

3 ЛИНЕЙНЫЙ АЛГОРИТМ

Алгоритм называется линейным, если он содержит N шагов, и все шаги выполняются последовательно друг за другом от начала до конца. Общий вид линейного алгоритма представлен на рисунке 24, где P_1, P_2, \dots, P_n - операторы.

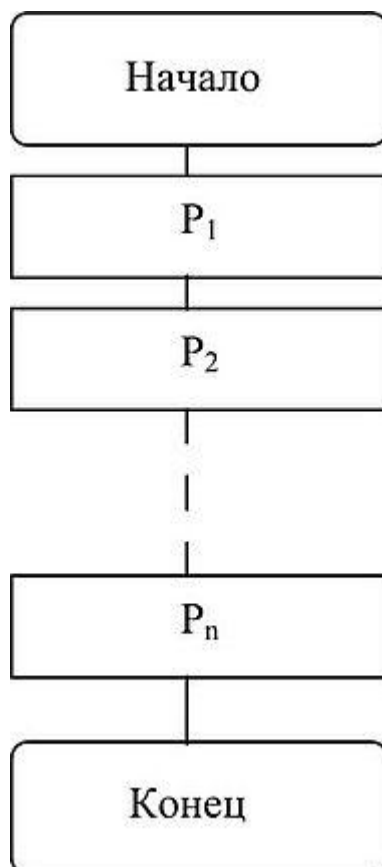


Рисунок 24 – Общий вид линейного алгоритма

Выполнение упражнений, представленных в лекциях, позволит вам лучше понять смысл работы того или иного оператора языка программирования Python. Однако перед тем как приступить к выполнению упражнений, мы еще раз напомним, что **присваивание в Python означает связывание значения с некоторым именем переменной**. Тем не менее, при объяснении материала, связанного с операторами присваивания, мы будем использовать более упрощенный вариант и выражаться в общем виде так: «**в ячейку N заносится число M** ».

3.1 Пояснение примеров заданий на линейный алгоритм

Задание 3.1. Какими будут значения переменных m и n после выполнения группы операторов?

```
m=25
n=m+1
m=m-25
print("Ответ= ", n)
print("Ответ= ", m)
```

Решение. Действие оператора присваивания заключается в занесении какого-либо значения в соответствующую ячейку памяти, причем старое значение, находящееся в ячейке, исчезает. Таким образом, если в ячейке **m** находится число 25, то в ячейку **n** после выполнения оператора присваивания **n=m+1** заносится число 26. Затем выполняется оператор **m=m-25**. Значение переменной **m** не изменялось, поэтому $25-25=0$, т. е. значение переменной **m=0**, а значение переменной **n=26**.

Задание 3.2. Каким будет значение переменной **n** после выполнения группы операторов?

```
m=20
n=10
m=m/n
n=m*n
n=n+30
print("Ответ= ", n)
```

Решение. После выполнения оператора присваивания **m=m/n** в ячейку **m** заносится число 2. Выполняется оператор **n=m*n**. В ячейке **m** находилось число 2, а в ячейке **n** - число 10, поэтому после выполнения операции умножения в ячейку **n** заносится число 20. Выполняется оператор **n=n+30**, и так как в ячейке **n** находилось значение 20, то значение переменной **n** будет равно 50.

Задание 3.3. Каким будет значение переменной **m** после выполнения группы операторов?

```
m=30
n=2
n=m/2
m=n
m=m+n
m=m
print("Ответ= ", m)
```

Решение. После выполнения оператора **n=m/2** в ячейку **n** заносится число 15. После выполнения оператора **m=n** в ячейке **m** будет также находиться число 15. Следовательно, число 30 заносится в ячейку **m** после

выполнения оператора **m=m+n**. Оператор **m=m** переприсваивает тоже самое значение переменной **m**. Ответ в задании: значение переменной **m** равно 30.

