# Знакомство с циклом while

# Скидки

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

В магазине акция: скидка 5% на товары, цена которых превышает 1000 рублей. Напишите программу, отчасти имитирующую работу кассового аппарата: вводятся цены покупаемых товаров, нужно вывести общую стоимость товаров с учётом скидки.

**Формат ввода**

Несколько действительных чисел — цены на товары. Каждое число записано в отдельной строке.  
Последнее число — отрицательное — сигнал остановки.

**Формат вывода**

Одно действительное число — общая стоимость товаров с учётом скидки.

**Пример**

| Ввод | Вывод |
| --- | --- |
| 25  2000  370.35  -1 | 2295.35 |

# password123

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Как известно, когда мы придумываем пароль от аккаунта ВКонтакте, электронной почты или другого сайта, к этому паролю часто предъявляются определённые требования по сложности.

Напишите программу, которая имитирует проверку пароля, придуманного пользователем. Пользователь вводит пароль, потом ещё раз его же, для подтверждения.

* если пароль, который ввёл пользователь, короче 8 символов – программа выводит "Короткий!" и просит повторить ввод.
* если пароль достаточно длинный, но введённый во второй раз пароль не совпадает с первым – программа выводит "Различаются" и просит повторить ввод.
* если же и эта проверка пройдена успешно, программа выводит "OK" (латинскими буквами) и завершает свою работу.

**Формат ввода**

Две строки — первый и второй пароль, введенные пользователем.

**Формат вывода**

Одна строка — результат проверки пароля.

### Пример 1

| Ввод | Вывод |
| --- | --- |
| пароль  паролъ | Короткий! |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| пароль123  пароль123 | OK |

# Учитель

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Когда Учитель достиг просветления, он понял, что должен раздать свои богатства, причём сделать это следующим образом: в первый день разделить все свои золотые монеты на 8 равных частей (счастливое число!), излишки (если таковые будут иметься) пожертвовать храму Будды, оставить себе одну восьмую часть, остальные раздать бедным.  
Во второй день вновь разделить оставшиеся монеты на 8 частей и повторить вышеуказанные манипуляции. И продолжать так до тех пор, пока у него не останется так мало монет, что при делении их на 8 равных частей они все окажутся излишком.

Оставшиеся монеты можно оставить себе. Кроме того, Учитель не тратит свои деньги (во всяком случае, в дни после просветления): его кормят ученики, а в быту он аскетичен.

Хотя Учитель знает, конечно, сколько у него золотых монет изначально, но он не может сообразить, сколько монет окажется в конце — всё-таки он Учитель духовных практик, а не математики или программирования. Помогите ему.

Иными словами, найдите первую цифру введённого числа при записи его в системе счисления с основанием 8.

**Формат ввода**

Одно целое число — изначальное количество монет у Учителя.

**Формат вывода**

Одно целое число — количество монет, которое останется у Учителя в конце.

**Пример**

| Ввод | Вывод |
| --- | --- |
| 129 | 2 |

# Таких берут в космонавты

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

После полета Юрия Гагарина в 1961 практически каждый мальчик СССР хотел стать космонавтом. Прошло уже более полувека, но профессия космонавт все так же престижна. К сожалению, не каждый желающий может пройти отбор, существуют высокие требования к уровню подготовки будущих космонавтов, а также ограничения по антропометрическим показателям. Например, рост космонавта не может быть больше 190 см и меньше 150 см.

Напишите программу, которая считывает рост претендентов в отряд космонавтов до тех пор, пока не будет введен «!». А затем выводит на первой строчке количество подходящих кандидатур, а на второй строке – минимальный и максимальный рост участников, отобранных в новый отряд космонавтов.

Гарантируется, что в отряд отберутся как минимум два летчика-космонавта.

**Формат ввода**

Несколько строк с ростом космонавтов и последняя строка «!».

**Формат вывода**

Две строки: количество кандидатур на первой, и минимальный и максимальный рост через пробел – на второй.

**Пример**

| Ввод | Вывод |
| --- | --- |
| 192  189  145  162  172  ! | 3  162 189 |

# 1024 и все-все-все

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Во многих задачах, связанных с компьютерами, особенно близких к аппаратной части, важную роль играют числа, являющиеся степенями двойки: 1, 2, 4, 8 и так далее. Напишите программу, которая проверяет, является ли введённое натуральное число степенью двойки. Если да, то выводится сама эта степень; если нет, выводится «НЕТ».

