

**ТРЕТО КОНТРОЛНО СЪСТЕЗАНИЕ НА
РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ЯМБОЛ, 6 май, 2018 г.
Група А, 11-12 клас**

ЗАДАЧА АК1. ЕТАЖЕРКА

В стаята на Пешо има етажерка с N тетрадки, които съдържат описание на всички гениални идеи, които витаят в неговата глава. Тетрадките са номерирани с числата от 1 до N . Пешо си има любимо подреждане на тетрадките върху етажерката, което не е задължително да бъде по нарастване на номерата им, и той никак не обича някой да му ги размества. Пешо е купил специален робот, който може да запомня подреждането на тетрадките и да пресмята броя на инверсиите в дадено подреждане. Роботът смята, че две тетрадки образуват инверсия, ако тетрадката с по-малък номер стои по-надясно от тетрадката с по-голям номер. Например в подреждането (2,1,5,3,4) има три двойки тетрадки, които образуват инверсия (2,1), (5,3) и (5,4), т.е. броят на инверсиите в това подреждане е 3.

За нещастие, след ремонт на стаята, тетрадките върху етажерката били разместени, а Пешо забравил любимото си подреждане. Роботът го помни, но той може да съобщава само броя на инверсиите в запомненото от него подреждане. Пешо може да поиска от робота да размени местата на две тетрадки в подреждането, което роботът помни в момента. След такава размяна, роботът ще запомни новото подреждане (като забрави старото) и ще съобщи броя на инверсиите в него. Пешо може да повтаря заявките си към робота до момента, в който реши, че има достатъчно информация, за да възстанови любимото си подреждане на тетрадките.

Вие ще изпълнявате ролята на Пешо, а програма на журито – ролята на робота.

Задача

Напишете функция *bookshelf()*, която ще се компилира с програма на журито и ще и задава въпроси докато възстанови първоначалната подредба на книгите..

Детайли по реализацията

Функцията *bookshelf()* трябва да има следния формат:
`void bookshelf(int N, long long Inv);`

Тя се вика веднъж от програмата на журито с аргументи N и Inv – броя на тетрадките и броя на инверсиите в първоначалната им подредба.

За комуникация с програмата на журито Ви се предоставят функциите:

`long long bookswap(int i, int j)`

и

`void answer(int p[])`

Чрез извикване на функция *bookswap()* Вие подавате заявка към програмата на журито (робота) да размени местата на тетрадките, които се намират на позиции i и j и да Ви върне броя на инверсиите в новото подреждане на тетрадките.

Когато Вашата програма реши, че е намерила първоначалното подреждане на тетрадките, тя трябва да извика функция *answer()*, като в масив p предаде на програмата на журито намереното подреждане – елементът $p[i]$ трябва да съдържа номера на тетрадката, която се намира на позиция с номер i .

Вие трябва да предадете към системата файл **bookshelf.cpp**, който съдържа функция *bookshelf()*. Той може да съдържа и друг код, необходим за работата на функция *bookshelf*, но не трябва да съдържа *main()*.

В началото си Вашият файл трябва да съдържа `#include "bookshelf.h"`.

**ТРЕТО КОНТРОЛНО СЪСТЕЗАНИЕ НА
РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ЯМБОЛ, 6 май, 2018 г.
Група А, 11-12 клас**

Ограничения

$$2 \leq N \leq 100\,000$$

$0 \leq$ брой на заявките към робота за разместване на тетрадки $\leq 200\,000$

В 10% от тестовете $2 \leq N \leq 400$;

В други 20% от тестовете $400 < N \leq 5\,000$;

В други 30% от тестовете $5\,000 < N \leq 50\,000$;

Оценяване

Всеки тест се оценява отделно.

