# Упражнения: Низ и текстови операции

Тествайте решението в Judge: <https://judge.softuni.bg/Contests/3176/String-Processing>.

## Валидни потребителски имена

Напишете програма, която чете **имена на потребители**, разделени със **запетая и интервал**. Отпечатайте **валидните** потребителски имена.

Едно потребителско име е **валидно**, когато:

* **Дължината** му е между **3 и 16 символа**
* **Съдържа** само бувки, числа, тирета и долни черти

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| sh, too\_long\_username, !lleg@l ch@rs, jeffbutt | jeffbutt |
| Jeff, john45, ab, cd, peter-ivanov, @smith | Jeff  John45  peter-ivanov |

## Повтаряне на низове

Напишете програма, която чете **масив от низове**. Всеки низ трябва да бъде повторен **N пъти**, където **N** е **дължината** на низа. Накрая ги отпечатайте.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| hi abc add | hihiabcabcabcaddaddadd |
| work | workworkworkwork |
| ball | ballballballball |

### Насоки

1. Четем **дължината на низа**

s

1. Инициализиране на **StringBuilder**.



1. Минаваме през **всеки елемент от масива­**

Logo

Description automatically generated with medium confidence

1. Намираме **дължината на думата**

Text

Description automatically generated

1. Отпечатваме **StringBuilder**.

## Подниз

На **първия ред** ще получите **низ**. На **следващия ред** ще получите още един **низ**. Напишете програма, която **изтрива всички** съвпадения от първия във втория низ. Накрая отпечатайте обработения низ.

### Примери

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| Ice  kicegiceiceb | kgb | Първо премахваме ice един път и получаваме "kgiciceeb".  После премахваме ice още веднъж и получаваме "kgiceb".  Накрая премахваме отново и получаваме "kgb" |

### Насоки

1. Четем входа
2. Намираме **първия индекс**, където се появява първия низ
   * Използваме метода **IndexOf()**
3. **Премахваме** съвпадението.
   * Използваме метода **Remove()**
4. **Повтаряме** тези стъпки, докато **не можем да открием** повече съвпадения

## Текст филтър

Напишете програма, която получава **текст** и **забранени думи**. Всички забранени думи трябва да бъдат заминени с "\*" . Забранените думи ще бъдат разделени със **запетая и интервал**.

Всички забранени думи ще бъдат в текста.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| Linux, Windows  It is not **Linux**, it is GNU/**Linux**. **Linux** is merely the kernel, while GNU adds the functionality. Therefore we owe it to them by calling the OS GNU/**Linux**! Sincerely, a **Windows** client | It is not \*\*\*\*\*, it is GNU/\*\*\*\*\*. \*\*\*\*\* is merely the kernel, while GNU adds the functionality. Therefore we owe it to them by calling the OS GNU/\*\*\*\*\*! Sincerely, a \*\*\*\*\*\*\* client |

### Насоки

1. Четем входа
2. Заменяме всички забранени думи с "\*"
   * Използваме метода **Replace()**
   * Използваме **new string(**char ch, int repeatCount**)** за да заменяме

## Файл

Напишете програма, която чете **път до файл**. Вашата задача е да извадите **името на файла** и неговото **разширение**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| C:\Internal\training-internal\Template.pptx | File name: Template  File extension: pptx |
| C:\Projects\Data-Structures\LinkedList.cs | File name: LinkedList  File extension: cs |
| C:\Projects\Data-Structures\README | File name: README  File extension: |
| Hello.txt | File name: Hello  File extension: txt |

## Шифър на Цезар

Напишете програма, която създава **криптиран низ**. Крипитирането работи като **всеки символ** се измества с **три позиции напред**. Например ако имаме **A**, то тя трябва да бъде заменена с **D**, **B** трябва да бъде заменена с **E** и т.н. Накрая отпечатайте **криптирания** **низ**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| Programming is cool! | Surjudpplqj#lv#frro$ |
| One year has 365 days. | Rqh#|hdu#kdv#698#gd|v1 |

## 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 |

## 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 |  |