Lab: Lists Basics

Please, submit your source code solutions for the described problems to the Judge System.

1. Strange Zoo

You are at the zoo, and the meerkats look strange.

You will receive **3 strings** on separate lines, representing the **tail**, the **body**, and the **head** of an animal in that order. Your task is to **re-arrange** the elements **in a list** so that the animal looks normal again:

- On the **first position** is the **head**;
- On the **second position** is the **body**;
- On the **last one** is the **tail**.

Example

Input	Output
my tail my body seems on place my head is on the wrong end!	['my head is on the wrong end!', 'my body seems on place', 'my tail']
tail body head	['head', 'body', 'tail']
T B H	['H', 'B', 'T']

Hints

We start by reading the three parts of the body:

```
1    tail = input()
2    body = input()
3    head = input()
```

Then, we create a list containing those three elements:

```
4    meerkat = [tail, body, head]
```

We swap the elements and print the list:

```
5    meerkat[0], meerkat[2] = meerkat[2], meerkat[0]
6    print(meerkat)
```

2. Courses

On the first line, you will receive a single number **n**. On the following **n** lines, you will receive **names** of courses. You should create a **list of courses and print it**.

Example

Input	Output

Follow us:

2 PB Python PF Python	['PB Python', 'PF Python']
4 Front-End C# Web JS Core Programming Fundamentals	['Front-End', 'C# Web', 'JS Core', 'Programming Fundamentals']

Hints

We read the number **n** from the console, and we create an **empty list**:

```python
n = int(input())
courses = []
```

Then, we create a loop that reads each course and adds it to the list:

```python
for i in range(n):
    current_course = input()
    courses.append(current_course)
```

Finally, we print the list:

```python
print(courses)
```

3. List Statistics

On the first line, you will receive a number **n**. On the following **n** lines, you will receive integers. You should **create** and **print** two lists:

- One with all the positive (including 0) numbers
- One with all the negative numbers

Finally, print the following message:

"Count of positives: {count_positives}

Sum of negatives: {sum_of_negatives}"

Example

Input	Output
5 10 3 2 -15 -4	[10, 3, 2] [-15, -4] Count of positives: 3 Sum of negatives: -19
6 11 2 35 599	[11, 2, 35, 599, 31, 20] [] Count of positives: 6 Sum of negatives: 0

31 20	

Hints

We start by reading the number n:

```python
n = int(input())
positives = []
negatives = []
```

Then, we create **a loop** that reads the **current number** and checks if it **is positive or not**:

```python
for n in range(n):
    current_number = int(input())
    if current_number >= 0:
        positives.append(current_number)
    else:
        negatives.append(current_number)
```

- If it is, we add it to the list of positive numbers.
- If it is not, we add it to the list of negative numbers.

Then we print the three lines:

```python
print(positives)
print(negatives)
print(f"Count of positives: {len(positives)}. Sum of negatives: {sum(negatives)}")
```

- To get the count of the positives, we can use the **len** function.
- To get the sum of the negatives, we can use the **sum** function.

4. Search

On the first line, you will receive a number **n**. On the second line, you will receive a word. On the following **n lines,** you will be given some **strings**. You should **add** them to a **list and print** them. After that, you should **filter out** only the strings that **include** the given **word** and **print** that list too.

Example

Input	Output
3 SoftUni I study at SoftUni I walk to work I learn Python at SoftUni	["I study at SoftUni", "I walk to work", "I learn Python at SoftUni"] ["I study at SoftUni", "I learn Python at SoftUni"]
4 tomatoes I love tomatoes I can eat tomatoes forever I don't like apples Yesterday I ate two tomatoes	["I love tomatoes", "I can eat tomatoes forever", "I don't like apples", "Yesterday I ate two tomatoes"] ["I love tomatoes", "I can eat tomatoes forever", "Yesterday I ate two tomatoes"]

Hints

We start by reading the number n and the word we would search for. Then, we create our empty list:

```python
n = int(input())
word = input()
strings = []
```

We create a loop that adds all the strings to our list. After that, we print it:

```python
for i in range(n):
    current_string = input()
    strings.append(current_string)
print(strings)
```

Finally, we create another loop to remove the strings we do not need by iterating through the strings reversed (so we don't skip elements when removing) and print the list again:

```python
for i in range(len(strings) - 1, -1, -1):
    element = strings[i]
    if word not in element:
        strings.remove(element)
print(strings)
```

5. Numbers Filter

On the first line, you will receive a single number **n**. On the following **n** lines, you will receive integers. After that, you will be given one of the following commands:

- even
- odd
- negative
- positive

Filter all the numbers that fit in the category (0 counts as a positive and even). Finally, print the result.

Example

Input	Output
5 33 19 -2 18 998 even	[-2, 18, 998]
3 111 -4 0 negative	[-4]

Follow us:

Hints

First, we read the number n. Then, we create the numbers list and the filtered list:

```python
n = int(input())
numbers = []
filtered = []
```

We create a loop that reads all the numbers and adds them to the list:

```python
for i in range(n):
    current_number = int(input())
    numbers.append(current_number)
```

Then, we read the command and check for all the cases:

```python
if command == "even":
    for number in numbers:
        if number % 2 == 0:
            filtered.append(number)
elif command == "odd":
    for number in numbers:
        if number % 2 != 0:
            filtered.append(number)
elif command == "negative":
    for number in numbers:
        if number < 0:
            filtered.append(number)
elif command == "positive":
    for number in numbers:
        if number >= 0:
            filtered.append(number)
print(filtered)
```

Finally, we print the filtered list.