Примерна комуникация с програмата на журито

Нека първоначалното подреждане на тетрадките, което Вашата програма трябва да познае, е 4 3 1 5 2. Броят на инверсиите в това подреждане е 6 – (4,3), (4,1), (4,2), (3,1), (3,2) и (5,2). Програмата на журито ще извика Вашата функция *bookshelf* така:

bookshelf(5,6);

Един от възможните диалози след това е:

Функция на журито, която викате	Резултат, който получавате
<i>bookswap</i> (1,2)	5
<i>bookswap</i> (1,2)	6
<i>bookswap</i> (3,2)	5
<i>bookswap</i> (3,2)	6
<i>bookswap</i> (5,4)	5
<i>bookswap</i> (5,4)	6
<i>bookswap</i> (4,1)	7
<i>bookswap</i> (4,1)	6
<i>bookswap</i> (5,1)	3
<i>bookswap</i> (5,1)	6
<i>bookswap</i> (5,2)	5
<i>bookswap</i> (5,2)	6
<i>bookswap</i> (5,3)	7
<i>bookswap</i> (5,6)	6
<i>answer</i> (4 3 1 5 2) Тук като параметър на <i>answer</i> , разбира се, стои масив.	С това работата на програмата приключва – Вашата програма даде верен резултат.

Локално тестване

За да можете да тествате Вашата функция *bookshelf()* на локалния си компютър, Ви се предоставят файлове *Lgrader.cpp* и *bookshelf.h*. Компилирайте ги заедно с вашия файл **bookshelf.cpp** и ще получите програма, с която да тествате функцията си. Програмата изисква от стандартния вход да се въведе следната последователност от данни:

От първи ред: две цели положителни числа – брой на тетрадките и брой на инверсиите в първоначалното подреждане.

От втори ред: номерата на тетрадките в първоначалното подреждане, разделени с интервали.

На изход ще получите първоначалното подреждане, което Вашата програма е „познала“.

**ТРЕТО КОНТРОЛНО СЪСТЕЗАНИЕ НА
РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ЯМБОЛ, 6 май, 2018 г.
Група А, 11-12 клас**

ЗАДАЧА АК2. ЗАЯВКИ

Дени реши да става самостоятелна, затова започна държавна работа като чиновник. В началото тя разполага с празна таблица (всяка клетка е със стойност 0) с 10^5 реда и N колони. Номерацията на редовете и колоните започва от 1.

През целия ден при Дени постъпват три вида заявки:

- **Заявки от първи вид:**

Задават се с номер на ред, начална колона *from*, крайна колона *to* и две допълнителни числа *st* и *d*, които задават съответно началната стойност и разликата за аритметична прогресия (*вижте и забележката по-долу*). Всеки от първите ($to - from + 1$) елемента от тази редица трябва да се добави към текущата стойност на съответната клетка от колона *from* до колона *to* на този ред.

- **Заявки от втори вид:**

Задават се с номер на ред, начална колона *from* и крайна колона *to* и се иска да се намери сборът от текущите стойности в клетките от колона *from* до колона *to* на този ред.

- **Заявки от трети вид:**

Задават се с две числа – номер на два реда, като Дени трябва да копира цялото съдържание **на първия от тях във втория**.

Помогнете на момичето, като напишете програма **queries**, която обработва пристигащите заявки.

Забележка: Аритметична прогресия с начална стойност st и разлика d се нарича числовата редица $\{a_1, a_2, a_3, \dots, a_k, \dots\}$, която се задава по следния начин: $a_1 = st$ и $a_k = a_{k-1} + d$ за $k \geq 2$.

Вход

От първия ред на стандартния вход се въвеждат целите положителни числа N и Q , разделени с интервал – брой колони и брой заявки. На следващите Q реда се въвеждат заявките.

Всяка заявка се задава с няколко цели числа, разделени с интервал.

- Ако първото число на реда е 1, то заявката е от първи вид и на реда има още пет числа – *row*, *from*, *to*, *st* и *d* (съответно номер на реда, начална и крайна колона и начална стойност и разлика на аритметичната прогресия).
- Ако първото число на реда е 2, то заявката е от втори вид и на реда има още три числа *row*, *from* и *to* (съответно номер на реда, начална и крайна колона).
- А ако първото число на реда е 3, то заявката е от трети тип и на реда има още две числа row_1 и row_2 (съответно номер на реда, **от който** трябва да се копира цялото съдържание и номер на реда, **в който** трябва да се копира).

