**Задание**:

Создайте новый репозиторий tasks-strings**-**sr

Выполните задачи и залейте на данный про репозиторий филь через gitbash.

**Задача 1.**

Напишите программу, которая выводит обозначения клеток псевдо-шахматной доски. Клетки нумеруются (заглавными) латинскими буквами слева направо и натуральными числами снизу вверх, после каждого обозначения клетки следует пробел. Доска квадратная, размер вводится с клавиатуры и не превышает 9.

**Формат ввода**

Натуральное число, не превышающее 9 — размер доски.

**Формат вывода**

Псевдо-шахматная доска в формате, описанном в условии и показанном в примере.

a = int(input())  
**if** a == 1:  
 print (**"A1"**)  
**elif** a == 2:  
 print(**"A1 B1"**)  
 print(**"A2 B2"**)  
**elif** a == 3:  
 print(**"A1 B1 C1"**)  
 print(**"A2 B2 C2"**)  
 print(**"A3 B3 C3"**)  
**elif** a == 4:  
 print(**"A1 B1 C1 D1"**)  
 print(**"A2 B2 C2 D2"**)  
 print(**"A3 B3 C3 D3"**)  
 print(**"A4 B4 C4 D4"**)  
**elif** a == 5:  
 print(**"A1 B1 C1 D1 E1"**)  
 print(**"A2 B2 C2 D2 E2"**)  
 print(**"A3 B3 C3 D3 E3"**)  
 print(**"A4 B4 C4 D4 E4"**)  
 print(**"A5 B5 C5 D5 E5"**)  
**elif** a == 6:  
 print(**"A1 B1 C1 D1 E1 F1"**)  
 print(**"A2 B2 C2 D2 E2 F2"**)  
 print(**"A3 B3 C3 D3 E3 F3"**)  
 print(**"A4 B4 C4 D4 E4 F4"**)  
 print(**"A5 B5 C5 D5 E5 F5"**)  
 print(**"A6 B6 C6 D6 E6 F6"**)  
**elif** a ==7:  
 print(**"A1 B1 C1 D1 E1 F1 G1"**)  
 print(**"A2 B2 C2 D2 E2 F2 G2"**)  
 print(**"A3 B3 C3 D3 E3 F3 G3"**)  
 print(**"A4 B4 C4 D4 E4 F4 G4"**)  
 print(**"A5 B5 C5 D5 E5 F5 G5"**)  
 print(**"A6 B6 C6 D6 E6 F6 G6"**)  
 print(**"A7 B7 C7 D7 E7 F7 G7"**)  
**elif** a == 8:  
 print(**"A1 B1 C1 D1 E1 F1 G1 H1"**)  
 print(**"A2 B2 C2 D2 E2 F2 G2 H2"**)  
 print(**"A3 B3 C3 D3 E3 F3 G3 H3"**)  
 print(**"A4 B4 C4 D4 E4 F4 G4 H4"**)  
 print(**"A5 B5 C5 D5 E5 F5 G5 H5"**)  
 print(**"A6 B6 C6 D6 E6 F6 G6 H6"**)  
 print(**"A7 B7 C7 D7 E7 F7 G7 H7"**)  
 print(**"A8 B8 C8 D8 E8 F8 G8 H8"**)  
**elif** a== 9:  
 print(**"A1 B1 C1 D1 E1 F1 G1 H1 J1"**)  
 print(**"A2 B2 C2 D2 E2 F2 G2 H2 J2"**)  
 print(**"A3 B3 C3 D3 E3 F3 G3 H3 J3"**)  
 print(**"A4 B4 C4 D4 E4 F4 G4 H4 J4"**)  
 print(**"A5 B5 C5 D5 E5 F5 G5 H5 J5"**)  
 print(**"A6 B6 C6 D6 E6 F6 G6 H6 J6"**)  
 print(**"A7 B7 C7 D7 E7 F7 G7 H7 J7"**)  
 print(**"A8 B8 C8 D8 E8 F8 G8 H8 J8"**)  
 print(**"A9 B9 C9 D9 E9 F9 G9 H9 J9"**)

**Пример**

| Ввод | Вывод |
| --- | --- |
| 4 | A4 B4 C4 D4  A3 B3 C3 D3  A2 B2 C2 D2  A1 B1 C1 D1 |

**Задача 2.**

Во многих интернет-сервисах при регистрации нужно указывать желаемое имя пользователя, причём в имени разрешается использовать только латинские буквы, цифры и символ «\_». Напишите программу, которая проверяет, подходит ли строка для того, чтобы быть именем пользователя в таком сервисе.

**Формат ввода**

Вводится одна строка — предполагаемое имя пользователя.

**Формат вывода**

Выводится одна строка: если введённая строка состоит только из маленьких латинских букв, цифр и символа «\_», то выводится «OK» (латиницей); иначе выводится: «Неверный символ: », после чего следует первый не подходящий символ из введённой строки.

stroka = input()  
**for** i **in** stroka:  
 **if not**(i.islower() **or** i.isdigit() **or** i==**'\_'**):  
 print(**'Неверный символ:'**, i)  
 **break  
else**:  
 print(**'OK'**)

**Пример**

| Ввод | Вывод |
| --- | --- |
| m&m's | Неверный символ: & |

**Задача 3.**

Напишите программу, обрабатывающую один раунд игры «Быки и коровы». Пользователь вводит две строки. Гарантируется, что это две строки одинаковой длины и что все символы в каждой из них разные. Необходимо вывести отдельно количество быков — символов, которые есть в обеих строках и стоят на одном и том же месте, и количество коров — символов, которые есть в обеих строках, но на разных местах.

