

Домашнее задание 3

Курс: Программирование [Advanced]
Мягкий deadline: 20 октября 2018 23:59
Жесткий deadline: 3 ноября 2018 23:59

Правила оформления кода

➡ <http://stanford.edu/class/archive/cs/cs106b/cs106b.1158/styleguide.shtml>

Дополнительный материал

➡ Что каждый программист должен знать о памяти
<http://rus-linux.net/lib.php?name=/MyLDP/hard/memory/memory.html>

Оценка

8 - 10 —> 3 балла к рейтингу
11 - 12 —> 4 балла к рейтингу
13 - 15 —> 5 балла к рейтингу

Кодекс чести (Honor code)

- ➡ В заголовке файла с решением я укажу (в комментариях) все источники помощи, включая книги, веб-страницы, друзей, преподавателей и т.д;
- ➡ Не буду использовать код других студентов для сдачи решений;
- ➡ Не буду пытаться маскировать код других студентов в своих решениях;
- ➡ Буду соблюдать правила оформления кода;
- ➡ Я согласен со всеми пунктами Кодекса Чести и обязуюсь следовать им;

[1][1 балл]. Дан массив чисел. Выведите все элементы массива, которые больше предыдущего элемента .

Формат входных данных:

В первой строке вводится количество элементов в массиве. Во второй строке вводятся элементы массива.

Формат выходных данных:

Выведите ответ на задачу.

Sample Input:

5
1 5 2 4 3

Sample Output:

5 4

[2][1 балл]. Дан массив целых чисел. Если в нем есть два соседних элемента одного знака, выведите эти числа. Если соседних элементов одного знака нет - не выводите ничего. Если таких пар соседей несколько - выведите первую пару.

Формат входных данных:

В первой строке вводится количество элементов в массиве. Во второй строке вводятся элементы массива. Все числа отличны от нуля.

Формат выходных данных:

Выведите ответ на задачу. Элементы выводятся в порядке не убывания.

Sample Input:

5
-1 2 3 -1 2

Sample Output:

2 3

[3][1 балл]. Выведите значение наименьшего из всех положительных элементов в массиве. Известно, что в массиве есть хотя бы один положительный элемент.

Sample Input:

5
5 -4 3 -2 1

Sample Output:

1

[4][1 балл]. Дан список, упорядоченный по неубыванию элементов в нем. Определите, сколько в нем различных элементов.

Формат входных данных:

В первой строке вводится количество элементов в массиве. Во второй строке вводятся элементы массива.

Формат выходных данных:

Выведите ответ на задачу.

Sample Input:

6
1 2 2 3 3 3

Sample Output:

3

[5][1 балл]. Переставьте соседние элементы массива ($A[0]$ с $A[1]$, $A[2]$ с $A[3]$ и т.д.). Если элементов нечетное число, то последний элемент остается на своем месте.

Sample Input:

5
1 2 3 4 5

Sample Output:

2 1 4 3 5

[6][1 балл]. Циклически сдвиньте элементы списка вправо ($A[0]$ переходит на место $A[1]$, $A[1]$ на место $A[2]$, ..., последний элемент переходит на место $A[0]$).

Sample Input:

5
1 2 3 4 5

Sample Output:

5 1 2 3 4

[7][1 балл]. Усовершенствуйте предыдущую программу для циклического сдвига элементов на N позиций, где N целое число. Если $N > 0$, то сдвиг производится вправо, если $N < 0$, то влево.

Формат входных данных:

В первой строке вводится количество элементов в массиве. Во второй строке вводится N . В третьей строке вводятся элементы массива.

Формат выходных данных:

Выведите ответ на задачу.

Sample Input 1:

```
5
3
1 2 3 4 5
```

Sample Output 1:

```
3 4 5 1 2
```

Sample Input 2:

```
5
-8
1 2 3 4 5
```

Sample Output 2:

```
4 5 1 2 3
```

[8][1 балл]. Дан массив. Выведите те его элементы, которые встречаются в массиве только один раз. Элементы нужно выводить в том порядке, в котором они встречаются в списке.

Sample Input 1:

```
6
1 2 2 3 3 3
```

Sample Output 1:

```
1
```

Sample Input 2:

```
8
4 3 5 2 5 1 3 5
```

Sample Output 2:

```
4 2 1
```

[9][1 балл]. N кеглей выставили в один ряд, занумеровав их слева направо числами от 1 до N. Затем по этому ряду бросили K шаров, при этом i-й шар сбил все кегли с номерами от l_i до r_i включительно. Определите, какие кегли остались стоять на месте.

Формат входных данных:

Программа получает на вход количество кеглей N и количество бросков K. Далее идет K пар чисел l_i, r_i , при этом $1 \leq l_i, r_i \leq N$.

Формат выходных данных:

Программа должна вывести последовательность из N символов, где j-й символ есть "I", если j-я кегля осталась стоять, или ".", если j-я кегля была сбита.

