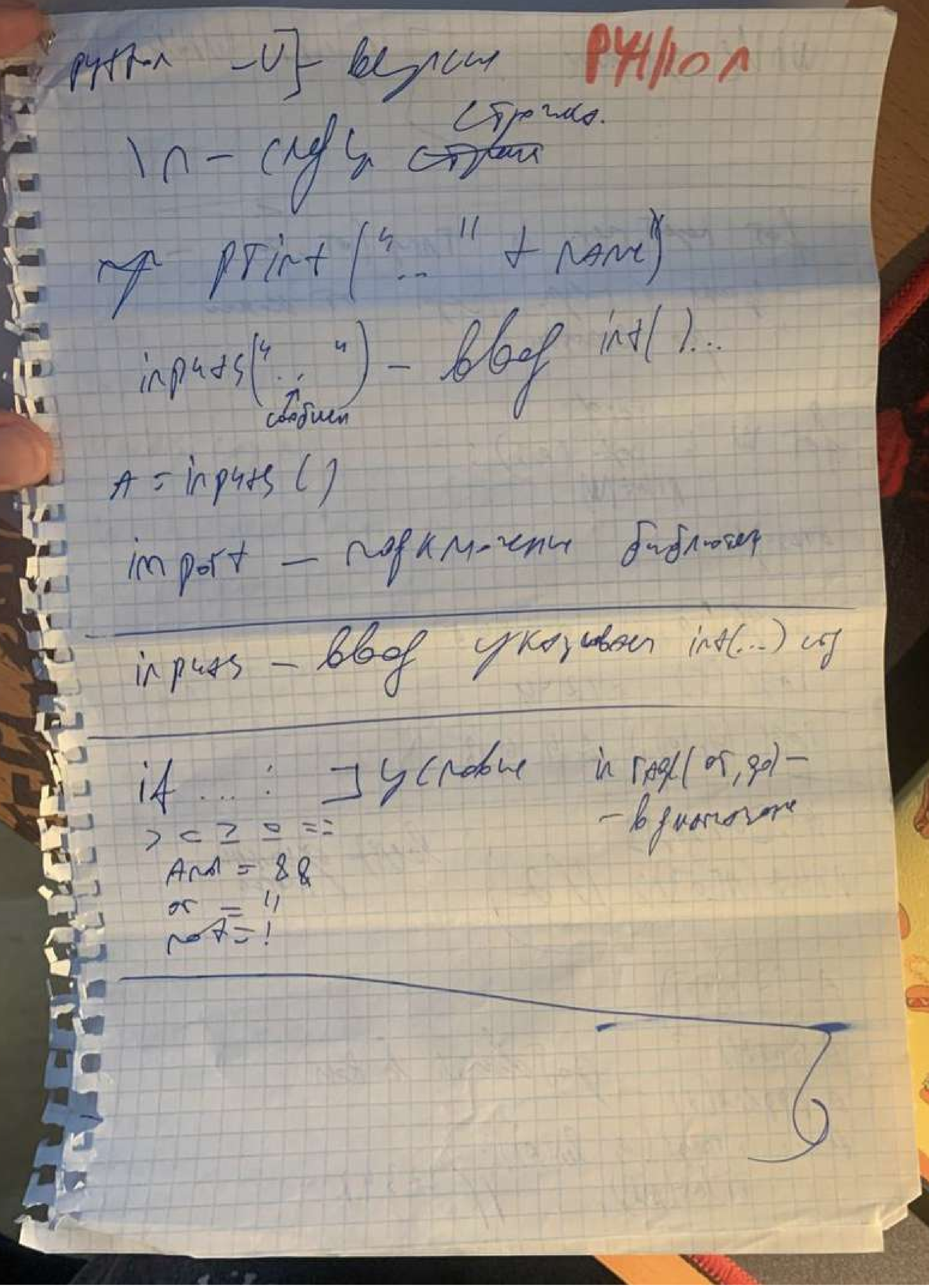


python -V] версия Python

\n – след. ~~строки~~ строчка.

print("..." + name")