**Формат ввода**

Одно целое число.

**Формат вывода**

Одно целое число (показатель степени) или строка «НЕТ».

**Пример**

| Ввод | Вывод |
| --- | --- |
| 302231454903657293676544 | 78 |

# Навигация – 1

Таракан ползает по узкому туннелю. Требуется давать ему указания, куда ползти. Проблема в том, что тараканы назад не ходят, поэтому кроме команды «вперед» есть команда «разворот». После команды «вперед» следует целое количество шагов, которое нужно пройти. Туннель открыт с одной стороны.

Таракан начинает в начале туннеля (координата 0) и смотрит направо. Слева туннель закрыт. Требуется вывести количество команд до момента, как таракан покинул туннель

**Формат ввода**

Изначально вводится длина туннеля. Затем следуют несколько команд. Каждая команда состоит из одного слова и, возможно, одного натурального числа на отдельной строке. Слово — одно из набора «вперед» и «разворот».

**Формат вывода**

Программа выводит на отдельных строках количество введенных команд до момента, как таракан покидает туннель. Гарантируется, что таракан покинет туннель.

### Пример 1

| Ввод | Вывод |
| --- | --- |
| 2  вперёд  2  разворот  вперёд  1 | 1 |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| 9  вперед  4  разворот  вперед  2  вперед  1  разворот  вперед  10  вперед  2 | 6 |

# Навигация - 2

Дополнить предыдущую задачу: туннель открыт с обеих сторон. Вводится также начальная координата таракана. Кроме количества команд требуется вывести, из какого конца выполз таракан (левый или правый). Сначала таракан смотрит направо.

**Формат ввода**

Вводится длина туннеля и координата таракана в нем. Затем следуют несколько команд. Каждая команда состоит из одного слова и, возможно, одного натурального числа на отдельной строке. Слово — одно из набора «вперед» и «разворот».

**Формат вывода**

### Программа выводит на отдельных строках количество введенных команд до момента, как таракан покидает туннель, а также конец, из которого он выполз (левый или правый). Гарантируется, что таракан покинет туннель.

### Пример 1

| Ввод | Вывод |
| --- | --- |
| 2  1  вперёд  2  разворот  вперёд  1 | 1  справа |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| 9  3  вперед  4  разворот  вперед  2  вперед  6  разворот  вперед  10  вперед  2 | 4 |

# Ищем клад — 1

Kлассная работа

макс. 3 балл.

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Мы находимся на острове, на котором закопан клад. Нам известно, где именно закопан клад, но этого мало: остров полон опасностей, и нужно перемещаться строго по указаниям карты, которая, к счастью, тоже имеется в нашем распоряжении.

Известны координаты клада и указания, которым нужно следовать, чтобы его найти. Однако указаний слишком много: они могут продолжаться даже после достижения его координат. Требуется сосчитать количество указаний до этой точки и направление взгляда в ней.

Мы находимся в точке с координатами (0, 0) и смотрим на север. Каждое указание карты состоит из одного слова и, возможно, одного натурального числа. Слово – одно из набора: «вперёд», «налево», «направо», «разворот» или «стоп».

После слова «вперёд» следует количество шагов, которое следует пройти в том направлении, куда мы в данный момент смотрим. Слова «налево» или «направо» означают, что нужно изменить направление взгляда под прямым углом, «разворот» — что прямо на обратное. Команда «стоп» означает остановку.

Найдите минимальное количество указаний карты, которое нужно выполнить, чтобы прийти к кладу.

**Формат ввода**

Сначала вводятся два числа на отдельных строчках: координаты клада по оси икс (запад-восток) и игрек (юг-север).  
Затем следует некоторое количество указаний карты. Каждое указание карты состоит из одного слова и, возможно, одного натурального числа на отдельной строке. Слово — одно из набора: «вперёд», «налево», «направо», «разворот» или «стоп».

**Формат вывода**

Программа выводит на отдельных строках минимальное количество указаний карты, которое нужно выполнить, чтобы прийти к кладу, и направление взгляда в этот момент (одно из: «север», «юг», «запад», «восток»). Гарантируется, что карта приводит к кладу.

### Пример 1

| Ввод | Вывод |
| --- | --- |
| -2  9  вперёд  9  налево  вперёд  2  разворот  вперёд  17  стоп | 3  запад |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| 0  1  вперёд  2  разворот  вперёд  1 | 3  юг |

integer division result too large for a float

результат целочисленного деления слишком велик для float