# Упражнения: Стрингообработка

Тествайте решението в Judge: <https://judge.softuni.org/Contests/4164/06-String-Processing-Advanced>.

## Умножение на големи числа

Дадени са ви **два реда** - на **първия** ще получите **число** (от 0 до 1050). На втория ред ще получите **цифра** (от 0 до 9). Трябва да отпечатате произведението на **числото от първия и втория ред**.

### Бележки

Не използвайте класа **BigInteger**. Приложете умножението само чрез операции с низове и прости изчисления в цикъл.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 23  2 | 46 |
| 9999  9 | 89991 |
| 92384723893198319246283210  4 | 3695388955727932769851328408 |

### Насоки

1. Прочетете двете числа от конзолата.
2. Създайте променливи reversedResult от тип **низ** и residue от тип int.
3. Създайте for-цикъл, който обхожда от **дължината на голямото число -1** до **0**.
4. В него създайте променлива, която има стойност **умножението на цифрата** с **позиция** i на for**-цикъла** **на голямото число** **\*** другото число.
5. Ако резултата е **по-голям** или **равен** на **10**, сложете стойност на residue **остатъка** на **резултата**. В противен случай, задайте стойност **0** на residue.
6. След **цикъла** проверете дали residue е различно от **0**. Ако е така, към reversedResult добавете residue.
7. Създайте две променливи result и isZero от **тип** bool.
8. След това създайте for**-цикъл**, който обхожда от **дължината на** reversedResult до **0**.
9. В него проверете дали reversedResult от i е **0** и дали isZero има стойност true.
10. Ако е така, прескочете стъпката на for**-цикъла**.
11. След това задайте стойност на isZero false.
12. Ако isZero е със стойност true, добавете към result reversedResult от i.
13. След **цикъла** проверете дали result е **празен**.
14. Ако е така, към result добавете **0**.
15. Накрая отпечатайте result.

## Експлозии

Напишете програма, която **прочита низ**. В него ще получите **експлозии**, маркирани с **'>'**. След знака ще получите число, което показва **дължината на експлозията**.

Трябва да премахнете **x символа** (където x е дължината на експлозията), започвайки след знака **'>'**.

Ако намерите **още една експлозия**, докато премахвате символи, трябва да я **добавите** към дължината на **предишната експлозия**.

Накрая трябва да **изтриете дължините** на експлозиите.

### Вход

Ще получите само **един ред** с низ.

### Изход

Отпечатайте какво е **останало след експлозията**.

### Бележки

* Винаги ще получавате **сила на експлозията**
* Пътят ще се състои само от букви от **латинската азбука**, **цели числа** и **символа** '>'
* Силата на ударите ще бъде в интервала [0…9]

### Примери

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| abv>1>1>2>2asdasd | abv>>>>dasd | Първата експлозия е на индекс 3 и има дължина 1. Премахваме само цифрата. Нашият низ изгледжда така: abv>>1>2>2asdasd. Втората експлозия има дължина 1. Премахваме само цифрата. Нашият изглежда така:abv>>>2>2asdasd. Третата експлозия има дължина 2. Премахваме цифрата. До този момент нашият масив изглежда така:abv>>>>2asdasd. Четвъртата експлозия има дължина 2. От предишната експлозия имаме останала дължина 1. Прибавяме я към сегашната. Изтриваме следващите три символа и получаваме abv>>>>dasd |
| pesho>2sis>1a>2akarate>4hexmaster | pesho>is>a>karate>master |  |

## \*Буквите променят числата

Наков обича математиката, но в същото време обича английската азбука. Затова е решил да създаде игра с цифри и букви на английски. Играта е много лесна. Трябва да получите низ, който се състои от **число между два символа**. В зависимост от това каква е буквата пред числото или след него, ще извършите различни математически операции върху числото, за да постигнете резултата.

**Първо** трябва да започнете от **буквата преди числото**:

* Ако е **главна буква,** трябва да **разделите** числото на **позицията на буквата в азбуката**.
* Ако е **малка буква,** трябва да **умножите** числото с **позицията на буквата в азбуката**.

След това започваме с буквата след числото.

* Ако е **главна буква**, трябва да **извадите** **позицията на буквата в азбуката** от числото
* Ако е **малка буква**, трябва да **съберете** числото с **позицията на буквата в азбуката.**

Тази игра станала много лесна на Наков. Затова решил да я усложни като приемат повече низове и намира тяхната сума. Започнал да я решава, но не успял, защото станала много сложна. Затова решил да ви попита дали ще можете да напишете програма, **която събира сумите на числата след операциите**.

Например ще ви получите **"A12b s17G":**

Имаме два низа. Стартираме с първата буква пред числото на първия низ. **A** е **главна буква** и на **първо място** в азбуката. Затова разделяме числото 12 на 1 **(12/1 = 12).** След това преминаваме към буквата след числото. **b** е **малка буква** и е на **позиция 2** в азбуката. Затова добавяме 2. Получаваме **(12+2=14).** Подобно е за буквата **s**, която е **малка буква** и е на **позиция 19** в азбуката, затова умножаваме с числото **(17\*19 = 323).** След това е буквата **G**, която е главна и е на позиция 7 в азбуката. Изваждаме от резултата и получаваме **323 – 7 = 316.** Накрая събираме двете суми и получаваме 330 (**14 + 316=330**).

### Вход

Входът ще бъде **един ред**, който съдържа низове, разделени с **интервал**.

### Изход

На конзолата трябва да отпечатате на нов ред: **сумата на всички операции**, закръглена до **втората цифра** след запетаята

### Бележки

* **Броят** на низовете ще бъде в обхвата [1 … 10].
* Числата между буквите ще бъдат в обхвата [1 … 2 147 483 647].

### Примери

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| A12b s17G | 330.00 | 12/1=12, 12+2=14, 17\*19=323, 323–7=316, **14+316=330** |
| P34562Z q2576f H456z | 46015.13 |  |
| a1A | 0.00 |  |