**Курс 1. Курсовая работа**

Привет, это Артем из [Mail.ru](http://Mail.ru)!

Завершается первый курс программы, мы наконец-то научились разделять наш код на подпрограммы. Сегодня будем использовать эту возможность на полную мощность!

Мы будем создавать программу, которая, на этот раз, помогает изучать коды азбуки Морзе. Для этого наша программа будет конвертировать строки в морзе-код, а затем проверять ответы пользователя. Все это с помощью функций.

<aside> 💡 ВАЖНО! Сначала **изучить материалы всех пройденных уроков**, а только потом приступать к выполнению курсовой работы.

</aside>

Пример работы программы:

\*\*Программа:\*\*

Сегодня мы потренируемся расшифровывать азбуку Морзе

Нажмите Enter и начнем

\*\*Пользователь:\*\*

\*\*Программа:\*\*

Слово 1 – ... -. .- -.- .

\*\*Пользователь:\*\*

Snake

\*\*Программа:\*\*

Верно, Snake!

Слово 2 – - .. -. -.--

\*\*Пользователь:\*\*

Shiny

\*\*Программа:\*\*

Неверно, Tiny!

Слово 3 .-- . .-.. .-.. .--. .-.. .- -.-- . -..

\*\*Пользователь:\*\*

wellplayed

\*\*Программа:\*\*

Верно, wellplayed!

Всего задачек: 3

Отвечено верно: 2

Отвечено неверно: 1

**Обратите внимание, в рамках этой работы вам необходимо задокументировать код.**

**Шаг 0.** Составьте список английских слов и фраз, которые будете расшифровывать.

Например: code, bit, list, soul, next

**Шаг 1.** Напишите функцию morse\_encode(word), которая переводит слова на английском языке в последовательности точек и тирe. Например

morse\_encode("little")

>>> .-.. .. - - .-.. .

morse\_encode("snake")

>>> ... -. .- -.- .

morse\_encode("wellplayed")

>>> .-- . .-.. .-.. .--. .-.. .- -.-- . -..

Словарь с кодами возьмите отсюда <https://gist.github.com/mohayonao/094c71af14fe4791c5dd>

**Шаг 2.** Напишите функцию get\_word() которая получает случайное слово из списка.

get\_word()

>>> bit

get\_word()

>>> next

get\_word()

>> lead

**Шаг 3.** Создайте в начале программы список answers = []. Напишите функцию print\_statistics(answers) которая на основе списка answers типа [True, False, False, True, False] выводит статистику типа

Всего задачек: 5

Отвечено верно: 2

Отвечено неверно: 3

**Шаг 4.** При старте программы выведите приветственную информацию.

Сегодня мы потренируемся расшифровывать морзянку.

Нажмите Enter и начнем

**Шаг 5.** Запустите цикл задавания вопросов. В одной игре – 5 вопросов.

В каждой итерации

* получайте случайное слово с помощью ранее написанной функции
* кодируйте его с помощью ранее написанной функции
* выводите для пользователя
* получайте ввод
* сравнивайте с загаданным словом
* комментируйте верный или неверный ответ
* верность ответа складывайте в переменную answers.

Слова могут повторяться во время тренировки, это не страшно.

**Шаг 6.** Выведите статистику с помощью вызова ранее написанной функции

Протестируйте работу приложения и отправьте ссылку на colab

**Несколько рекомендаций по оформлению кода**

1. Размещайте функции в начале кода, сразу после импортов, отделяя каждую функцию двумя пустыми строками, это требование PEP 8.

# Верно

import random

def foo():

pass

def bar():

pass

# Неверно

def foo():

pass

import random

def bar():

pass

1. Добавляйте docstring к каждой функции, размещайте пояснения именно там, а не в комментариях.

# Верно

def get\_random\_word(words):

"""

Получает случайное слово

из списка слов

"""

word = random.choice(words)

return word

# Неверно

# Получает случайное слово

# из списка слов

def get\_random\_word(words):

word = random.choice(words)

return word

**Критерии выполнения:**

* [ ] Вопросы задаются корректно.
* [ ] Все функции задокументированы (те добавлены комментарии или докстринги)
* [ ] Слова преобразуются корректно.
* [ ] Return используется корректно.
* [ ] Статистика считается верно.
* [ ] Функции написаны и работают как нужно.