Задание 3.4. Какое число будет выведено в качестве ответа после выполнения группы операторов?

```
m=11
n=m*2
m=m+n
m=m
print("Ответ= ", m)
```

Решение. После выполнения оператора **n=m*2** в ячейку **n** заносится число 22, которое складывается с числом 11 после выполнения оператора **m=m+n**. Оператор **m=m** переприсваивает значение переменной **m**. Следовательно, на экран выводится число 33.

Задание 3.5. Какое значение будет находиться в ячейке **a** после выполнения группы операторов?

```
a=26
a="Папа"
b=a
b=54
b=False
a=b
print("Ответ= ", a)
```

Решение. Динамическая типизация, используемая в Python, позволяет связывать переменную с типом данных в момент присваивания значения. Следовательно, размещая в ячейке **a** сначала числовое значение (**a=26**), затем строковое (**a="Папа"**), мы не нарушаем никак правил, связанных с типами данных в Python. Поскольку последним оператором, который выполнится в данном фрагменте программы, будет оператор **a=b**, а в **b** находилось логическое значение **False**, соответственно, и в ячейке **a** будет размещено значение **False**.

Задание 3.5. Какое значение будет находиться в ячейке **d** после выполнения группы операторов?

```
a=3.333
b=3.332
c=min(a,b)
x=round(c)
d=pow(x,3)
print("Ответ= ", d)
```

Решение. Упражнение предполагает наличие знаний об основных математических функциях языка Python. Напомним, что функции **min()**, **round()** и **pow()** не требуют подключения модуля **math**. Функция **min()** с аргументами **a** и **b** вернет число 3.332 как меньшее из двух чисел. Результатом применения функции **round()** к аргументу **c** станет значение 3, поскольку будет возвращен результат округления до ближайшего меньшего целого значения для чисел с дробной частью меньше 0.5. Функция **pow()** возвращает результат возведения числа **x** в степень 3, следовательно, ответом станет число 27.

3.2 Примеры решения задач

Задача 3.1. Вычислите значения арифметических выражений и выведите на экран результаты вычислений.

Исходные данные: $x=1,4444$; $b=0,318$; $t=2,1$; $a=1,3$;

$$y = 9x^2 + \sin^2 x \cdot \sqrt{a-b};$$

$$z = \sqrt[3]{x^t} \left(ax^3 - \frac{x^2}{2!} \right).$$

Запишем арифметические выражения на языке программирования.

$$y=9*x*x+\sin(x)*\sin(x)*\text{sqrt}(a-b);$$

$$z=\exp(1/3*\log(\text{pow}(x,t)))*((a*x*x*x-(x*x)/(1*2))).$$

Разработка алгоритма решения задачи представлена на рисунке 25.

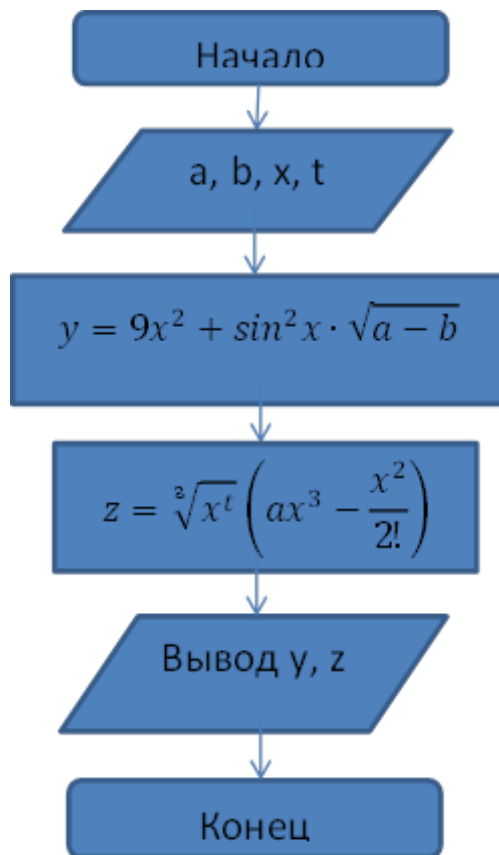


Рисунок 25 – Алгоритм решения задачи 3.1

Код программы, отвечающий за решение задачи, приведен в листинге ниже.

