

Добрый день, уважаемый кандидат.

Комментарии к выполнению заданий:

- Обязательно снабдить код комментарием (на английском языке)
- В первую очередь интересуют исходники
- Задания могут быть сделаны на любом языке программирования, если есть возможность на Java - лучше на Java.
- Обязательны выполнения всех трех задач

Вопрос: "если Вы желаете набрать людей для обучения, зачем задавать задания уровня олимпиады по программированию?"

Ответ: "Нам нужно фильтровать людей, которые с одной стороны ничего не умеют еще в наших технологиях, а с другой, которые смогут решать сложные задания и у которых работает голова.

Если человек сможет разобраться и решить подобные задачи, то и сможет быстро обучиться. Если бросит все на пол пути, то скорей всего и с него ничего толкового не выйдет."

Удачи!

1. Если мы из корректно записанного арифметического выражения, содержащего числа, знаки операций и открывающие и закрывающие круглые скобки выбросим числа и знаки операций, а затем запишем оставшиеся в выражении скобки без пробелов между ними, то полученный результат назовем правильным скобочным выражением [скобочное выражение "(()())" - правильное, а "())" и "())(" - нет]. Найти число правильных скобочных выражений, содержащих N открывающихся и N закрывающихся скобок. N вводится с клавиатуры. N неотрицательное целое число.

Пример:

N = 1 (по одной скобке открывающейся и закрывающейся) - ответ 1

()
)(
)
((

Только один правильный вариант

Для введенного числа 2 - 2 :

((
(())

То есть только два варианта, когда все открытые скобки правильно открываются/закрываются.

И так далее.

2. You are given a list of cities. Each direct connection between two cities has its transportation cost (an integer bigger than 0). The goal is to find the paths of minimum cost between pairs of cities. Assume that the cost of each path (which is the sum of costs of all direct connections belonging to this path) is at most 200000. The name of a city is a string containing characters a,...,z and is at most 10 characters long.2)

Input

s [the number of tests ≤ 10]
n [the number of cities ≤ 10000]
NAME [city name]
p [the number of neighbors of city *NAME*]
nr cost [*nr* - index of a city connected to *NAME* (the index of the first city is 1)]
 [*cost* - the transportation cost]
r [the number of paths to find ≤ 100]
NAME1 NAME2 [*NAME1* - source, *NAME2* - destination]
[empty line separating the tests]

Output

cost [the minimum transportation cost from city *NAME1* to city *NAME2* (one per line)]

Example

Input:

```
1
4
gdansk
2
2 1
3 3
bydgoszcz
3
1 1
3 1
4 4
torun
3
1 3
2 1
4 1
warszawa
2
2 4
3 1
2
```

gdansk warszawa
bydgoszcz warszawa

Output:

3

2

3. Find the sum of the digits in the number $100!$ (i.e. 100 factorial)
{Correct answer: 648}