Sample Input 1:

```
10 3
8 10
2 5
3 6
```

Sample Output 1:

```
I.....I....
```

[10][1 балл]. Дан массив положительных целых чисел. Программа должна вывести массив целых чисел представляющий следующее положительное число которое задает исходный массив. Цифры расположены в массиве так, что первая значащая цифра числа находится на нулевой позиции массива. Каждый элемент массива содержит единственную цифру. Предполагается что массив не может начинаться с нуля.

Формат входных данных:

В первой строке вводится количество элементов в массиве. Во второй строке вводятся элементы массива.

Sample Input 1:

```
3
1 2 3
```

Sample Output 1:

```
1 2 4
```

Объяснение: заданный массив представляет целое число 123. Следующее число будет 124.

Sample Input 2:

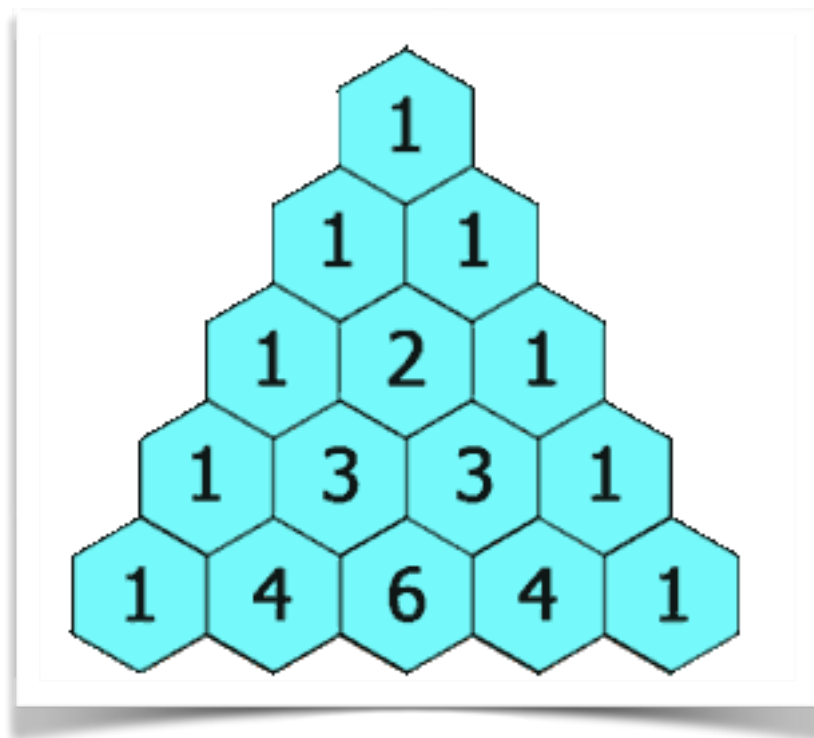
```
4
1 2 3 9
```

Sample Output 3:

```
1 2 4 0
```

Объяснение: заданный массив представляет целое число 1239. Следующее число будет 1240.

[11][1 балл]. Вводится положительное целое число numRow, задающее число рядов треугольника Паскаля. Программа должна вывести на экран треугольник Паскаля размера numRow.



Треугольник Паскаля - такой треугольник в котором каждое число равно сумме двух расположенных над ним чисел. На вершине и по бокам стоят единицы. Строки треугольника симметричны относительно вертикальной оси.

Формат входных данных:

Программа получает на вход целое число numRow.

Формат выходных данных:

Программа должна вывести треугольник Паскаля.

Sample Input:

5

Sample Output:

```
[1]
[1,1]
[1,2,1]
[1,3,3,1]
[1,4,6,4,1]
```

[12][1 балл]. Вводится положительное целое число $K < 34$. Программа должна вернуть K -ый ряд треугольника Паскаля. Индексация рядов треугольника начинается с нуля. Определение и пример треугольника Паскаля см. в задании 11.

Sample Input 1:

3

Sample Output 1:

1 3 3 1

[13][1 балл]. Вводится строка S и символ C . Программа должна вернуть массив значений представляющий кратчайшие расстояния от символа C в строке S .

Формат входных данных:

В первой строке вводится строка S длиной от 1 до 10000. Во второй строке вводится единственный символ C . Гарантируется что C содержится в S . Все символы в C и S заданы в нижнем регистре.

Формат выходных данных:

Выведите ответ на задачу.

Sample Input 1:

loveprogramming

o

Sample Output 1:

[1, 0, 1, 2, 2, 1, 0, 1, 2, 3, 4, 5, 6, 7, 8]

[14][2 балла]. Даны действительные коэффициенты a , b , c , при этом $a \neq 0$. Решите квадратное уравнение $ax^2 + bx + c = 0$ и выведите все его корни.

Формат входных данных:

Вводятся три действительных числа.

Формат выходных данных:

Если уравнение имеет два корня, выведите два корня в порядке возрастания, если один корень — выведите одно число, если нет корней — не выводите ничего.

Sample Input 1:

1

-1

-2

Sample Output 1:

-1 2