Изход

За всяка заявка от втори тип програмата трябва да изведе на стандартния изход един ред с намерения сбор.

Ограничения

- ♣ **Максималната разрешена памет е 16 МВ.**
- ♣ $1 \leq N \leq 10^5$

**ТРЕТО КОНТРОЛНО СЪСТЕЗАНИЕ НА
РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ЯМБОЛ, 6 май, 2018 г.
Група А, 11-12 клас**

Броят на редовете в таблицата е 10^5 . Началните стойности и разликите на аритметичните прогресии са цели числа в интервала от -10^5 до 10^5 .

- ♣ $1 \leq Q \leq 10^5$
- ♣ За всички заявки, в които се съдържат начална колона *from* и крайна колона *to*, е изпълнено $1 \leq from \leq to \leq N$.

Подзадачи

Подзадача	Точки	N	Q	Други ограничения
1	20	$\leq 10^4$	$\leq 10^5$	Няма заявки от трети тип. Всички заявки са за един и същи ред.
2	10	$\leq 10^5$	$\leq 2 \cdot 10^4$	Няма заявки от трети тип.
3	15	$\leq 10^4$	$\leq 10^5$	Няма заявки от трети тип.
4	25	$\leq 10^5$	$\leq 2 \cdot 10^4$	Няма допълнителни ограничения.
5	30	$\leq 10^4$	$\leq 10^5$	Няма допълнителни ограничения.

Точките за дадена подзадача се получават, ако бъдат преминати успешно всички тестове за нея.

Пример

Вход	Изход	Обяснение на примера										
10 15	1	След първата заявка в първия ред на таблицата има следните числа:										
1 1 1 5 1 2	9	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	3	5	7	9	0	0	0	0	0
1	3	5	7	9	0	0	0	0	0			
2 1 1 1	25											
2 1 5 5	0	След копирането (заявка № 6), вторият ред се изпълва със същите числа. След извършването на заявка за промяна върху него, вторият ред е с тези числа:										
2 1 1 10	25	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td></tr></table>	1	3	5	7	9	1	3	5	7	9
1	3	5	7	9	1	3	5	7	9			
2 2 1 10	50											
3 1 2	25											
2 2 1 10	10	Забележете, че първият ред не се променя!										
1 2 6 10 1 2	50	След изпълняване на заявка № 12, първите два реда на таблицата изглеждат така:										
2 2 1 10	0	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>-11</td><td>-13</td><td>-15</td><td>-17</td><td>-19</td></tr></table>	0	0	0	0	0	-11	-13	-15	-17	-19
0	0	0	0	0	-11	-13	-15	-17	-19			
2 1 1 10	-75	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td></tr></table>	1	3	5	7	9	1	3	5	7	9
1	3	5	7	9	1	3	5	7	9			
2 2 5 6												
1 1 1 10 -1 -2												
2 2 1 10												
2 1 1 5												
2 1 6 10												

**ТРЕТО КОНТРОЛНО СЪСТЕЗАНИЕ НА
РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ЯМБОЛ, 6 май, 2018 г.
Група А, 11-12 клас**

ЗАДАЧА АКЗ. „ЗАПАДЕН СВЯТ“

Ели с интерес следи сериала Westworld („Западен Свят“). В новия сезон са показани N пълни години от живота в бъдещето, като всяка сцена представлява случка през някой месец от тези N години. Всеки месец от всяка година е показан точно по веднъж – тоест сезонът има точно $12 * N$ сцени (по една за всеки месец от показаните N години).

Редът на сцените, описващи месеците за всяка година, е запазен, т.е., преди сцената за Май 2058 вече е излъчена тази за Април 2058, а някъде след нея ще бъде излъчена сцената за Юни 2058, като между трите сцени **не може да има друга сцена от 2058 година**. Интригуващото в сериала е, че макар и всяка от годините да върви последователно, самите години са размесени една с друга. С други думи, възможен е, например, такъв сценарий: първо да са показани Януари, Февруари и Март от третата година, после Януари от осмата, после Януари и Февруари от първата, после Април от третата, после Януари до Юни от десетата и т.н.

