1. Создайте генераторную функцию которая в качестве аргумента принимать путь к файлу «unsorted\_names.txt» и букву английского алфавита, открывает файл по данному пути и генерирует последовательность из имен начинающихся на указанную букву

>>> names\_with\_letter = names\_gen(«unsorted\_names.txt», «A»)

>>> next(names\_with\_letter)

Amelia

>>> next(names\_with\_letter)

Adrienne

или

>>> for name in names\_with\_letter:

print(i, end=““)

Amelia

Adrienne

…

2. Реализуйте свой пользовательский класс итератора с именем MySquareIterator, который дает квадраты элементов коллекции, по которой он итерируется.

Пример:

>>> lst = [1, 2, 3, 4, 5]

>>> itr = MySquareIterator(lst)

>>> for el in itr:

print(el)

>>> 1 4 9 16 25

3. Создайте класс Validator который позволяет проводить проверку данных пользователя при

регистрации передаваемых в виде кортежа (login, password, email)

В данном классе реализовать метод validate(user\_data), который позволяет проверить передаваемый кортеж по правилам:

login — от 6 до 10 символов английского алфавит и цифр 0-9 в любой последовательности

password — не менее 8 символов, буквы в верхнем и нижнем регистре, не менее одного специального символа (+-/\*! и т.д)

email — присутствует символ @, оканчивается . и 2 символами (.by)

Проверку на соответствие правилам выполнить регулярными выражениями. По результатам работы метода validate пользователь получит True если все 3 элемента прошли проверку, в противном случае - False

4\*. Реализуйте класс итератора EvenRange, который принимает начало и конец интервала в качестве аргументов инициализации и выдает только четные числа во время итерации.

Если пользователь попытается выполнить итерацию после того, как он выдал все возможные числа следует вывести «Out of number!».

\_Примечание: вообще не используйте функцию range()\_

Пример:

>>> er1 = EvenRange(7,11)

next(er1)

>>> 8

next(er1)

>>> 10

next(er1)

>>> "Out of numbers!"

next(er1)

>>> "Out of numbers!"

>>> er2 = EvenRange(3, 14)

>>> for number in er2:

print(number)

>>> 4 6 8 10 12 "Out of numbers!"

5\*. Реализуйте генератор, который будет бесконечно генерировать числа Фибоначчи (https://en.wikipedia.org/wiki/Fibonacci\_number).

Пример:

>>> gen = endless\_fib\_generator()

>>> while True:

print(next(gen))

>>> 1 1 2 3 5 8 13 ...