

Разбор некоторых недочётов Code-style

student_src/01-D/student-seq_sum.py

```
1  n= int(input())
2  s=0
3  k=0
4  i=0
5  for i in range(n):
6      k=float(input())
7      s=k+s
8  print(s)
```

better-seq_sum.py

```
1  count = int(input())
2
3  seq_sum = 0. # !
4
5  for _ in range(count): # !
6      num = float(input())
7      seq_sum += num # !
8
9  print(seq_sum)
```

student_src/02-B/student-digit_sum-2.py student_src/02-B/student-digit_sum-1.py

```
1  x = int(input())
2  a = x // 100
3  b = (x // 10) % 10
4  c = x % 10
5  print(a + b + c)
```

```
1  N = input()
2  s = 0
3
4  for i in N:
5      i = int(i)
6      s = s + i
7
8  print(s)
```

student_src/02-B/student-digit_sum-1.py

```
1 N = input()
2 s = 0
3
4 for i in N:
5     i = int(i)
6     s = s + i
7
8 print(s)
```

student_src/02-B/student-digit_sum-2.py

```
1 x = int(input())
2 a = x // 100
3 b = (x // 10) % 10
4 c = x % 10
5 print(a + b + c)
```

better-digit_sum.py

```
1 number = int(input())
2
3 digit_sum = 0
4
5 while number > 0:
6     digit_sum += number % 10
7     number //= 10
8
9 print(digit_sum)
```

student_src/01-C/student-triangle_type.py

```
1  a = int(input())
2  b = int(input())
3  c = int(input())
4  if a >= (b + c) or b >= (a + c) or c >= (a + b) or a <= 0 or b <= 0 or c <= 0:
5      print('impossible')
6  elif a**2 == b**2 + c**2 or b**2 == a**2 + c**2 or c**2 == a**2 + b**2:
7      print('right')
8  elif b**2 + c**2 > a**2 and a**2 + c**2 > b**2 and a**2 + b**2 > c**2:
9      print('acute')
10 else:
11     print('obtuse')
```

better-triangle_type.py

```
1  a, b, c = int(input()), int(input()), int(input())
2
3  isexist = (a + b > c) and (b + c > a) and (a + c > b)
4  if not isexist:
5      print('impossible')
6  else:
7      # cos theorem
8      #  $c^2 = a^2 + b^2 - 2 a b \cos \text{angle}(a b)$ 
9      #  $\Rightarrow$ 
10     #  $\cos \text{angle}(a b) = -0.5 (c^2 - a^2 - b^2) / (a b)$ 
11
12     cos_ab_sign = - ( c**2 - a**2 - b**2 )
13     cos_bc_sign = - ( a**2 - b**2 - c**2 )
14     cos_ca_sign = - ( b**2 - c**2 - a**2 )
15
16     cos_mul_sign = cos_ab_sign * cos_bc_sign * cos_ca_sign
17
18     if cos_mul_sign == 0:
19         print('right')
20     elif cos_mul_sign > 0:
21         print('acute')
22     else: # cos_mul_sign < 0
23         print('obtuse')
```

student_src/01-E/student-units-1.py

```
1 a = int(input())
2 animal = input()
3 if animal == 'parrot':
4     if a // 10 == 0:
5         print(1)
6     else:
7         print(a // 10)
8 elif animal == 'monkey':
9     if a // 90 == 0:
10        print(1)
11    else:
12        print(a // 90)
13 else:
14     if a // 300 == 0:
15         print(1)
16     else:
17         print(a // 300)
```

better-units-1.py

```
1 python_len = int(input()) # !
2 animal_name = input() # !
3
4 if animal_name == 'parrot':
5     if python_len // 10 == 0:
6         print(1)
7     else:
8         print(python_len // 10)
9 elif animal_name == 'monkey':
10    if python_len // 90 == 0:
11        print(1)
12    else:
13        print(python_len // 90)
14 elif animal_name == 'elephant': # !
15    if python_len // 300 == 0:
16        print(1)
17    else:
18        print(python_len // 300)
19 else: # !
20     pass
```

better-units-2.py

better-units-1.py

```
1 python_len = int(input())          # !
2 animal_name = input()              # !
3
4 if animal_name == 'parrot':
5     if python_len // 10 == 0:
6         print(1)
7     else:
8         print(python_len // 10)
9 elif animal_name == 'monkey':
10    if python_len // 90 == 0:
11        print(1)
12    else:
13        print(python_len // 90)
14 elif animal_name == 'elephant':    # !
15    if python_len // 300 == 0:
16        print(1)
17    else:
18        print(python_len // 300)
19 else:                                # !
20    pass
```

```
1 python_len = int(input())
2 animal_name = input()
3
4 if animal_name == 'parrot':
5     eff_len = python_len // 10    # !
6
7     if eff_len == 0:              # !
8         eff_len = 1               # !
9
10    print(eff_len)                 # !
11
12 elif animal_name == 'monkey':
13     eff_len = python_len // 90
14
15     if eff_len == 0:
16         eff_len = 1
17
18    print(eff_len)
19
20 elif animal_name == 'elephant':
21     eff_len = python_len // 300
22
23     if eff_len == 0:
24         eff_len = 1
25
26    print(eff_len)
27
28 else:
29    pass
```


better-units-3.py

```
1  parrot_len = 10          # !
2  monkey_len = 90          # !
3  elephant_len = 300       # !
4
5  python_len = int(input())
6  animal_name = input()
7
8  if animal_name == 'parrot': animal_len = parrot_len
9  elif animal_name == 'monkey': animal_len = monkey_len
10 elif animal_name == 'elephant': animal_len = elephant_len
11 else:
12     # print('Undefined animal:', animal_name)          # UX    # !
13     # print('Supported animals: parrot, monkey, elephant') # UX    # !
14     exit(1)
15
16 eff_len = python_len // animal_len          # !
17
18 eff_len = 1 if eff_len == 0 else eff_len     # !
19
20 print(eff_len)
```

- Переменные
 - Не плодить лишние
 - Выбирать говорящие имена
- Унифицирование алгоритмов
 - Когда много накапливается частных случаев, скорее всего, что-то не так
 - DRY : Do no Repeat Yourself – когда встречаются повторения в коде, выделите их в отдельную сущность
- Триада чтение-обработка-вывод/запись
- Задача - модель - алгоритм – программа
- Новые синтаксические фичи
 - тернарный оператор
 - внутренняя переменная: _