Мы ввели дополнительную переменную **z1** для вычисления значения **x**, возведенного в степень **t**, для более короткой записи в выражении, вычисляющем значение **z**.

Оператор **exit(0)** вызовет появление окна сообщения (рисунок 26), в котором спрашивается о том, хотите ли вы завершить запущенный на выполнение процесс (программу). В случае положительного ответа программа прекратит свою работу и, кроме того, закроется интерактивная сессия Python.

```
from math import *
a=float(input("Введите значение a="))
b=float(input("Введите значение b="))
x=float(input("Введите значение x="))
t=float(input("Введите значение t="))
y=9*x*x+ sin(x)*sin(x)*sqrt(a-b)
z1=log(pow(x,t))
z=pow(z1,1/3)*((a*x*x*x-(x*x)/(1*2)))
print("\n Значение y= ", y)
print("\n Значение z= ", z)
exit(0)
```

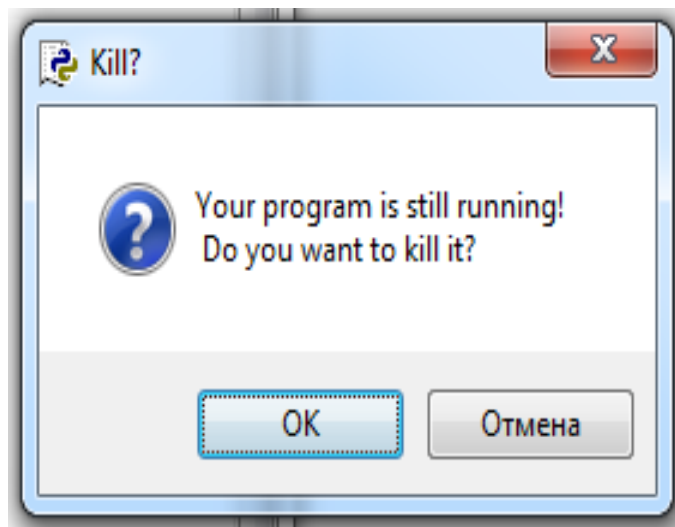
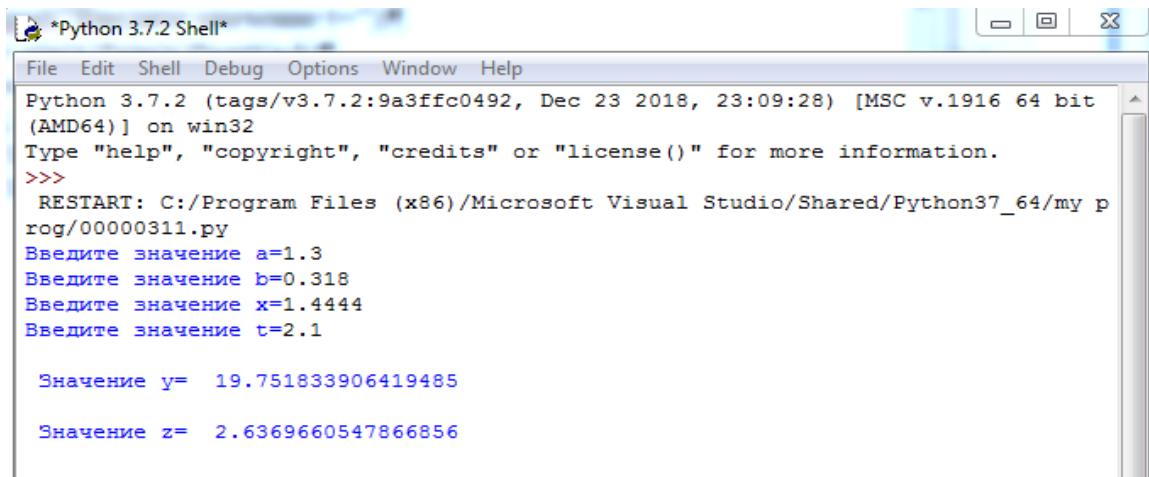


