

Hw3

Исследуем ряд $\exp(x)$

```
<x=1
eps=10**(-6)
s=x
t=x
i=1
while abs(t/s)>eps:
    t=(x/i)*t
    s=s+t
    i=i+1
import math
a=math.fabs(s-math.exp(x))/math.exp(x)

print ('s=',s, 'tmax=',t, 'a=',a, 'i=', i)>
```

```
hw2.py x
1 x=1
2 eps=10**(-2)
3 s=x
4 t=x
5 i=1
6 while abs(t/s)>eps:
7     t=(x/i)*t
8     s=s+t
9     i=i+1
10 import math
11 a=math.fabs(s-math.exp(x))/math.exp(x)
12 print ('s=',s, 'tmax=',t, 'a=',a, 'i=', i)
13
14
15
C:\windows\SYSTEM32\cmd.exe
s= 2.7166666666666663 tmax= 0.008333333333333333 a= 0.0005941848175817597 i= 6
-----
(program exited with code: 0)
Для продолжения нажмите любую клавишу . . .
```

sum	t max	ошибка	степен ь eps
2.7166666666666666 3	0.008333333333333333	0.000594184817581759 7	-2
2.7180555555555555 4	0.0013888888888888887	8.324114928800986e- 05	-3
2.718253968253968 4	0.0001984126984126983 9	1.125202597843745e- 06	-4

sum	t max	ошибка	степен ь eps
2.718281828446759 4	1.605904383682161e-10	2.220446049250313e- 16	-8
2.718281828459045 5	1.561920696858622e-16	1.6337129034990842e- 16	-16

- Из данной таблицы мы видим, что при значениях $x=1$ алгоритм сходится к 2.7182(e). Значение тем ближе к e , чем больше точность.
- Количество точных знаков меньше ожидаемого, так как мы привысили точность.
- После того, как наша точность превысила машинную, значение суммы ряда остается постоянным(при $x=1$ $s=2.7182818284590455$).

Используем "плохую" версию алгоритма.

```

import math
x=1
eps=10**(-5)
s=x
t=x
i=1
while abs(t/s)>eps:
    import math
    t=x**i/math.factorial(i)
    s=s+t
    i=i+1
    import math
a=math.fabs(s-math.exp(x))/math.exp(x)
print ('s=',s, 'tmax=',t, 'a=',a, 'i=', i)
>

```

```

1 import math
2 x=1
3 eps=10**(-5)
4 s=x
5 t=x
6 i=1
7 while abs(t/s)>eps:
8     import math
9     t=x**i/math.factorial(i)
10    s=s+t
11    i=i+1
12 import math
13 a=math.fabs(s-math.exp(x))/math.exp(x)
14 print ('s=',s, 'tmax=',t, 'a=',a, 'i=', i)
15
C:\windows\SYSTEM32\cmd.exe
s= 2.71827876984127 tmax= 2.48015873015873e-05 a= 1.125202597843745e-06 i= 9

-----
(program exited with code: 0)

Для продолжения нажмите любую клавишу . . .

```

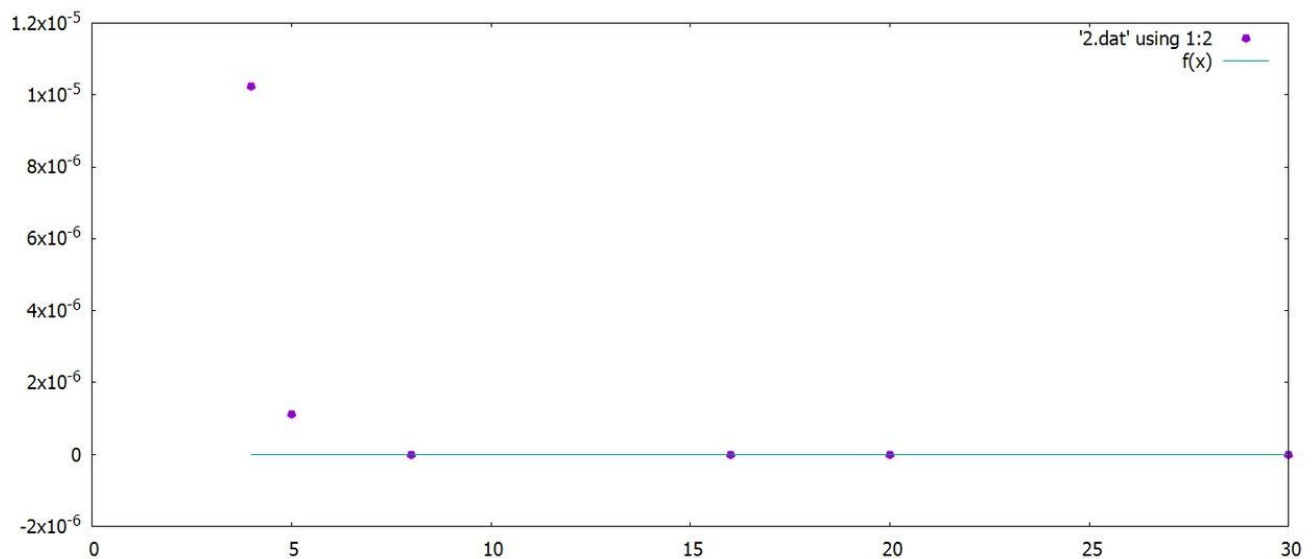
"Хороший"	"Плохой"
2.71827876984127	2.71827876984127
7.38905609893089	8.389046015712681
21.476989610413156	22.08553692318766

Используем встроенную функцию `math.exp(x)`. И сравним с суммой ряда при
дольшой точности

<code>math.exp(x)</code>	Сумма ряда
2.718281828459045	2.7182818284590455
7.38905609893065	8.389056098930649
20.085536923187668	21.476989610413156

- Из данной таблицы мы видим, что ряд не совсем точно сходится к истинному значению.

```
hw2.py x hw2.2.py x
1 x=10
2 eps=10**(-6)
3 s=x
4 t=x
5 i=1
6 import math
7 b=math.exp(x)
8 while abs(t/s)>eps:
9     t=(x/i)*t
10    s=s+t
11    i=i+1
12 import math
13 a=math.fabs(s-math.exp(x))/math.exp(x)
14
15 print ('s=',s, 'tmax=',t, 'a=',a, 'i=', i, 'b', b)
16
17
18 C:\windows\SYSTEM32\cmd.exe
s= 22035.46579480671 tmax= 5.391340061957266e-47 a= -40505.79908543065 i= 89 b 22026.465794806718
-----
(program exited with code: 0)
Для продолжения нажмите любую клавишу . . .
```



- При больших точностях функция ошибок от количества слагаемых выходит на константу.