

Практическое задание № 6

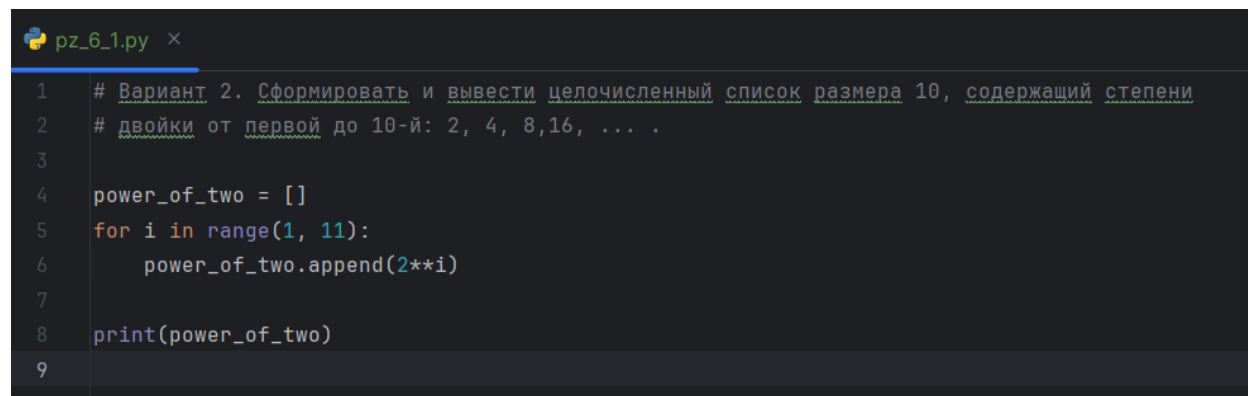
Тема: составление программ со списками в IDE PyCharm Community.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со списками в IDE PyCharm Community.

Постановка задачи:

Задача № 1: Создать программу, которая выводит целочисленный список размера 10, содержащий степени двойки от первой до 10-й: 2, 4, 8, 16,

Текст программы:



```
1 # Вариант 2. Сформировать и вывести целочисленный список размера 10, содержащий степени
2 # двойки от первой до 10-й: 2, 4, 8, 16, ... .
3
4 power_of_two = []
5 for i in range(1, 11):
6     power_of_two.append(2**i)
7
8 print(power_of_two)
9
```

Протокол программы:

[2, 4, 8, 16, 32, 64, 128, 256, 512, 1024]

Process finished with exit code 0

Тип алгоритма: циклический

Задача № 2: Создать программу, которая находит номера элементов списка размера N, которые больше своего левого соседа, и количество таких элементов. Найденные номера выводит в порядке их убывания.

Текст программы:

```
pz_6_1.py  pz_6_2.py x
1  # Вариант 2. Дан список размера N. Найти номера тех элементов списка, которые больше своего
2  # левого соседа, и количество таких элементов. Найденные номера выводить в
3  # порядке их убывания.
4
5  1 usage new *
6  def larger_elements(arr):
7      result = []
8      for i in range(1, len(arr)):
9          if arr[i] > arr[i-1]:
10             result.append(i)
11      result.reverse()
12      return result, len(result)
13
14 list = [1, 3, 2, 4, 5, 2, 7, 6]
15 print(list)
16 indices, count = larger_elements(list)
17 print("Найденные номера в порядке убывания:", indices)
18 print("Количество таких элементов:", count)
```

Протокол программы:

[1, 3, 2, 4, 5, 2, 7, 6]

Найденные номера в порядке убывания: [6, 4, 3, 1]

Количество таких элементов: 4

Process finished with exit code 0

Тип алгоритма: циклический, ветвление

Задача № 3: Создать программу, которая обнуляет элементы списка размера N, расположенных между его минимальным и максимальным элементами (не включая минимальный и максимальный элементы).

Текст программы:

```
pz_6_1.py  pz_6_2.py  pz_6_3.py x
1  # Вариант 2. Дан список размера N. Обнулить элементы списка, расположенные между его
2  # минимальным и максимальным элементами (не включая минимальный и
3  # максимальный элементы).
4
5  numbers = [3, 8, 2, 7, 1, 6, 4, 5]
6  print(numbers)
7
8  # Индексы минимального и максимального элементов
9  min_index = numbers.index(min(numbers))
10 max_index = numbers.index(max(numbers))
11
12 # Обнуление элементов между минимальным и максимальным элементами
13 if min_index < max_index:
14     for i in range(min_index + 1, max_index):
15         numbers[i] = 0
16 else:
17     for i in range(max_index + 1, min_index):
18         numbers[i] = 0
19
20 print(numbers)
```

Протокол программы:

[3, 8, 2, 7, 1, 6, 4, 5]

[3, 8, 0, 0, 1, 6, 4, 5]

Process finished with exit code 0

Тип алгоритма: циклический, ветвление

Вывод: В ходе работы закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрёл навыки составления программ со списками в IDE PyCharm Community.