Рисунок 26 – Запрос подтверждение программы о ее остановке

Результат работы программы представлен на рисунке 27.



```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:/Program Files (x86)/Microsoft Visual Studio/Shared/Python37_64/my prog/00000311.py
Введите значение a=1.3
Введите значение b=0.318
Введите значение x=1.4444
Введите значение t=2.1

Значение y= 19.751833906419485

Значение z= 2.6369660547866856
```

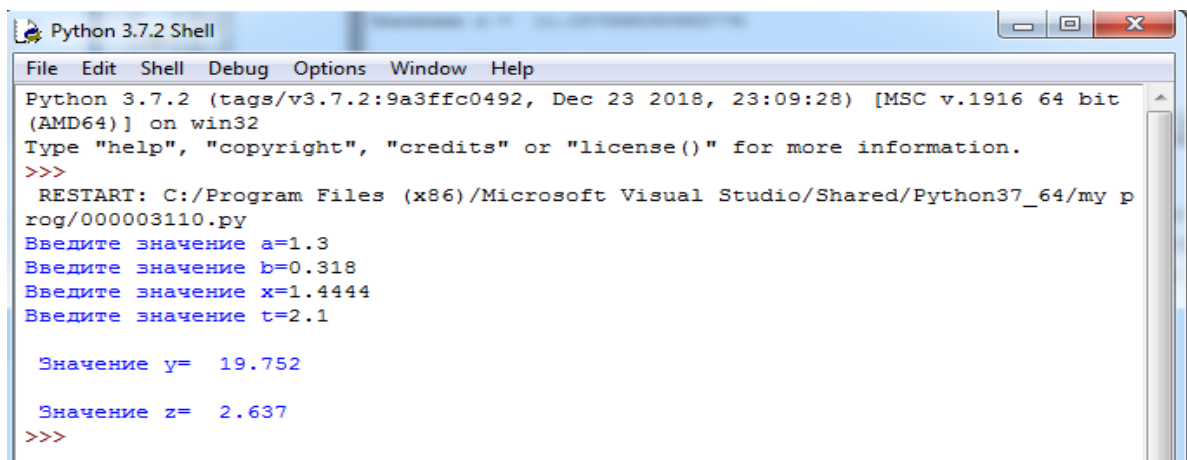
Рисунок 27 – Результат выполнения программы для задачи 3.1

Ограничить количество знаков после точки в вещественной записи числа поможет метод **format**. С самим понятием «метод» мы познакомимся позже, а сейчас посмотрим, как применить его к выводу данных для того, чтобы результат стал более читаемым:

```
print("\n Значение y= ", '{0:.3f}'.format(y))
print("\n Значение z= ", '{0:.3f}'.format(z))
```

Метод **format** записывается сразу после точки и применяется к результирующей ячейке, для которой мы хотим уменьшить количество знаков. В фигурных скобках указывается так называемый **спецификатор формата**. **0** - это поле, определяющее **индекс позиции**, при этом нумерация начинается с нуля (можно задать ширину поля, например **0:10**, тогда выводимое **число будет «отодвинуто» вправо на 10 позиций**), **.3** - **точность вычислений**, т. е. количество знаков после точки, **f** - **вещественное число в десятичном представлении**.

Результат работы программы с методом **format** представлен на рисунке 28.



```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:/Program Files (x86)/Microsoft Visual Studio/Shared/Python37_64/my prog/000003110.py
Введите значение a=1.3
Введите значение b=0.318
Введите значение x=1.4444
Введите значение t=2.1

Значение y= 19.752

Значение z= 2.637
>>>
```

Рисунок 28 – Результат выполнения программы для задачи 3.1 с использованием метода **format**

Задача 2. Найдите емкость конденсатора C , если известны: площадь пластин s и расстояние между ними l . Следует учесть, что в конденсатор вставлена металлическая пластина толщиной d , параллельная его обкладкам.

$$C = \frac{E_0 S}{L - d},$$

где C - емкость конденсатора;

E_0 - электрическая постоянная, равна $8,85 \cdot 10^{12}$.

