

Большая цифра

Дано выражение, записанное в 250-ричной системе счисления, в котором одна из цифр каждого числа заменена на x (цифра имеет одно и то же значение в каждом числе):

$$98x76x5+1x435$$

Определите наибольшее значение x , при котором выражение кратно числу 147.

В ответ **запишите только число**, решение прикладывать не нужно.

Пожалуйста, обратите внимание, у вас есть только **2 попытки** на решение данной задачи. Засчитывается лучшая.

Архиватор битовых строк

Ограничение времени	1 секунда
Ограничение памяти	64 Мб

Для сжатия строк, состоящих только из символов 0 и 1, используется следующий алгоритм:

1. Инициализация таблицы:
 - В таблицу записываются все символы алфавита:
"0" → код 0
"1" → код 1

Следующий свободный код — 2.

2. Процесс кодирования:
 - Пусть P — текущая подстрока (в начале пустая)
 - Для каждого очередного символа C в строке:
 - Если конкатенация $P + C$ уже есть в таблице, то присваиваем $P = P + C$
 - Иначе:
 - Выводим код для текущей подстроки P (она гарантированно есть в таблице)
 - Добавляем $P + C$ в таблицу с очередным свободным кодом
 - Присваиваем $P = C$

После обработки всех символов выводим код для оставшейся подстроки P

3. Битовое представление:
 - Размер кода в битах динамически увеличивается:
 - Начинаем с 1 бита на код (хватает для кодов 0 и 1)
 - Как только в таблице появляется код со значением 2^k

2k, переходим на k+1

- k+1 бит для всех последующих кодов

Пример:

Коды 0-1: 1 бит (0, 1)

После появления кода 2 (бинарно 10): 2 бита на код

После появления кода 4 (бинарно 100): 3 бита на код

После появления кода 8 (бинарно 1000): 4 бита на код и т. д.

Напишите программу, которая считывает бинарную строку S и кодирует её по описанному алгоритму.

Формат ввода

Одна строка S, состоящая только из символов 0 и 1.

Формат вывода

Выведите две строки:

- коды через пробел (целые числа)
- коэффициент сжатия, округлённый до 6 знаков после запятой

Расчёт коэффициента сжатия:

$$R = \frac{\text{исходный_размер_в_битах}}{\text{размер_закодированной_последовательности_в_битах}}$$

Исходный размер: длина_строки бит (каждый символ '0' или '1' считается за 1 бит)

Закодированный размер: сумма битов для всех кодов с учётом динамического изменения размера кода

Пример

Ввод	Вывод
01001	0 1 0 2 0.625

Примечания

Разбор примера

Инициализация

Словарь: {"0": 0, "1": 1}

Следующий свободный код: 2

Шаг 1

$p=""$, $c='0' \rightarrow p="0"$

Шаг 2

$p="0"$, $c='1' \rightarrow pc="01"$ нет в словаре

Вывод код 0 (1 бит)

Добавляем "01" $\rightarrow$ код 2

$next_code=3$, проверка: код 2 $== 2^1 \rightarrow$ битность=2 $p="1"$

Шаг 3

$p="1"$, $c='0' \rightarrow pc="10"$ нет в словаре

Вывод код 1 (2 бита, т.к. уже перешли на 2 бита)

Добавляем "10" $\rightarrow$ код 3

$next_code=4$, проверка: код 3 $!= 4 \rightarrow$ битность=2

$p="0"$

Шаг 4

$p="0"$, $c='0' \rightarrow pc="00"$ нет в словаре

Вывод код 0 (2 бита)

Добавляем "00" $\rightarrow$ код 4

$next_code=5$, проверка: код 4 $== 4 \rightarrow$ битность=3

$p="0"$

Шаг 5

$p="0"$, $c='1' \rightarrow pc="01"$ есть в словаре (код 2)

$p="01"$

Конец: вывод код для "01" $\rightarrow 2$ (3 бита)

Итоговые результаты

Итого: [0, 1, 0, 2]

Биты: $1 + 2 + 2 + 3 = 8$ бит

Коэффициент: $5 / 8 = 0.625$

В примере строка слишком короткая для эффективного сжатия