

Олимпиада проводится четыре часа, даны задания на письменное решение и программное, каждая задача имеет разный уровень сложности и оценивается по баллам.

Задачи для решения письменно.

Писать аккуратно. На листочках указать фамилию имя отчество, кафедру и группу. К доказательствам привести примеры.

Задача 1 (5 баллов).

В системе счисления (СС) с основанием $N > 2$ число делится на $1(N-1)$ – двузначное число, у которого первая цифра равна 1, а вторая – $N-1$, если сумма этого числа с отброшенной последней цифрой и удвоенного значения последней цифры также делится на $1(N-1)$. Например, в десятичной СС число 627_{10} делится на 19_{10} , т.к. $62 + 7 \cdot 2 = 76_{10}$, а 76_{10} делится на 19_{10} . Другой пример – в шестнадцатеричной СС число $41E_{16}$ делится на $1F_{16}$, т.к. $41 + E \cdot 2 = 5D_{16}$, а $5D_{16} = 93_{10}$ делится на $1F_{16} = 31_{10}$. Докажите это утверждение.

Задача 2 (3 балла).

Доказать, что целое число X с основанием системы счисления N и разрядностью k , подвергнутое циклическому сдвигу влево, можно получить умножением на N и получением остатка от деления на N в степени $k - 1$.

Например, для числа 1923_{10} циклический сдвиг влево соответствует 9231_{10} , при $N = 10$, это аналогично операции $1923 \cdot 10 \bmod 9999$.

Задача 3 (2 балла).

Дана сумма 2026 любых натуральных чисел - X , X при этом оказалось нечётным числом. Каким числом - чётным или нечётным - является произведение этих чисел?

Привести доказательство.

Задача 4 (3 балла).

Найти остаток от деления 7^n на 8 (n – любое натуральное число). Привести вывод.

Задача 5 (2 балла).

Написать алгоритм обмена переменных x , y , z используя только эти переменные

В x должно оказаться y .

В y должно оказаться z .

В z должно оказаться x .

Не использовать языковые конструкции с кортежами, например, в Python $x, y = y, x$ или в C# $(x, y) = (y, x)$ и т.п.

Задачи для решения программно.

Задача 1 (5 баллов).

Для каждого числа N из множества натуральных чисел существует подмножество D_N , называемое подмножеством цифровых перестановок данного числа. Элементы такого подмножества представляют собой натуральные числа, образованные в результате всех возможных перестановок цифр, из которых состоит число N . Очевидно, что число элементов такого подмножества составляет $P(D_N) = P_N = N!$, если все цифры числа N различны. Примеры подмножеств цифровых перестановок:

- $P_1 = [1]$;
- $P_{12} = [12, 21]$;
- $P_{123} = [123, 132, 213, 231, 312, 321]$;
- $P_{1000} = [1, 10, 100, 1000]$.

Назовем суперпростым число, все элементы подмножества цифровых перестановок которого являются простыми числами. Требуется написать программу, находящую все суперпростые числа, не превышающие заданного числа N .

Формат ввода/вывода. Входной текстовый файл с именем `input.txt` содержит единственное натуральное число N в диапазоне $[1, 1000000]$. Выходной файл `output.txt` должен содержать все суперпростые числа, не превышающие N , расположенные в порядке возрастания. Каждое число должно располагаться на отдельной строке.

Пример ввода:

15

Пример вывода:

2

3

5

7

11

13

Задача 2 (5 баллов).

Дана прямоугольная матрица A целых чисел размером N строк и M столбцов. Требуется построить такой маршрут, начинающийся в левом верхнем элементе матрицы $A_{1,1}$, и заканчивающийся в нижнем правом элементе матрицы A_{NM} , чтобы сумма всех посещенных элементов была минимальной. Находясь в каком-либо элементе $A_{i,j}$, можно перейти в соседний элемент матрицы, находящийся либо слева ($A_{i,j-1}$), либо справа ($A_{i,j+1}$), либо сверху ($A_{i-1,j}$), либо снизу ($A_{i+1,j}$). При этом нельзя выходить за пределы матрицы. Например:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

В данной матрице оптимальным будет маршрут вправо $\rightarrow$ вправо $\rightarrow$ вправо $\rightarrow$ вниз $\rightarrow$ вниз $\rightarrow$ вниз. Минимальная сумма элементов равна $1 + 2 + 3 + 4 + 8 + 12 + 16 = 46$.

Формат ввода/вывода. В первой строке входного файла `input.txt` указаны через пробел размеры матрицы (сначала количество строк N , затем количество столбцов M). Размеры матрицы N, M находятся в диапазоне $[1, 1000]$. Далее в N строках по M столбцов содержатся элементы матрицы, разделенные пробелами. Значения элементов матрицы также находятся в диапазоне от 1 до 1000. В выходной файл `output.txt` должно быть записано ровно одно целое число – минимально возможная сумма посещенных элементов.

Пример ввода

```
4 4
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

Пример вывода

```
46
```

Задача 3 (4 балла).

Дана последовательность целых чисел длины n . Требуется определить, существует ли в этой последовательности подпоследовательность, являющаяся арифметической прогрессией длины не менее k , при этом каждый элемент этой подпоследовательности должен быть уникальным.

Формат ввода/вывода. В первой строке входного файла `input.txt` содержатся два целых числа n и k . Во второй строке содержится n целых чисел — элементы последовательности. В выходной файл `output.txt` выведите "Да", если такая подпоследовательность существует, иначе — "Нет".

Ограничения и подсказки:

Элементы последовательности могут иметь значения в диапазоне от -10^9 до 10^9 .

N ограничено до 10^6 .

Постарайтесь решить задачу за $O(n \cdot \log n)$ или лучше.

Пример ввода

10 3

1 2 3 4 5 6 1 1 1 1

Пример вывода

Да.

Задача 4 (2 балла).

По городу на двух кольцевых маршрутах двигаются два автобуса, один проходит свой маршрут за X минут, другой за Y минут, определить через какое ближайшее время они встретятся на единственной для них общей остановке, с которой они выехали в одно и то же время. Написать эффективную программу для вычисления этого времени в минутах. X и Y ограничены от 1 до 10^9 .

Формат ввода/вывода. На вход подается файл `input.txt` с целыми числами X и Y . На выходе файл `output.txt` с числом в минутах.

Пример ввода

30 45

Пример вывода

90

Задача 5 (3 балла).

Вывести все пары простых чисел для четного N , из которых можно составить сумму дающую число N . Формат ввода/вывода. На вход в подается файл input.txt с числом N . На выходе файл output.txt с количеством пар и сами пары на отдельных строках. N ограничено до 10^6 .

Пример ввода

100

Пример вывода

6

3 97

11 89

17 83

29 71

41 59

47 53