

Практическое занятие №6

Тема: Составление программ ветвящейся структуры в IDE PyCharm Community

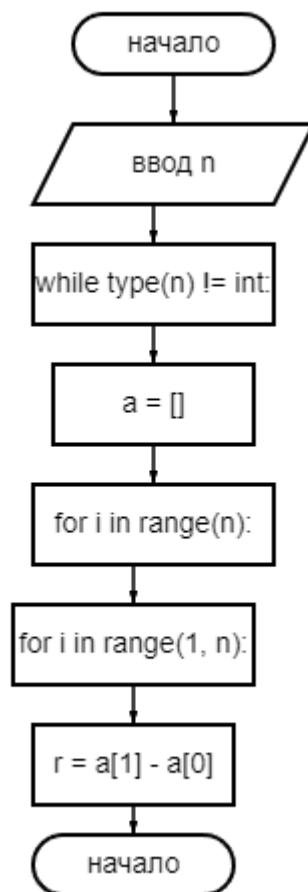
Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ ветвящейся структуры в IDE PyCharm Community.

Постановка задачи №1:

Дан целочисленный список размера N, не содержащий одинаковых чисел. Проверить, образуют ли его элементы арифметическую прогрессию. Если образуют, то вывести разность прогрессии, если нет - вывести 0.

Тип алгоритма: Линейный.

Блок-схема алгоритма:



Текст программы:

```
n = (input("введите N: "))
while type(n) != int:
    try:
        n = int(n)
    except ValueError:
        print("Неправильно ввели!")
```

```

n = (input("введите N: "))
a = []
for i in range(n):
    a.append(int(input("введите a[{}] : ".format(i+1))))

r = a[1] - a[0]
for i in range(1, n):
    if r != a[i] - a[i-1]:
        r = 0

print(r)

```

Протокол работы программы:

```

введите N: 3
введите a[1] : 1
введите a[2] : 2
введите a[3] : 3
1

```

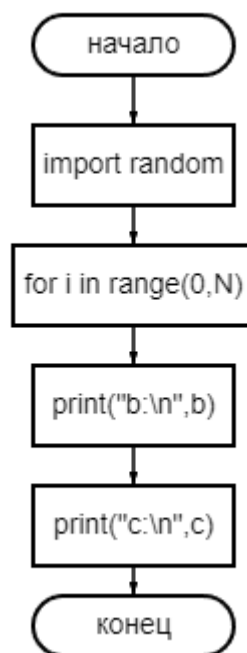
Process finished with exit code 0

Постановка задачи №2:

Дан список А размера N. Сформировать новый список В того же размера, элементы которого определяются следующим образом: $BK = 2 * AK$, если $AK \leq 5$ $AK/2$ в противном случае.

Тип алгоритма: Линейный.

Блок-схема алгоритма:



Текст программы:

```
def main():
    a = [[0.0] * 2 for _ in range(10)]
    n = (input("введите N: "))
    while type(n) != int:
        try:
            n = int(n)
        except ValueError:
            print("Неправильно ввели!")
            n = (input("введите N: "))
    for i in range(n):
        print(f"a[{i + 1}]:")
        a[i][0] = float(input("  x : "))
        a[i][1] = float(input("  y : "))

    amax = 0
    rmax = 0.0
    for i in range(n):
        if a[i][0] < 0 and a[i][1] > 0:
            r = math.sqrt(a[i][0] ** 2 + a[i][1] ** 2)
            if r > rmax or i == 0:
                rmax = r
                amax = i

    print(f"A {amax + 1} :")
    print(f"x: {a[amax][0]}")
    print(f"y: {a[amax][1]}")

if __name__ == "__main__":
    main()
```

Протокол работы программы:

введите N: 2

```
a[1]:
  x : 1
  y : 2
a[2]:
  x : 3
  y : 4
A 1 :
x: 1.0
y: 2.0
```

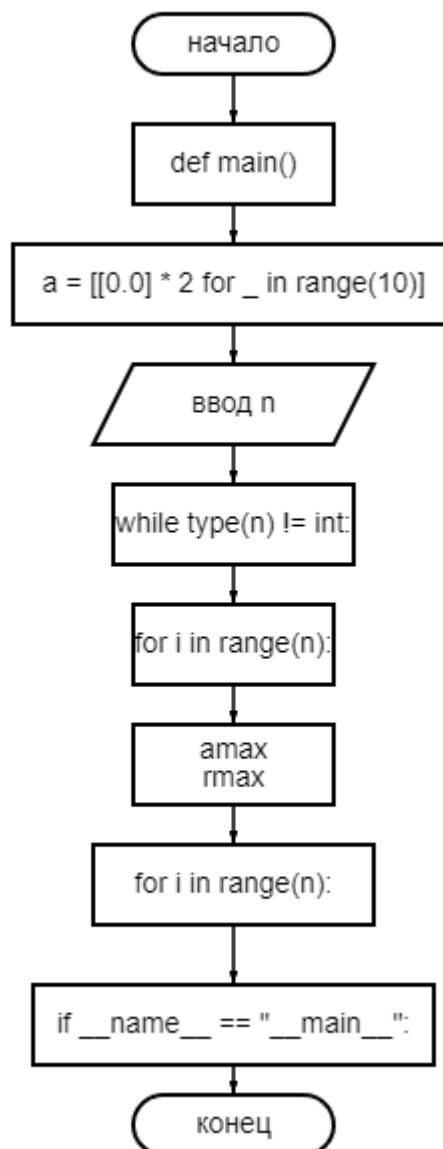
Process finished with exit code 0

Постановка задачи №3:

Дано множество A из N точек (точки заданы своими координатами x, y). Среди всех точек этого множества, лежащих в первой или третьей четверти, найти точку, наиболее близкую к началу координат. Если таких точек нет, то вывести точку с нулевыми координатами.

Тип алгоритма: Линейный.

Блок-схема алгоритма:



Текст программы:

```
def main():  
    a = [[0.0] * 2 for _ in range(10)]
```

```

n = (input("введите N: "))
while type(n) != int:
    try:
        n = int(n)
    except ValueError:
        print("Неправильно ввели!")
        n = (input("введите N: "))
for i in range(n):
    print(f"a[{i + 1}]:")
    a[i][0] = float(input("  x : "))
    a[i][1] = float(input("  y : "))

amax = 0
rmax = 0.0
for i in range(n):
    if a[i][0] < 0 and a[i][1] > 0:
        r = math.sqrt(a[i][0] ** 2 + a[i][1] ** 2)
        if r > rmax or i == 0:
            rmax = r
            amax = i

print(f"A {amax + 1} :")
print(f"x: {a[amax][0]}")
print(f"y: {a[amax][1]}")

if __name__ == "__main__":
    main()

```

Протокол работы программы:

введите N: 3

```

a[1]:
  x : 2
  y : 4
a[2]:
  x : 6
  y : 7
a[3]:
  x : 1
  y : 2
A 1 :
x: 2.0
y: 4.0

```

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработала навыки составления программ линейной структуры в IDE PyCharm Community.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.