# Функции. Области видимости переменных

# Длина окружности и площадь круга

Kлассная работа

макс. 1 балл.

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

Напишите две функции circle\_length(radius) и circle\_area(radius), принимающие радиус окружности и возвращающие длину этой окружности и площадь круга соответственно.

Также напишите функцию main(), которая спрашивает у пользователя радиус окружности, а затем при помощи функций circle\_length(radius) и circle\_area(radius) выводит на экран длину и площадь круга, разделённые пробелом.

Учтите, что радиус окружности – это вещественное число.

Число ππ взять равным 3.14159.

### Пример 1

| Ввод | Вывод |
| --- | --- |
| print(circle\_length(5)) | 31.4159 |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| print(circle\_area(10)) | 314.159 |

### Пример 3

| Ввод | Вывод |
| --- | --- |
| main()  0.5 | 3.14159 0.7853975 |

## Примечания

Будьте внимательны: ваши функции должны называться в точности так, как указано в задании и принимать аргументы в том же порядке.

Вы можете запускать функцию main(), пока тестируете программу, но в отправленном на проверку решении эта функция должна быть определена и не должна запускаться.

# Корни квадратного уравнения

Kлассная работа

макс. 1 балл.

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

Напишите две функции larger\_root(p,q) и smaller\_root(p,q), которые возвращают больший и меньший корни **приведённого** квадратного уравнения x2 + px + q = 0. В качестве аргументов функции должны принимать коэффициенты p и q. Считать, что дискриминант неотрицателен.

Также сделайте функцию discriminant(a,b,c), которая вычисляет и возвращает дискриминант квадратного уравнения, записанного в общем виде: ax2 + bx + c = 0.

Функции вычисления корней должны использовать функцию вычисления дискриминанта.

Наконец, сделайте функцию main(), которая запрашивает у пользователя два вещественных числа p и q, и в ответ печатает на первой строке дискриминант уравнения x2 + px + q = 0, на второй строке – меньший и больший корни. Она должна выводить оба корня, даже если они совпадают.

## Формат ввода

Ввод функции main()  
-2.5  
1

## Формат вывода

Вывод функции main():  
2.25  
0.5 2.0

### Пример 1

| Ввод | Вывод |
| --- | --- |
| print(smaller\_root(2, 1)) | -1.0 |

### Пример 2

| Ввод | Вывод |
| --- | --- |
| print(larger\_root(2, 1)) | -1 |

### Пример 3

| Ввод | Вывод |
| --- | --- |
| print(discriminant(1, 2, 1)) | 0 |

## Примечания

Будьте внимательны: ваши функции должны называться в точности так, как указано в задании и принимать аргументы в том же порядке.

Вы можете запускать функцию main(), пока тестируете программу, но в отправленном на проверку решении эта функция должна быть определена, и не должна запускаться.

# Длинный чек

Kлассная работа

макс. 2 балл.

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

Для работы небольшого магазинчика вас попросили сделать небольшой набор функций, обеспечивающих работу с чеками. В качестве пробной версии вам нужно реализовать всего два действия: добавление покупки к набору покупаемых товаров, и печать чека. После того как чек напечатан должно быть возможно вводить новый список товаров.

По правилам оформления документов в каждом чеке вы должны напечатать сначала номер чека (нумерация начинается с единицы) и число товаров, затем список товаров с ценами и в конце — итоговую стоимость. Из-за того, что вы не знаете число товаров, пока кассир не «пробьет» последний товар, нельзя печатать чек по мере ввода товаров. Приходится сначала добавить все необходимые товары и только затем печатать весь чек целиком. В конце чека печатается черта (состоящая из пяти знаков минус), по которой он отрывается от ленты.

Организуйте две функции: **add\_item(itemName, itemCost)**и **print\_receipt()**, которые реализуют указанный функционал. Форматирование сделайте по шаблону в примере.

В случае, когда чек пуст, print\_receipt не должна печатать чек. Не забудьте, что после того как чек напечатан, набор выбранных товаров должен обнулиться, чтобы можно было вводить товары для нового чека.

## Пример

| Ввод | Вывод |
| --- | --- |
| add\_item('Блокнот', 100)  print\_receipt()  add\_item('Ручка', 70)  print\_receipt()  print\_receipt()  add\_item('Булочка', 15)  add\_item('Булочка', 15)  add\_item('Чай', 5)  print\_receipt()  add\_item('Булочка', 15)  add\_item('Булочка', 15)  # (Отменить чек) - этот чек не печатаем | Чек 1. Всего предметов: 1  Блокнот - 100  Итого: 100  -----  Чек 2. Всего предметов: 1  Ручка - 70  Итого: 70  -----  Чек 3. Всего предметов: 3  Булочка - 15  Булочка - 15  Чай - 5  Итого: 35  ----- |
|  |  |

## Примечания

Это относится к большинству задач, но не лишним будет напомнить, что решить задачу можно несколькими способами.

# НРЗБРЧВ

Kлассная работа

макс. 2 балл.

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

В письменности языка НРЗБРЧВ отсутствуют гласные буквы и знаки препинания, в остальном язык ничем не отличается от русского языка. Чтобы из слова русского языка получить слово языка НРЗБРЧВ, достаточно выкинуть из слова все гласные. Чтобы перевести предложение, нужно перевести каждое слово и выкинуть знаки препинания. Регистр букв следует оставить неизменным.  
Напишите функцию **translate(text)**, которая переводит текст с русского языка на НРЗБРЧВ.  
Свой результат функция должна записать во внешнюю переменную **translated\_text**.