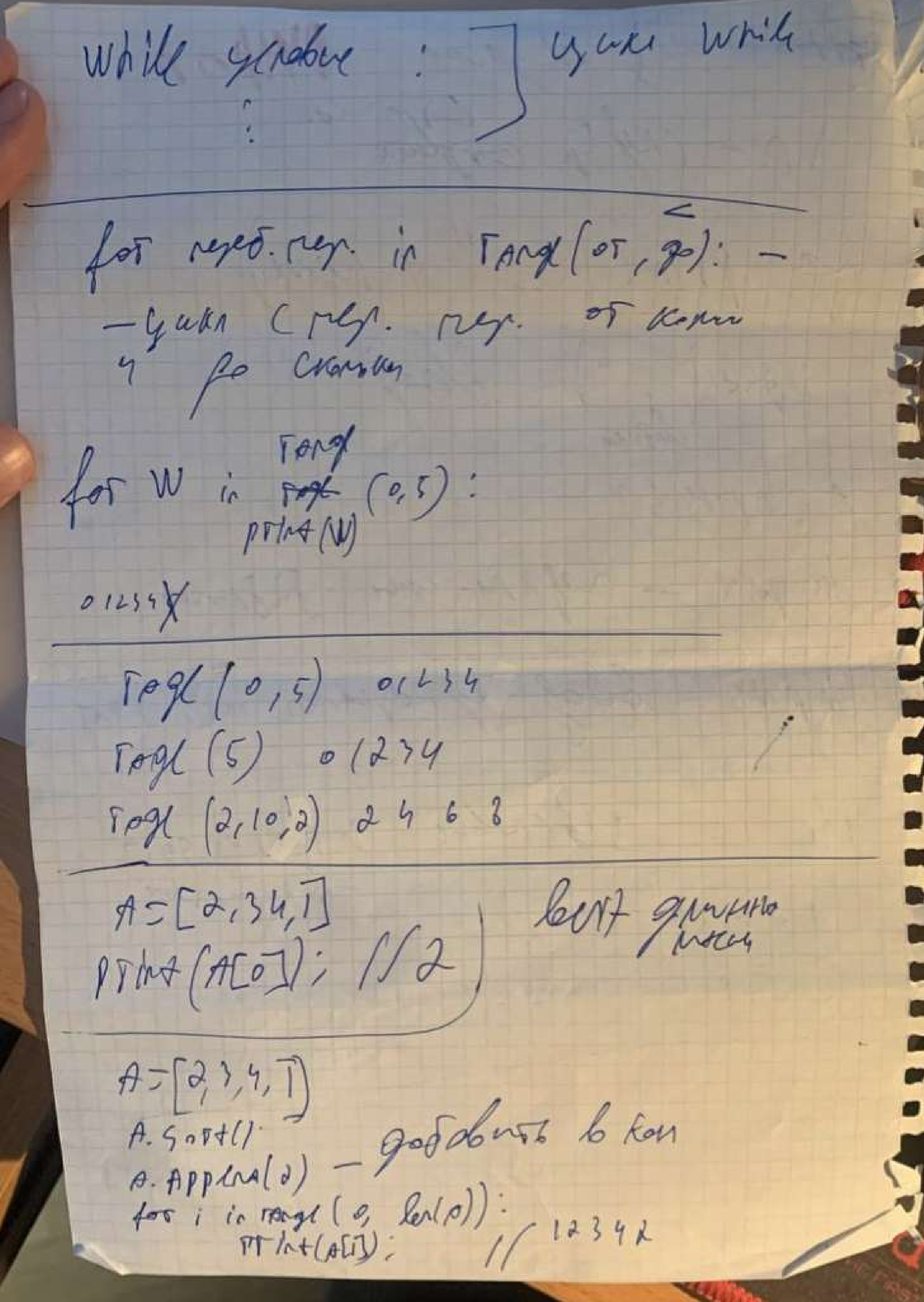
input("...") – ввод int()...
сообщен

A = input()

import – подключение библиотек

input – ввод указываем int(...) итд

if ... :] условие in range(05, 90) – в диапазоне
> < ≥ ≤ ==
And = &&
or = ||
not = !



```
while условие :   ] цикл while
   :
```

for перем.пер. in range(от, до): — <
— цикл с перем. пер. от какого
ч до скольки

```
for W in ~~rage~~ range(0,5):
    print(W)
```

0 1 2 3 4 ~~5~~

```
range(0,5)     0 1 2 3 4
range(5)       0 1 2 3 4
range(2,10,2)  2 4 6 8
```

```
A=[2,3,4,1]
print(A[0]); // 2
```

len — длинна масс

```
A=[2,3,4,1]
A.sort()
A.append(2) — добавить в кон
for i in range(0, len(A)):
    print(A[i]);   // 1 2 3 4 2
```

def имя ф-и (параметры): —

— ф-я

в питоне есть многофайловость.

```
Main                          Voids
import Voids                  def sayhi():
Voids.sayhi()                     print('hi')
```