**Формат ввода**

Две строки.

s1 = input()  
s2 = input()  
bulls = 0  
cows = 0  
**for** i **in** set(s1) & set(s2):  
 **if** len(s1) == len(s2):  
 **if** s1.index(i) == s2.index(i):  
 bulls +=1  
 **else**: cows += 1  
 **else**:  
 print(**"неравное количество символов"**)  
print(**'bulls:'**,bulls,**'cows:'**,cows)

**Формат вывода**

Два целых числа через пробел — количество быков и коров.

**Пример**

| Ввод | Вывод |
| --- | --- |
| питон  пилот | 3 1 |

# Задача 4

Напишите программу, которая считывает слова, слово «стоп» — сигнал остановки (оно не должно принимать участие в анализе). Из введённых слов нужно выбрать самое длинное и самое короткое (гарантируется, что все они имеют разную длину) и проверить, есть ли все буквы короткого слова в длинном, при этом количество повторений букв не учитывается. Вывести «ДА» или «НЕТ» в зависимости от этого.

## Формат ввода

Слова разной длины, из которых нужно выбрать кратчайшее и длиннейшее.  
После этих слов следует слово «стоп».

slovaк = input()  
min = slovaк  
max = slovaк  
y = **'ДА'**n = **'НЕТ'  
while True**:  
 slovaк = input()  
 **if** slovaк == **'стоп'**:  
 **if** len(set(min) - set(max)) == 0:  
 print(y)  
 **break  
 else** :  
 print(n)  
 **break  
  
 if** len(slovaк) > len(max):  
 max = slovaк  
 **if** len(slovaк) < len(min):  
 min = slovaк

## Формат вывода

ДА или НЕТ

### Пример 1

| Ввод | Вывод |
| --- | --- |
| грот  вигвам  гаррота  стоп | ДА |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| чаепитие  чашка  печенье  стоп | НЕТ |

### Пример 3

| Ввод | Вывод |
| --- | --- |
| чечевица  светопреставление  престол  стоп | ДА |

**Задача 5**

Напишите программу, которая коварно превращает некоторые полезные советы в вредные.

**Формат ввода**

Сначала вводится натуральное число N — количество полезных советов.  
На следующих N строках вводятся полезные советы. Гарантируется, что длина каждого — не менее 3 символов.

**Формат вывода**

Нужно вывести те же советы в том же порядке, однако у тех из них, что начинаются со слова «не» (которое также может быть записано как «Не») и пробела, выводить это начальное «не» и пробел не нужно.

**for** яzov **in** range(int(input())):  
 lst = input().split()  
 **if** lst[0].lower() == **'не'**:  
 print(**' '**.join(lst[1:]))  
 **else**:  
 print(**' '**.join(lst))

**Пример**

| Ввод | Вывод |
| --- | --- |
| 3  Мой руки перед едой!  Не суй пальцы в розетку!  Берегись автомобиля! | Мой руки перед едой!  суй пальцы в розетку!  Берегись автомобиля! |

**Задача 6**

Напишите программу, которая укорачивает заголовки новостей, чтобы их анонсы поместились в ленте событий на сайте новостного агентства. Если длина заголовка превышает максимальную допустимую длину анонса, следует укоротить его и добавить в конце многоточие (в виде трёх точек), чтобы получившийся анонс имел ровно максимальную допустимую длину.

**Формат ввода**

На первой строке вводится натуральное число — максимальная допустимая длина анонса.  
На второй строке вводится натуральное число N — количество заголовков.  
Далее вводится N заголовков, каждый на отдельной строке. Гарантируется, что заголовки не короче 4 символов.

**Формат вывода**

Для каждого заголовка выводится соответствующий анонс: если длина заголовка не превышает максимальную, то он и есть анонс, иначе его следует укоротить согласно условию.

max = int(input())  
n = int(input())  
  
**for** i **in** range(n):  
 title = input()  
 **if** len(title) <= max:  
 print(title)  
 **else**:  
 shortened\_title = title[:max-3] + **"..."** print(shortened\_title)

**Пример**

| Ввод | Вывод |
| --- | --- |
| 25  3  Начался саммит по глобальному потеплению  Завтра Новый год!  Python и Java конкурируют за звание самого популярного языка программирования | Начался саммит по глоб...  Завтра Новый год!  Python и Java конкурир... |

**Примечания**

Следует учесть, что многоточие входит в максимальную длину.

**Задача 6**

Найдите, в каких строках из введённых и в каком месте упоминается кот.

**Формат ввода**

На первой строке вводится натуральное число N — количество строк.  
Далее следуют N строк.

**Формат вывода**

Для каждой строки, в которой есть сочетание символов «кот», нужно вывести (в порядке появления таких строк) номер этой строки (нумерация начинается с единицы) и номер символа, с которого начинается первое вхождение этой подстроки (нумерация символов также с единицы).

nekot = int(input())  
baza = []  
**for** \_ **in** range(nekot):  
 baza.append(input())  
  
**for** i, line **in** enumerate(baza):  
 index = line.find(**"кот"**)  
 **if** index != -1:  
 print(i+1, index+1)

**Пример**

| Ввод | Вывод |
| --- | --- |
| 3  локоть  молоко  пёс и кот | 1 3  3 7 |