## Формат ввода

translated\_text = None  
translate("Удивительный факт, но текст на языке НЕРАЗБОРЧИВО оказывается довольно просто читать. Достаточно небольшой тренировки - и вы сможете это делать.")

## Формат вывода

translated\_text == "двтльнй фкт н ткст н зк НРЗБРЧВ кзвтс двльн прст чтть Дсттчн нбльшй трнрвк в смжт т длть"

## Примечания

Слова в НРЗБРЧВ разделяются ровно одним пробелом. Обратите внимание, что при переводе некоторые короткие слова «съедаются» полностью. Следите за тем, чтобы в этом случае не появлялись лишние пробелы.

# Айболит

Kлассная работа

макс. 2 балл.

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

На прием к доктору Айболиту как всегда большая очередь. Так как желающих очень много, то было принято решение в регистратуре открыть еще два окна, при этом написать программу для автоматизации приема больных.

Напишите функцию **hello(name)**, которая принимает на вход имя пациента, приветствует его и сообщает, что его медицинскую карту ищут.

Для поиска медицинской карты напишите функцию **search\_card(name)**, которая тоже принимает на вход имя пациента. Эта функция выводит информацию о номере карточки пациента или о том, что она не найдена. Список имён пациентов, пронумерованный с 1, находится в списке **base**, который является глобальным. Номер пациента соответствует номеру карты.

## Пример

| Ввод | Вывод |
| --- | --- |
| base = ["Иван", "Юлия Иванкова"]  hello("Иван")  search\_card("Иван")  hello("Юлия Иванова")  search\_card("Юлия Иванова") | Здравствуйте, Иван! Вашу карту ищут...  Ваша карта с номером 1 найдена  Здравствуйте, Юлия Иванова! Вашу карту ищут...  Ваша карта не найдена |

# Делайте ваши ставки

Kлассная работа

макс. 3 балл.

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

Боб хочет собрать всех своих друзей-предпринимателей вместе. Но для того, чтобы заставить занятых друзей прийти к нему, следует сделать что-то необычное. У Боба есть ферма, на которой живут 10 лошадей. Он хочет провести скачки со ставками.

Помогите Бобу написать функцию **do\_bet()** для совершения ставок, которая первым аргументом принимает номер лошади, а вторым –- ставку. Лошади пронумерованы от 1 до 10. На одну и ту же лошадь нельзя сделать ставку дважды. Также нельзя делать нулевую ставку.

## Пример

| Ввод | Вывод |
| --- | --- |
| do\_bet(1, 10)  do\_bet(1, 100)  do\_bet(2, 0)  do\_bet(2, 200) | Ваша ставка в размере 10 на лошадь 1 принята  Что-то пошло не так, попробуйте еще раз  Что-то пошло не так, попробуйте еще раз  Ваша ставка в размере 200 на лошадь 2 принята |

# Бюрократия

Kлассная работа

макс. 3 балл.

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

Отдел кадров одной крупной компании осознал, что тратит много времени на заполнение одних и тех же данных в куче разных бумажек для заявки на отпуск. Они попросили вас упростить их работу, чтобы им приходилось только один раз ввести имя, фамилию и некоторые вспомогательные данные. А сразу несколько справок печатать подстановкой указанных данных в разные шаблоны.

Чтобы облегчить страдания бюрократов, вы взялись им помочь. Они и раньше использовали Python для печати документов, но для каждого документа они передавали все параметры раз за разом — примерно так:

print\_application\_for\_leave("Иван Петров", "1 июня – 20 июня")  
print\_holiday\_money\_claim("Иван Петров", "15 тысяч пиастров")  
print\_attorney\_letter("Иван Петров", "1 июня – 20 июня", "Василий Васильев")

Больше всего отдел кадров раздражало, что приходится повторять параметры в каждом документе, к тому же есть риск ошибиться. Можно было бы решить эту задачу с помощью введения переменных, но набор документов нужен каждый раз немного разный (иногда какой-то документ не нужен, иногда наоборот нужно несколько копий). Кроме того, даже с использованием переменных функциям все равно пришлось бы указывать кучу параметров и не путать, какие параметры функции нужны, а какие — необязательны. Да и работать с переменными отдел кадров пока не умеет.

Вы сошлись на том, что теперь отделу кадров придется заполнять данные, вызывая функции языка Python с правильными параметрами. Одна функция — **setup\_profile(name, vacation\_dates)** — предназначена для того, чтобы задать всю информацию про работника. А ещё несколько функций пользуются установленными значениями, чтобы оформить документы на отпуск: заявление на отпуск **print\_application\_for\_leave()**, заявление на выплату отпускных **print\_holiday\_money\_claim(amount)**, и доверенность на передачу своих служебных полномочий заместителю **print\_attorney\_letter(to\_whom)**.

Напишите эти четыре функции, так чтобы каждая печатала соответствующую справку. Примеры справок приведены в тестовом выводе. Не ошибайтесь в точных формулировках документов, бюрократы это не любят.

## Пример

| Ввод | Вывод |
| --- | --- |
| setup\_profile("Иван Петров", "1 июня – 20 июня")  print\_application\_for\_leave()  print\_application\_for\_leave()  print\_holiday\_money\_claim("15 тысяч пиастров")  print\_attorney\_letter("Василий Васильев") | Заявление на отпуск в период 1 июня – 20 июня. Иван Петров  Заявление на отпуск в период 1 июня – 20 июня. Иван Петров  Прошу выплатить 15 тысяч пиастров отпускных денег. Иван Петров  На время отпуска в период 1 июня – 20 июня моим заместителем назначается Василий Васильев. Иван Петров |

## Примечания

Если setup\_profile вызвана второй раз, необходимо перейти к документам второго сотрудника и т. д.