

## 1. Код

```
def factorial(n):
    result = 1
    for i in range(1, n+1):
        result *= i;
    return result

def nth_sum(n, x):
    result = 0.
    for i in range(2, n+2):
        result += ((-1)**i) * (x**i) / factorial(i)
    return result

args = input('Enter n and x\n').split()
n = int(args[0])
x = float(args[1])

for i in range(1, n+1):
    print(i, nth_sum(i, x))
```

2.

Проверим работу программы посчитав отдельно первые три частичные суммы (я считал в вольфраме) и сравнив результаты.

Возьмем  $n=3$  и  $x=0.5$ .

$n=1$  (сумма от 2 до 2):  $(-1)^2 * (0.5^2) / 2! = 0.125$

$n=2$  (сумма от 2 до 3):  $(-1)^2 * (0.5^2) / 2! + (-1)^3 * (0.5^3) / 3! = 0.1041(6)$

$n=3$  (сумма от 2 до 4):  $(-1)^2 * (0.5^2) / 2! + (-1)^3 * (0.5^3) / 3! + (-1)^4 * (0.5^4) / 4! = 0.1067708(3)$

Ответ программы

```
Enter n and x
3 0.5
1 0.125
2 0.10416666666666667
3 0.10677083333333334

Process finished with exit code 0
```

Совпадает!

3.

Несколько скринов вывода

```
Enter n and x
5 0
1 0.0
2 0.0
3 0.0
4 0.0
5 0.0

Process finished with exit code 0
```

Нули т.к. в числителе есть x, который все обнуляет.

```
Enter n and x
9 0.001
1 5e-07
2 4.998333333333333e-07
3 4.998333749999999e-07
4 4.998333749916666e-07
5 4.99833374991668e-07
6 4.99833374991668e-07
7 4.99833374991668e-07
8 4.99833374991668e-07
9 4.99833374991668e-07

Process finished with exit code 0
```

Суммы одинаковые начиная со пятой, т.к. слагаемые становятся достаточно малы и теряются из-за ограничения на кол-во знаков в выводе.

## 4. Вопросы

1) Как объявляется переменная в языке Python? В каком месте мы можем её объявить?

В пайтоне динамическая типизация, т.е. указывать тип переменной не обязательно. При этом в пайтоне нет declaration, есть только defenition. Т.е. переменная создается в момент, когда ей присваивается значение. Например в с++ можно написать

```
int i;
```

```
int i = 0;
```

В пайтоне есть только второй вариант.

Объявлять переменные можно где угодно.

Примеры:

```
a = 10
```

```
s = 'some random string'
```

```
c, d = 1, 2
```

2) Как объявляется цикл в языке Python?

есть несколько вариантов.

```
for i in range(n):
```

```
    #something
```

Это аналог `for(int i = 0; i < n; ++i)` из с++.

```
while (condition):
```

```
    #something
```

В пайтоне также можно итерироваться и по спискам\строкам\итд, поэтому код вида

```
for i in list:
```

```
    #something
```

тоже будет работать. Он переберет все элементы списка по очереди.

3) Как осуществляется считывание информации из консоли и печать данных в консоль в языке Python?

Считывание – функция `input()`. Она возвращает строку, поэтому если нужно считать, например, число, то эта строка просто приводится к числу –

`int(input())`. Если надо считать несколько чисел, то сначала строка разбивается на части с помощью `split()`, а затем `int()` применяется к каждой из новых, коротких строк – `map(int, input().split())`

Вывод – функция `print()`. Имеет несколько необязательных параметров, вроде `sep` и `end`. См. <https://docs.python.org/3/library/functions.html#print>