Упражнения: Масиви

Тествайте решението в Judge: https://judge.softuni.bg/Contests/3169/Arrays.

1. Влак

Ще Ви бъде даден броя на вагоните n. На следващите n редове ще бъде посочен броя на хората във всеки вагон. Отпечатайте броя на хората във всеки вагон и на следващият ред, общият брой на хората във влака.

Примерен вход и пример

Вход	Изход				
3 13 24 8	13 24 8 45				
6 3 52 71 13 65 4	3 52 71 13 65 4 208				
1 100	100 100				

2. Зиг-Заг масиви

Напишете програма, която създава 2 масива. Ще Ви бъде даден число n. На следващите n редове ще получавате по 2 числа. От 2 масива както е показано по-долу.

Примерен вход и пример

Вход	Изход
4 1 5 9 10 31 81 41 20	1 10 31 20 5 9 81 41
2 80 23 31 19	80 19 23 31

3. Завъртане на масиви

Напишете програма, която получава масив и брой ротаций, които трябва да изпълните (първият елемент отива накрая). Отпечатайте получения масив.











Примерен вход и пример

Вход	Изход			
51 47 32 61 21 2	32 61 21 51 47			
32 21 61 1 4	32 21 61 1			
2 4 15 31 5	4 15 31 2			

4. Топ числа

Напишете програма, която намира топ числата в масива. Топ число е най-голямото число спрямо другите елементи отдясно.

Примерен вход и пример

Вход	Изход		
1 4 3 2	4 3 2		
14 24 3 19 15 17	24 19 17		
27 19 42 2 13 45 48	48		

Еднакви суми

Напишете програма, която определя дали съществува елемент в масива, така че сумата на елементите отляво да е равна на сумата на елементите отдясно (може да има не само един 1 такъв елемент). Ако няма елементи от ляво/дясно, тяхната сума се зачита за 0. Отпечатайте индекса, който отговаря на условието, или "не" ако няма такъв индекс.

Примерен вход и пример

Вход	Изход	Обяснения
1 2 3 3	2	В a[2] -> <mark>лява сума = 3</mark> , дясна сума = 3 a[0] + a[1] = a[3]
1 2	no	В a[0] -> лява сума = 0, дясна сума = 2 В a[1] -> лява сума = 1, дясна сума = 0 Не съществува такъв индекс
1	0	В а[0] -> лява сума = 0, дясна сума = 0
1 2 3	no	Не съществува такъв индекс
10 5 5 99 3 4 2 5 1 1 4	3	В a[3] -> <mark>лява сума = 20</mark> , дясна сума = 20













a[0] + a[1] + a[2] = a[4] + a[5] + a[$a[7] + a[8] + a[9] + a[10]$	6] +
a[/] + a[o] + a[ɔ] + a[ɪʊ]	

6. Максимална последователност от равни елементи

Напишете програма, която намира най-дългата еднаква редица от елементи в масива от числа. Ако има няколко такива редици отпетайте най-лявата

Примерен вход и пример

Вход						Изход				
2	1	1	2	3	3	2	2	2	1	2 2 2
1	1	1	2	3	1	3	3			1 1 1
4	4	4	4							4 4 4 4
0	1	1	5	2	2	6	3	3		1 1

7. *Фабрика Камино

Клониращата фабрика в Камино има поръчка за клониране на войски. Вашата задача е да намерите найдобрата ДНК последователност, която да се използва за продукцията.

Ще получите ДНК дължина докато не получите командата "Clone them!". Ще Ви бъде дадена ДНК редица от единици и нули разделени чрез "!"(един или няколко).

Вие трябва да изберете последователността с най-дългата редица. Ако има няколко редици с еднаква дължина на последователност от единици, отпечатайте тази с най-левия начален индекс, ако има няколко дължини с еднаква дължина и еднакъв първоначален индекс, изберете подредицата с най-голямата сума от нейните елементи.