Ели гледа внимателно сериала и отбелязва за всяка сцена дали има нещо, което показва кое време е. Продуцентите на сериала са се постарали нищо да не издава годините, но пък момичето може да отгатне част от месеците: например, ако се вижда календар, на който е показано кой месец е; или пък е Коледа, от което момичето заключава, че е Декември и т.н.

След като приключват всички сцени, Ели осъзнава, че е записвала месеците само с първите им букви: 'J' за Януари, 'F' за Февруари, 'M' за Март, 'A' за Април, 'M' за Май, 'J' за Юни, 'J' за Юли, 'A' за Август, 'S' за Септември, 'O' за Октомври, 'N' за Ноември и 'D' за Декември. Месеците, за които не е откривала индикатори, е отбелязвала с '?'.

Сега Ели се чуди колко възможни „сюжета“ може да има сериалът. Под „сюжет“ ще разбираме различна последователност от двойки (месец, година), които отговарят на нейните записки. Например, примерът от втория параграф отговаря на J?MJJ?A?? MAMJ, но има и много други възможни „сюжети“ – един от тях би бил същия като горния, но последното 'J' да отговаря за Януари на някоя от „незапочнатите“ години, вместо Юни от десетата. Сега момичето ви моли да ѝ помогнете, като определите колко възможни сюжета би могъл да има сериалът.

Вход

На първия ред на стандартния вход ще бъде зададен броят години N . На втория ред ще има стринг S с $12 * N$ символа, всеки от които ще бъде от множеството {'J', 'F', 'M', 'A', 'S', 'O', 'N', 'D', '?'} – той представлява записките на Ели.

Изход

На стандартния изход програмата трябва да изведе един ред, съдържащ само едно цяло число – броя различни „сюжети“, които би могъл да има сериалът. Тъй като числото може да е много голямо, изведете само остатъка му при деление на 1 000 000 007.

Ограничения

- ❖ $1 \leq N \leq 13$
- ❖ $|S| = 12 * N$

**ТРЕТО КОНТРОЛНО СЪСТЕЗАНИЕ НА
РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ЯМБОЛ, 6 май, 2018 г.
Група А, 11-12 клас**

Примери

Вход	Изход
2 J??FMAMM?J?JJ???N?A?D???	72
4 ?J?FMFFMA??JJAA???J????ASM??A?JJMJJA?S00?DSON? ND	822595037

Коментари по примерите

В първия пример първото 'J' може да е както от първата година, така и от втората (**2 варианта**). Следват две въпросителни, които могат да са или "JF" или "FJ". Ако са "JF", обаче, се образуват две последователни 'F'-а, които могат да са Февруари месец на която и да е от двете години (**3 варианта**). Следват Март и Април на някоя от двете години – тъй като и двете са до Февруари, не знаем коя от тях е, така че имаме още **2 варианта**. Двете последователни 'M'-та могат да са или (Март, Май) или (Май, Март) – още **2 варианта**. Въпросителната след тях е еднозначно определена – тя не може да е нищо друго освен 'A' (Април), тъй като иначе няма как да направим двете последователни 'J'-та по-нататък. Четвъртата въпросителна също не може да е нищо друго освен 'M'. Така двете последователни 'J'-та могат да са или (Юни, Юли) или (Юли, Юни). Забележете, че тук ако са (Юни, Юли) бихме могли да имаме Юли както на едната, така и на другата година - тоест тук имаме нови **3 варианта**. Следват три въпросителни, които, обаче, трябва да са "ASO", за да може следващият месец да е Ноември. Въпросителната преди последното 'A' трябва да е 'J', тъй като 'A'-то в този момент не може да е нищо друго освен Август. Тъй като знаем къде е и последният месец от по-напредналата година ('D'-то), то останалите въпросителни са от другата година. Така имаме $2 * 3 * 2 * 2 * 3 = 72$ сюжета.

Във втория пример има 5 822 595 072 възможни сюжета. Остатъкът при деление на 5822595072 на 1000000007 е 822595037, което е и отговорът за този тест.