**Задача A. Распаковка строки (10)**

Будем рассматривать строки, состоящие из заглавных латинских букв. Например: AAAABCCCCCDDDD. Длина этой строки равна 14. Повторяющиеся символы могут быть удалены и заменены числами, определяющими количество повторений, и самим символом. Данная строка может быть представлена как 4AB5C4D. Описанный метод мы назовем упаковкой строки.

Напишите программу, которая берет упакованную строчку и восстанавливает по ней исходную строку.

**Входные данные.** Входные данные состоят из одной упакованной строки. Максимальная длина строки не превышает 80.

**Выходные данные.** Восстановленная строка. При этом строка должна быть разбита на строчки длиной ровно по 40 символов (за исключением последней, которая может содержать меньше 40 символов).

**Примеры**

|  |  |  |
| --- | --- | --- |
| **№** | **Пример ввода** | **Пример вывода** |
| 1 | 3A4B7D | AAABBBBDDDDDDD |
| 2 | 95AB | AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAB |

**Задача B. Единичный НОД (10)**

Заданы два натуральных числа в десятичной системе счисления, состоящие из единиц. В первом числе ровно N единиц, а во втором их ровно M. Требуется найти НОД этих чисел.

Напомним, что НОД (наибольший общий делитель) двух чисел a и b — это такое максимальное число c, что b делится на c и a делится на c.

**Входные данные. Д**ва целых числа N и M (1 ≤ N, M ≤ 2000).

**Выходные данные.** Одно число **-** НОД чисел N и M.

**Примеры**

|  |  |  |
| --- | --- | --- |
| **№** | **Пример ввода** | **Пример вывода** |
| **1** | **1 1** | **1** |
| **2** | **1 2** | **1** |

**Задача C. Сумма (15)**

Задано натуральное число x. Найдите число способов представить его в виде суммы четырех натуральных чисел: x = a + b + c + d, где a <= b <= c <= d.

**Входные данные.** Одно целое число x (1 <= x <= 1500).

**Выходные данные.** Одно число – полученное количество способов искомого представления..

**Примеры**

|  |  |  |
| --- | --- | --- |
| ***№*** | ***Пример ввода*** | ***Пример вывода*** |
| **1** | 3 | 0 |
| **2** | 5 | 1 |

**Задача D. Нить Ариадны (35)**

Тезею из лабиринта Минотавра помог выйти клубок ниток. Требуется написать программу, которая вводит маршрут Тезея в лабиринте и находит кратчайший обратный путь, по которому Тезей сможет выйти из лабиринта, не заходя в тупики и не делая петель.

**Входные данные.** Входным данным является маршрут Тезея, который представлен строкой, состоящей из букв: N, S, W, E и длиной не более 200.

Буквы означают:

N - один "шаг" на север, S - один "шаг" на юг, W - один "шаг" на запад, E - один "шаг" на восток.

**Выходные данные.** Аналогично входному данному, найденный обратный путь. Если маршрут неоднозначен, то следует выбирать согласно следующему приоритету: N, E, S, W.

**Пример**

|  |  |  |
| --- | --- | --- |
| ***№*** | ***Пример ввода*** | ***Пример вывода*** |
| ***1*** | EENNESWSSWE | NWW |

**Задача E. Прыжки по буквам (30)**

Дана цепочка из N символов, состоящая из прописных латинских букв. Необходимо пройти с первого символа цепочки до последнего символа, прыгая не более чем на K символов. Стоимость прыжка, при котором символ не меняется, равна 0, а стоимость прыжка на другой символ равна 1. Требуется написать программу, которая вычислит наименьшую стоимость перехода с первого на последний символ.

**Входные данные. В** первой строке входных данных содержится два целых числа: длина цепочки N (2 ≤ N ≤ 105) и максимальная длина прыжка K (1 ≤ K < N). Во второй строке содержится цепочка из N латинских букв.

**Выходные данные.** Одно число - минимальную стоимость перехода.

**Пример**

|  |  |  |
| --- | --- | --- |
| ***№*** | ***Пример ввода*** | ***Пример вывода*** |
| ***1*** | 10 2 ABABBCACBC | 2 |