След като получите последната команда "Clone them!" трябва да отпечатате събраната информация в следния формат:

"Best DNA sample {най-добърият индекс на редицата} with sum:{най-добърата сума на редицата . "

"{ДНК редицата, разделена по празно място}"

Вход / Ограничения

- Първият ред съдържа дължината на редиците-число в обхвата [1...100];
- На следващите редове доката не получите "Clone them!" Вие ще получавате редеци (поне една) от единици и нули разделени чрез "!" (един или няколко).

Изход

Изхода трябва да отпечатан на конзолата и да съдържа два реда със следният формат:

"Best DNA sample {редицата с най-добър индекс} with sum: {редицата с най-голяма сума}" "{ДНК редицата, разделена с интервал}"

Примерен вход и пример

Вход	Изход	Обяснения













5 1!0! <mark>1!1</mark> !0 0! <mark>1!1</mark> !0!0 Clone them!	Best DNA sample 2 with sum: 2. 0 1 1 0 0	Получаваме 2 редици с еднаква дължина на подредици от единици , но се отпечатва втората защото подредицата стартира от индекс[1].
Вход	Изход	Обяснения
4 1!1!0!1 1!0!0!1 1!1!0!0 Clone them!	Best DNA sample 1 with sum: 3. 1 1 0 1	Получаваме 3 редици. Първата и третата имат еднаква дължина от подредици от единици ->, и двете стартират от индекс[0], но се принтира първата, защото й сумата е по-голяма.

8. *Какалинки

Дадено Вие поле с размер и индекси на калинки в полето. След това на всеки нов ред до подаване на командата "end", калинката променя позицията наляво или надясно с дадена дължина.

Командата на калинката изглежда така: "O right 1". Това означава че малкото насеко е на индекс 0 и трябва да прелети един индекс надясно. Ако калинката кацне на друга калинка, тя продължава да лети в същата директория със същата посока. Ако калинката излети извън полето, тя изчезва.

Например, представете си, че Ви е дадено поле с размер 3 и има калинка на индекс 0 и 1. Ако калинката на индекс 0 трябва да прелети надясно с дължина 1 (0 right 1) ще се опита да кацне на индекс 1, но тъй като там има друга калинка, тя ще продължи по-надясно с допълнителна дължина 1, кацайки на индекс 2. След това, ако същата калинка трябва да лети надясно с дължина 1 (2 right 1), тя ще кацне някъде извън полето, така че отлита:



Ако Ви бъде даден индекс на калинка, където няма нищо не се случва. Ако Ви бъде даден индекс на калинка извън полето, нищо не се случва.

Вашата работа е да напишете програма, която симулира летенето на калинките. Накрая отпечатвайте всички **клетки в полето, разделени с празен интервал**. За всяка клетка, в която има калинка, отпечатайте " $\mathbf{1}$ ", а за всяка празна клетка отпечатайте "0". За примера по-горе изходът трябва да бъде '0 1 0'.

Вход

- На първият вход ще получавате число размерът на полето
- На вторият ред ще получавате **индексите** на калинките разделени с празно място. **Индиксите могат** да бъдат или да не бъдат в полето
- На следващите редове докато не получите командата "end", Вие получавате команди в следният формат: "{индекс на калинката} {директория} {дължина на летене}"
- Отпичатайте всички клетки със следният формат: "{клетка} {клетка} ... {клетка}"
 - Ако в клетка има калинка, отпечатайте '1'
 - Ако клетка е празна, отпечатайте '0'

Ограничения

- Размерът в полето варира в обхвата [0 ... 1000]
- Индиксите на калинките ще бъдат в обхавата [-2,147,483,647 ... 2,147,483,647]
- Номерът на командите ще бъдат в обхвата [0 ... 100]
- Дължината на полетите ще бъдат в обхвата [-2,147,483,647 ... 2,147,483,647]



















Примерен вход и пример

Вход	Изход	Обяснения
3	010	1 1 0 - първоначално поле
0 1		0 1 1 - полето след "0 right 1"
0 right 1		0 1 0 - полето след "2 right 1"
2 right 1		
end		

Вход	Изход		
3	000		
0 1 2			
0 right 1			
1 right 1			
2 right 1			
end			

Вход	0	бяс	сне	Нν	1Я
5	0	0	0	1	0
3					
3 left 2					
1 left -2					
end					