Разработка алгоритма решения задачи представлена на рисунке 29.

Код программы, отвечающий за решение задачи, приведен ниже.

```
s=float(input("Введите площадь пластин S="))
l=float(input("Введите расстояние между пластинами l="))
d=float(input("Введите толщину пластины d="))
E0=8.85e-12
c=E0*s/(l-d)
print("\n Емкость конденсатора C=", '{0:10.12f}'.format(c))
print("\n Нажмите ENTER для выхода")
```

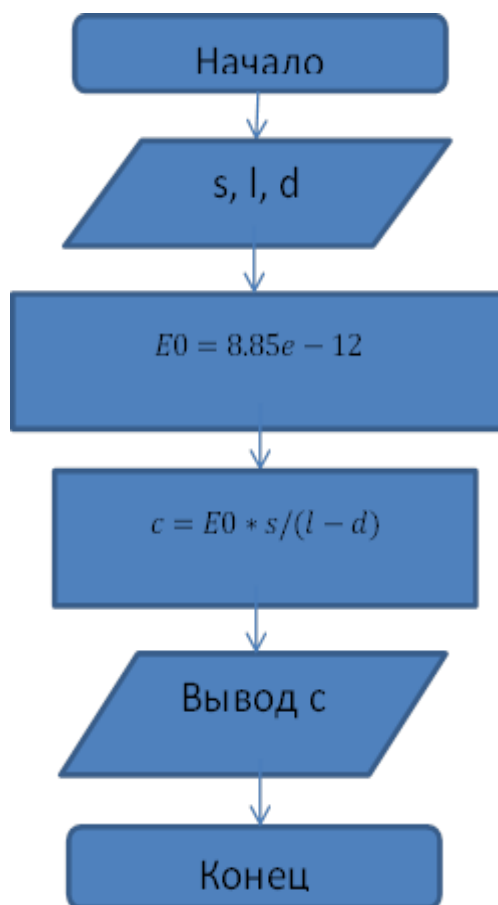


Рисунок 29 – Алгоритм решения задачи 3.2

3.3 Контрольные вопросы

1. Какой алгоритм называется линейным?

2. Нарисуйте общий вид линейного алгоритма.
3. С какой целью используется оператор **exit(O)** в программах, написанных на языке Python?
4. Поясните назначение метода **format** и приведите примеры его применения.

3.4 Задачи для самостоятельного решения

1. Разработайте алгоритм и программу, в которой вычисляются площадь и объем сферы.

$$S_{\text{сф}} = 4\pi R^2, V_{\text{сф}} = \frac{SR}{3}.$$

2. Разработайте алгоритм и программу, в которой подсчитывается количество часов минут и секунд в заданном числе суток.

$$Ch = 24 \cdot S, \text{ Minut} = 60 \cdot Ch, \text{ Sec} = 60 \cdot \text{Minut}.$$

3. Разработайте алгоритм и программу, в которой вычисляется площадь треугольника по трем сторонам. Вычисление проводится по формуле Герона.

$$S = \sqrt{p(p-a)(p-b)(p-c)},$$

где S - площадь треугольника; $p=(a+b+c)/2$ - полупериметр; a, b, c - длины сторон треугольника.

4. Разработайте алгоритм и программу решение системы линейных уравнений:

$$\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2' \end{cases}$$

по правилу Крамера:

$$X = \frac{c_1b_2 - c_2b_1}{a_1b_2 - a_2b_1}, \quad Y = \frac{a_1c_2 - a_2c_1}{a_1b_2 - a_2b_1}.$$

5. Разработайте алгоритм и программу определения объемов цилиндра и конуса с радиусом основания R=5 см и высотой H=8 см.

$$V_{\text{ц}} = h\pi R^2, V_{\text{к}} = \frac{1}{3}h\pi R^2.$$

6. Разработайте алгоритм и программу определения общего сопротивления электрической цепи, если имеются три резистора R₁, R₂, R₃.

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}; \quad R = \frac{R_1R_2R_3}{R_2R_3 + R_1R_3 + R_1R_2}.$$