**Задача С1. Инверсии**

**Д**ени днес научи в училище какво е инверсия в пермутация – казваме, че и с *i* < *j* за пермутацията , , …, са в инверсия, ако > . Тя започнала да пише пермутации и да брои колко инверсии имат. Бързо и доскучало, затова решила да преброи колко пермутации на ***N*** елемента имат ***K*** на брой инверсии. Това се оказала нелека задача, затова напишете програма **inversions**, с която да помогнете на **Д**ени.

**Вход**

От първият ред на стандартния вход се въвеждат две числа: ***N*** – броят елементи и ***K***– броят инверсии.

**Изход**

На един ред на стандартния изход изведете едно единствено число – търсеният брой пермутации. Понеже този брой може да е много голям, изведете само остатъка му при деление с 109+7.

**Ограничения**

* 1 ≤ ***N***≤103
* 1 ≤ ***K***≤104
* Максималната разрешена памет е 32 МВ.

**Оценяване**

* в 10% от тестовете ***N***≤10
* в 60% от тестовете ***N***≤200
* в 80% от тестовете ***N*** ≤ 300

**Примери**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 10 1 | 9 |
| 4 3 | 6 |
| 9 13 | 17957 |

**Задача С2. Гориво**

**Д**ени гледа да разнообразява пътя си до училище, и се чуди по колко различни начина може да стигне до там изминавайки минимален път. Можем да представим пътя й като права отсечка, по която има **N** различни точки (бензиностанциите, на които тя може да спира и презарежда). За съжаление родителите ѝ не са ѝ подарили особено хубава кола (ще цитираме **Д**ени: „Много бързо свършва резервоарът на пустия Вейрон!”) и не винаги е възможно тя да стигне от една точка до всяка друга. Всеки интересен обект можем да бележим с разстоянието му в километри от началото на отсечката (където е нейният дом). Така домът й бива отбелязан с 0, училището с ***К***, а с , , …, - бензиностанциите. При пълен резервоар **Д**ени може да измине **L** километра. Възможно е и Ели да не спира никъде, ако това е валиден път (тоест ако **К** ≤ ***L***). Напишете програма fuel, с помощта на която **Д**ени ще може да извършва необходимите пресмятания.

**Вход**

На първия ред на стандартния вход ще бъдат зададени целите числа **N**, **К** и **L**, разделени с интервали. На следващия ред ще има **N** числа - , , …, , отбелязващи разстоянията от началото на отсечката до всяка от бензиностанциите, подредени във възходящ ред.

**Изход**

На един ред на стандартния изход изведете едно единствено число – търсеният брой начини **Д**ени да стигне до училище. Понеже този брой може да е много голям, изведете само остатъка му при деление с 109+7.

**Ограничения**

* 1 ≤ ***N***≤106
* 1 ≤ ≤ ≤ … ≤ ≤ **К**, **L** ≤ 109

**Примери**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснение на примера** |
| 3 10 7  3 5 9 | 6 | (0, 3, 10); (0, 5, 10); (0, 3, 5, 10); (0, 3, 9, 10); (0, 5, 9, 10), (0, 3, 5, 9, 10) |
| 5 13 7  1 2 3 4 5 | 0 | Няма да има гориво дори да заради на бензиностанция 5. |
| 42 66 13  1 3 4 5 7 9 10 11 14 27 28 29 30 31 32 33  34 35 36 37 38 39 40 41 42 44 46 47 48 49  51 53 54 56 57 58 59 60 61 62 64 65 | 188682339 | Тук е изведен само остатъка на отговора при деление с 109+7. |

**Задача С3. Игра**

На всяка от ***N*** на брой карти е написано по едно цяло положително число, което е по-малко от 1000. Картите са подредени една до друга. **Д**ени е измислила нова игра. В нея се извършват ***N*** - 2 на брой хода. При всеки ход се премахва една от картите в редицата без първата и последната. За всяка отстранена карта се получават точки, които се изчисляват, като се умножи числото, записано тази карта със сбора от числата, записани на съседните две. Накрая остават само две карти. Напишете програма **game**, която намира най-големия брой точки, които могат да се получат при тази игра.

**Вход**

На първия ред на стандартния вход е зададено числото ***N***. На следващия ред са зададени числата, написани върху картите, като е спазена подредбата им в редицата.

**Изход**

На един ред на стандартния изход изведете едно единствено число – максималният брой точки, които могат да се получат при изиграване на описаната игра.

**Ограничения**

* 3 ≤ ***N***≤700

**Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 4  4 5 6 2 | 86 |

**Задача С4. Скрити думи**

**Д**ени има правоъгълна таблица от букви с ***N*** реда и ***M*** колони. Крис й е казала, че е „скрила” в таблицата някои от дадени ***K*** думи. Една дума от *S* символа се счита за скрита в таблицата ако съществува последователност от *S* съседни (потенциално повтарящи се) клетки, чиито символи образуват думата. Две клетки са съседни ако имат обща страна (тоест са съседни по хоризонтала или по вертикала). Помогнете на **Д**ени, като напишете програма **hidden**, която намира кои от дадените думи се срещат в таблицата и кои – не.

**Вход**

На първия ред на стандартния вход ще бъдат зададени целите числа ***N*** и ***M***. Следват ***N*** реда, всеки от които съдържа последователност от ***M*** на брой малки букви от английската азбука. След тях следва цялото число ***K*** - броят на потенциално скритите в таблицата думи. Следват ***K*** на брой думи , , *…*, от същата азбука.

**Изход**

За всяка от думите изведете на стандартния изход по един ред, съдържащ *"Yes"* или *"No"*, в зависимост дали съответната дума е скрита в таблицата или не.

**Ограничения**

* 1 ≤ ***N****,* ***M****,* ***K****,* ≤50
* В 50% от тестовете таблицата ще е генерирана на случаен принцип

**Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 5 6  glmnta  csoikf  dgotzf  hcnats  oemcpw  7  nato  god  meh  cat  catod  gg  stanchoemnogogotin | Yes  No  No  Yes  No  No  Yes |