

Практическое занятие №6

Тема: Составление программ со списками в IDE PyCharm Community.

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

Постановка задачи № 1:

```
#Дан список ненулевых значений размера N. Проверить, образуют ли его  
элементы геометрическую прогрессию  
#Если образуют, то вывести знаменатель прогрессии, если нет - 0
```

Тип алгоритма: Ветвление

Код алгоритма №1:

```
def function(a):  
    if len(a) < 2:  
        return False  
  
    b = a[1] / a[0]  
  
    for i in range(2, len(a)):  
        if a[i] / a[i-1] != b:  
            return 0  
        else: return b  
  
c = int(input("Введите размер массива"))  
  
list = []  
  
if c < 0:  
    print("Введите положительное число")  
else:  
    t = 0  
  
    while t < c:  
        list.append(random.randint(0, 100))  
        t += 1  
  
list.sort()  
  
print(list)  
  
d = function(list)  
  
if d:  
    print(f"Элементы образуют прогрессию с знаменателем {d}")  
else: print("Не образуют")
```

Вывод, если число соблюдает условие:

```
Введите размер массива10
[28, 31, 41, 43, 53, 62, 86, 94, 96, 99]
Не образуют
```

Вывод, если число не соблюдает условие:

```
Введите размер массива-5
Введите положительное число
```

Постановка задачи № 2:

```
#Дан целочисленный список А размера N. Переписать в новый целочисленный
# список В того же размера вначале все элементы исходного списка с четными
# номерами, а затем — с нечетными: A2, A4, A6, ..., A1, A3, A5, ... . Условный
# оператор не использовать.
```

Тип алгоритма: Линейный

Код алгоритма:

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
1 usage 2 DanilRumyantsev
def func(a):
    altln = a[::2]

    naltln = a[1::2]

    b = altln + naltln
    return b

d = func(a)

print(f"Сортированный массив {d}")
```

Вывод программы:

```
Сортированный массив [1, 3, 5, 7, 9, 2, 4, 6, 8, 10]
```

Постановка задачи № 2:

```
# Дано множество A из N точек (точки заданы своими координатами x, y). Найти
# пару
# различных точек этого множества с максимальным расстоянием между ними и
# само
# это расстояние (точки выводятся в том же порядке, в котором они
# перечислены при
# задании множества A).
```

Тип алгоритма: Ветвление

Код алгоритма:

```
def dist(a):
    n = len(a[0])
    max_dist = 0
    max_point = (0, 0)

    for i in range(n):
        for j in range(i + 1, n):
            x1, y1 = a[0][i], a[1][i]
            x2, y2 = a[0][j], a[1][j]

            dist = math.sqrt((x2 - x1)**2 + (y2 - y1)**2)

            if dist > max_dist:
                max_dist = dist
                max_point = ((x1, y1), (x2, y2))

    return max_point, max_dist

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

res_point, res_dist = dist(a)
print(f"Максимальное расстояние: {res_dist}")
print(f"Пара точек: {res_point}")
```

Вывод программы:

```
Максимальное расстояние: 1.4142135623730951
Пара точек: ((1, 5), (2, 6))
```

Вывод: Я закрепил знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со списками в IDE PyCharm Community.