R-read W-write A-дополнить

With open ("путь", "тип открытия") as samefaile: —

— открытие документа

.write — написать

.readall — читать все

.readline() — прочесть одну строку (A = file.readline())

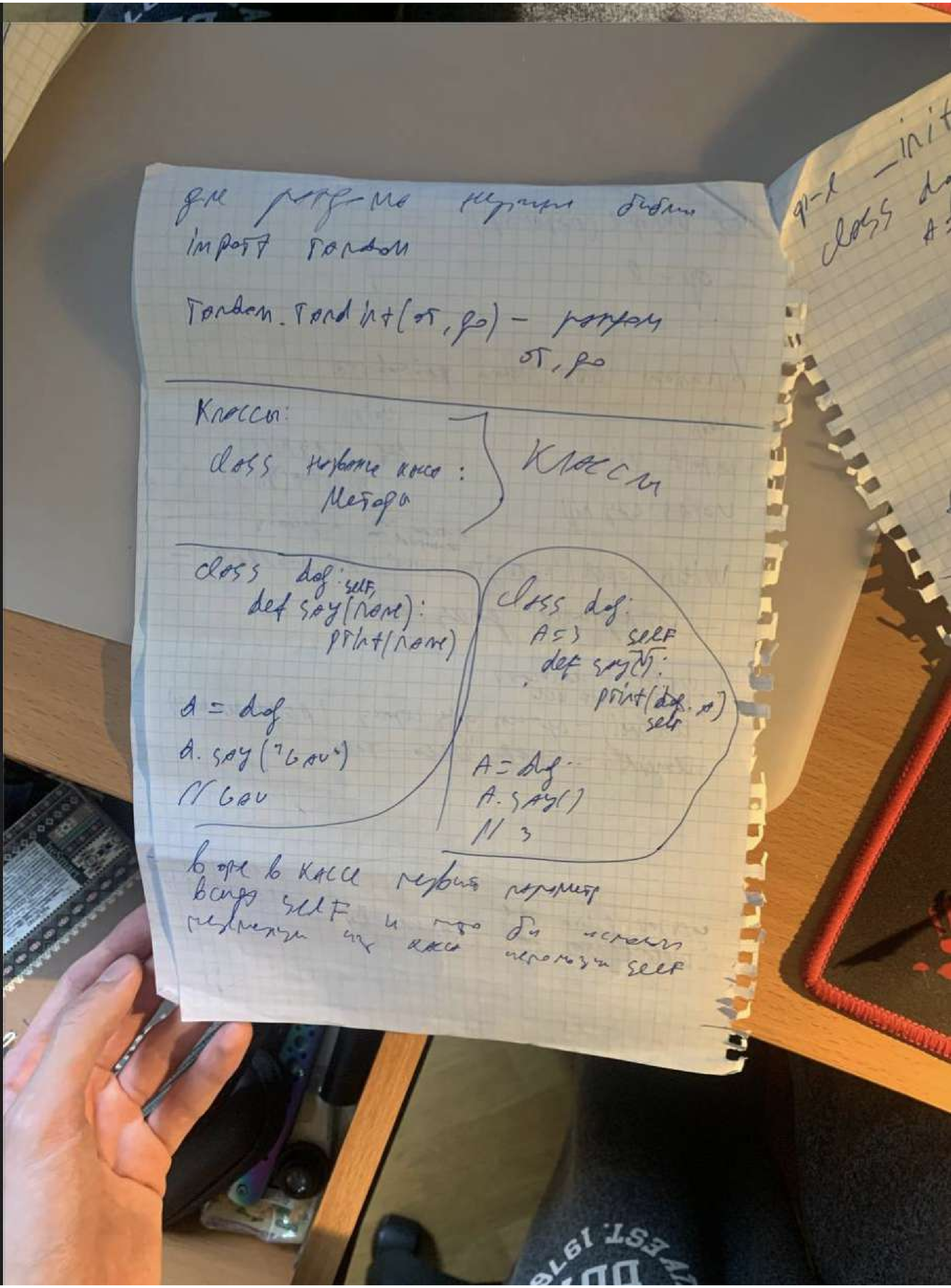
~~.readlines() — (читать весь файл~~

у строк есть ф-и

.replace(стар. нов) — замена букв

A = lol

A = A.replace('l', ' ')



для рандома нужно библ
import random

random.randint(от, до) - рандом от, до

Классы:

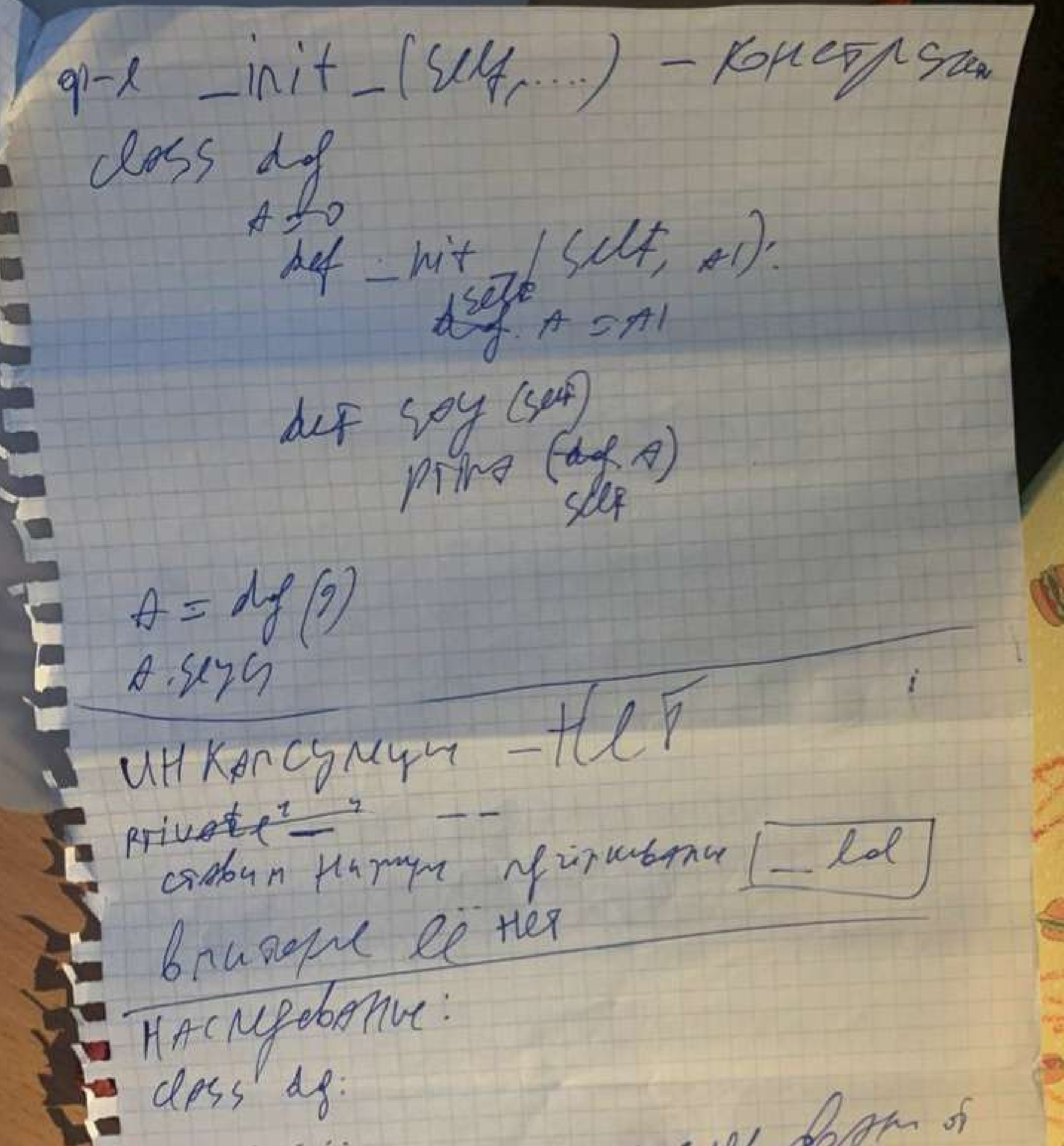
class название класса :
    Методы
} Классы

```
class dog:
    def say(self, name):
        print(name)

A = dog
A.say("GAV")
// GAV
```

```
class dog:
    A=3
    def say(self):
        print(~~dog~~ self.A)

A = dog.
A.say()
// 3
```

вообще в классе первый параметр всегда self и что бы использовать переменную из класса используй self

ф-я __init
class d
A=



ф-я _init_(self, ...) - конструктор

```
class dog
    A = 0
    def _init_(self, A1):
        ~~dog~~ self.A = A1

    def say(self)
        print(~~dog~~ self.A)

A = dog(9)
A.say()
```

Инкапсуляции - нет

~~private~~ __

ставим нижнее подчеркивание _ lol

в питоне ее нет

Наследование:

```
class dog:
    ......
class hea(dog): - наследован от dog
```

Полиморфизм - сам определи