

Задача 5. Разложение на простые

Источник:	основная I
Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	3 секунды
Ограничение по памяти:	разумное

Внимание: эта задача проверяется на **emailtester**.

Дано несколько натуральных чисел, нужно разложить каждое число X на простые множители. Искомое разложение имеет вид:

$$X = p_1^{t_1} * p_2^{t_2} * \dots * p_k^{t_k}$$

При этом все числа p_i должны быть простыми и упорядоченными строго по возрастанию, а числа t_i должны быть целыми положительными.

Требуется реализовать следующую функцию для поиска разложения:

```
//структура, в которой хранится разложение на простые множители
typedef struct Factors {
    int k;                //сколько различных простых в разложении
    int primes[32];       //различные простые в порядке возрастания
    int powers[32];       //в какие степени надо эти простые возводить
} Factors;

//функция, которая находит разложение числа X и записывает его в структуру res
void Factorize(int X, Factors *res);
```

Использование хедеров в этой задаче запрещается. Функция **Factorize** должна быть реализована в исходном файле **factorize.c**. Всё остальное (точка входа, чтение/запись и пр.) должно быть в исходном файле **main.c**. Отправлять нужно только эти два исходных файла.

Учтите, что файлы в вашей программе будут частично подменяться на файлы “жюри”, чтобы проверить, что вы действительно разделили программу на части как описано.

Формат входных данных

В первой строке дано целое число M — количество запросов ($1 \leq M \leq 1\,000$). В каждой из остальных M строк записано одно целое число X ($1 \leq X \leq 10^9$).

Формат выходных данных

Для каждого числа X нужно вывести его разложение на простые множители в отдельной строке. Формат вывода должен быть полностью аналогичен формату примера. Сначала пишется число X , потом знак равенства. Если простых в разложении нет ($k = 0$), то надо написать просто единицу. Иначе нужно вывести все степени в виде p^t , разделённые знаком умножения. Между соседними токенами должен стоять ровно один пробел.

Пример

input.txt	output.txt
15	1 = 1
1	2 = 2 ¹
2	3 = 3 ¹
3	4 = 2 ²
4	5 = 5 ¹
5	6 = 2 ¹ * 3 ¹
6	7 = 7 ¹
7	8 = 2 ³
8	9 = 3 ²
9	10 = 2 ¹ * 5 ¹
10	1000000000 = 2 ⁹ * 5 ⁹
1000000000	999999999 = 3 ⁴ * 37 ¹ * 333667 ¹
999999999	999999987 = 3 ¹ * 7 ² * 6802721 ¹
999999987	999999997 = 71 ¹ * 2251 ¹ * 6257 ¹
999999997	999999937 = 999999937 ¹
999999937	