \dots, a_N$, първоначално пълен със стойности 0. Дадени са и Q заявки от два типа:

1. Добави x към $a_l, a_{l+1}, \dots, a_r$.
2. Изведи $\sum_{i=l}^r a_i$.

Напишете програма, която отговаря на заявките.

Детайли по имплементацията

Програмата, която ще напишете, трябва да е написана на псевдомашинен код в текстов файл (txt). Вашият код може да използва посочените по-долу регистри, както и помощен масив $m_0, m_1, \dots, m_{10^6-1}$. Масивът и регистрите пазят 64-битови цели числа.

Нека означим $|R|$ като стойността на регистър R . Езикът поддържа следните инструкции:

IN R

Тази инструкция въвежда число в регистър R .

OUT R

Тази инструкция извежда числото в регистър R .

MOV R1 R2

Тази инструкция копира $|R2|$ в регистър $R1$.

MOV R1 x

Тази инструкция, използвана по този начин, запазва числото x в регистър $R1$.

ADD R1 R2

Тази инструкция добавя $|R2|$ към $|R1|$ и резултатът остава в $R1$.

SUB R1 R2

Тази инструкция изважда $|R2|$ от $|R1|$ и резултатът остава в $R1$.

DIV R1 R2

Тази инструкция целочислено дели $|R1|$ на $|R2|$ и резултатът остава в $R1$.

MUL R1 R2

Този инструкция умножава $|R1|$ по $|R2|$ и резултатът остава в $R1$.

CMP R1 R2

Този инструкция сравнява $|R1|$ и $|R2|$ и запазва резултата в регистър RES . Резултатът е 0, ако $|R1| > |R2|$ и 1, ако $|R1| \leq |R2|$.

LOAD R1 R2

Този инструкция зарежда $m_{|R1|}$ в регистър $R2$.

STORE R1 R2

Този инструкция запазва $|R2|$ в $m_{|R1|}$.

JZ N

Този инструкция казва на интерпретатора да отиде на ред N във Вашия код, единствено ако $|RES| = 0$.

JNZ N

Този инструкция казва на интерпретатора да отиде на ред N във Вашия код, единствено ако $|RES| \neq 0$.

RETURN

Този инструкция прекратява изпълнението на програмата.

; comment

MOV A B ; comment

Можете да използвате коментари в кода си, използвайки точка и запетая.

- $R, R1, R2$: регистър. Може да бъде A, B, C, D, E, F, G, H или RES . Имате право да пишете във всички от тях.
- N : номер на ред в кода. Индексацията започва от 1.

Вход

На първия ред от входа се въвеждат N и Q . На всеки от следващите Q реда се въвеждат t, l, r , където t е типът на заявка. Ако $t = 1$, на края на реда се въвежда и числото x .

Изход

За всяка заявка от тип 2 се извежда отговорът.

Ограничения

- $1 \leq N, Q \leq 3 \cdot 10^4$
- $1 \leq l \leq r \leq N$
- $1 \leq x \leq 10^9$
- $1 \leq t \leq 2$

Подзадачи

Подзадача	Точки	Необходимите подзадачи	Допълнителни ограничения
1	10	—	$N = 1, Q \leq 500$
2	35	1	$N, Q \leq 500$
3	60	1	За заявки от тип 1, $l = r$.
4	45	1 – 3	—

Примерна програма

Нека разгледаме примерна програма на този език, която въвежда две числа и извежда сумата им:

```
IN A
IN B
ADD A B
OUT A
```

Да разгледаме друга примерна програма, която въвежда две числа и извежда по-голямото от тях:

```
IN A
IN B
CMP A B
JZ 6
JNZ 8
OUT A
RETURN
OUT B
RETURN
```

Примерен грейдър

Предоставеният грейдър чете кода от файл `nod2.txt` и го изпълнява.