Лабораторная работа 5

Функции и цикл `while`

Во всех задачах с функциями сначала опишите функцию, а затем вызовите её ниже в основной программе.

Проверка функций

Для целых чисел и строк можно использовать `assert result == expected`. Для булевых значений достаточно написать `assert result`, если функция возвращает `true` и `assert not result`, если возвращает `false`. Для вещественных чисел обычно достаточно `isclose` с относительной погрешностью `rel_tol`. Если ожидаемое значение равно нулю или очень близко к нему, добавляют и абсолютную погрешность `abs_tol`.

Задача 1. Функция без `return`

Опишите функцию `show_sign(number)`, которая выводит строку `положительное`, `отрицательное` или `нулевое`. Функция ничего не должна возвращать.

Вопрос: можно ли такую функцию поместить в `print`?

Ответ: ...

Задача 2. Функция со строковым результатом

Опишите функцию `sign_description(number)`, которая возвращает строку `положительное`, `отрицательное` или `нулевое`. Выведите результат её работы.

Задача 3. Количество корней

Опишите функцию `root_count(a, b, c)`, возвращающую количество корней квадратного уравнения $a \cdot x^2 + b \cdot x + c = 0$, где $a \neq 0$.

Пример проверки. Запустите ячейку с функцией, затем эту ячейку. Допишите еще несколько тестов.

Задача 4. Площадь круга

Опишите функцию `circle_area(radius)`, возвращающую площадь круга. Внутри функции используйте локальную переменную $\pi = 3.14$.

Пример проверки. Запустите ячейку с функцией, затем эту ячейку. Допишите еще несколько тестов.

Задача 5. Периметр равнобедренного треугольника

Опишите функцию `triangle_perimeter(base, height)`, возвращающую периметр равнобедренного треугольника по основанию и высоте, проведённой к основанию.

Задача 6. Цена с параметрами по умолчанию

Опишите функцию `format_price(price, digits=2, currency='руб.')`, возвращающую строку с ценой, округлённой до `digits` знаков после запятой. Вызовите её обычным способом и с именованными аргументами.

Задача 7. Время в секундах

Опишите функцию `time_to_hms(seconds)`, возвращающую количество полных часов, минут и секунд в данном количестве секунд. Распакуйте результат в три переменные.

Пример проверки. Запустите ячейку с функцией, затем эту ячейку. Допишите еще несколько тестов.

Задача 8. Информация о прямоугольнике

Опишите функцию `rectangle_info(a, b)`, возвращающую площадь и периметр прямоугольника. В основной программе распакуйте оба результата.

Пример проверки. Запустите ячейку с функцией, затем эту ячейку. Допишите еще несколько тестов.

Задача 9. Високосный год

Опишите булеву функцию `is_leap_year(year)`. Год високосный, если он делится на 4, кроме годов, делящихся на 100 и не делящихся на 400.

Пример проверки. Запустите ячейку с функцией, затем эту ячейку. Допишите еще несколько тестов.

Задача 10. Количество цифр

Опишите функцию `digit_count(number)`, возвращающую количество цифр положительного целого числа.

Задача 11. Переворот числа

Опишите функцию `reverse_digits(number)`, возвращающую для положительного целого `number` число, записанное теми же цифрами в обратном порядке.

Задача 12. Двойной факториал

Опишите функцию `double_factorial(N)`, вычисляющую двойной факториал целого положительного числа N по формуле:

$$N!! = N \cdot (N-2) \cdot (N-4) \dots$$

Задача 13. Число Фибоначчи

Опишите функцию `fibonacci(N)`, возвращающую N-е ($N > 0$) число Фибоначчи: $F_1 = F_2 = 1$.

Задача 14. Игра «Угадай число»

Игрок задает нижнюю `A` и верхнюю `B` границы диапазона для загадываемого компьютером числа (вводит 2 целых числа с клавиатуры). После ввода с помощью `assert` следует проверить, что нижняя граница `A` не больше верхней `B`.

Далее игрок задает максимально доступное количество попыток `MAX_ATTEMPTS` (вводит 1 положительное целое число с клавиатуры). После ввода с помощью `assert` следует проверить, что данное число больше нуля и меньше числа элементов в диапазоне $[A, B]$.

Компьютер загадывает число `X` из заданного пользователем диапазона $[A, B]$. Игрок должен угадать это число за `MAX_ATTEMPTS` попыток.

Игрок вводит число `N` из заданного диапазона: если это то самое загаданное число, игра заканчивается; в противном случае компьютер сообщает, как `N` соотносится с загаданным `X` (то есть меньше ли `X`, чем полученное `N`, или больше), и игра продолжается — игрок снова пытается угадать число.

Если игрок не угадал число за отведённое число шагов, он проигрывает.

При реализации игры нужно обратить внимание на несколько деталей:

- Должны быть заданы корректные границы диапазона, то есть нижняя граница `A` не должна быть больше верхней `B`.
- На каждом шаге угадывания числа игрок видит номер попытки и диапазон значений, который он уже отгадал.
- Если игрок вводит число вне текущего диапазона, он проигрывает.
- Используйте `while True` и `break`.

Примеры запусков игры:

Пример выигрыша.
Введите нижнюю границу дипазона A: -10
Введите верхнюю границу дипазона B: 10
Введите максимальное количество попыток MAX_ATTEMPTS: 15

Задан диапазон: -10..10.
Компьютер загадал число. Попробуйте отгадать его! У вас есть 15 попыток.

Попытка #1/15.
Введите число из диапазона -10..10: 0
Загаданное число больше 0.

Попытка #2/15.
Введите число из диапазона 1..10: 5
Загаданное число меньше 5.

Попытка #3/15.
Введите число из диапазона 1..4: 3
Загаданное число больше 3.

Попытка #4/15.
Введите число из диапазона 4..4: 4

Число отгадано! Это 4.
Вы угадали число за 4 шагов.

Пример проигрыша: число выходит за границы диапазона.

Введите нижнюю границу дипазона A: 1
Введите верхнюю границу дипазона B: 5
Введите максимальное количество попыток MAX_ATTEMPTS: 4

Задан диапазон: 1..5.
Компьютер загадал число. Попробуйте отгадать его! У вас есть 4 попыток.

Попытка #1/5.
Введите число из диапазона 1..5: 4
Загаданное число меньше 4.

Попытка #2/5.
Введите число из диапазона 1..3: 5

Вы проиграли!
Было загадано число 2.

Пример проигрыша: исчерпано количество попыток.

Введите нижнюю границу дипазона A: 0
Введите верхнюю границу дипазона B: 50
Введите максимальное количество попыток MAX_ATTEMPTS: 5

Задан диапазон: 0..50.
Компьютер загадал число. Попробуйте отгадать его! У вас есть 5 попыток.

Попытка #1/5.
Введите число из диапазона 0..50: 25
Загаданное число больше 25.

Попытка #2/5.
Введите число из диапазона 26..50: 35
Загаданное число меньше 35.

Попытка #3/5.
Введите число из диапазона 26..34: 30
Загаданное число меньше 30.

Попытка #4/5.
Введите число из диапазона 26..29: 29
Загаданное число меньше 29.

Попытка #5/5.
Введите число из диапазона 26..28: 27
Загаданное число больше 27.

Вы проиграли! Исчерпано допустимое количество попыток.
Было загадано число 28.