

# Лабораторная работа №6

Научное программирование

---

Леонтьева К. А., НПМмд-02-23

03 октября 2023

Российский университет дружбы народов

Москва, Россия

Изучить в Octave методы расчета пределов, частичных сумм, суммы ряда, а также методы вычисления интегралов и аппроксимирования суммами

**Анонимная функция** - особый вид функций, которые объявляются в месте использования и не получают уникального идентификатора для доступа к ним. Обычно при создании анонимные функции либо вызываются напрямую, либо ссылка на функцию присваивается переменной, с помощью которой затем можно косвенно вызывать данную функцию.

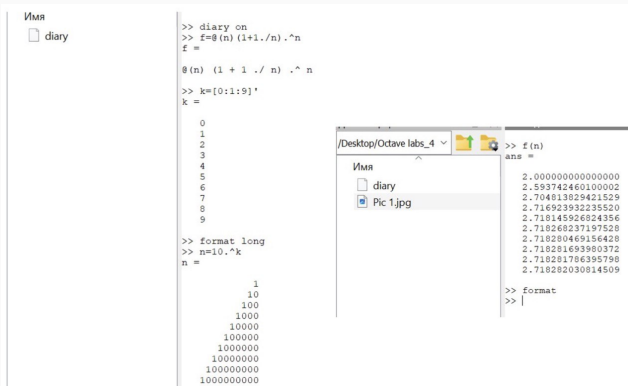
В Octave анонимные функции определяются с помощью синтаксиса

`@(argument-list) expression.`

Любые переменные, которые не найдены в списке аргументов, наследованы от объема включения. Анонимные функции полезны для создания простых функций без имени от выражений или для обертывания вызовов к другим функциям для адаптации их к использованию функциями как `quad`, которая применяется при вычислении интегралов.

## Ход выполнения лабораторной работы

- Оценили предел:  $\lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n$
- Полученный результат близок к теоретическому значению предела -  $e$



```
>> diary on
>> f=@(n) (1+1./n).^n
f =

@(n) (1 + 1 ./ n) .^ n

>> k=[0:1:9]'
k =

0
1
2
3
4
5
6
7
8
9

>> format long
>> n=10.^k
n =

1
10
100
1000
10000
100000
1000000
10000000
100000000
1000000000

>> f(n)
ans =

2.000000000000000
2.593742460100002
2.704813829421529
2.716923932235520
2.718145926824356
2.718268237197528
2.718280469156428
2.718281693980372
2.718281786395798
2.718282030814509

>> format
>> |
```

Figure 1: Рис.1: Оценка выражения под знаком предела

## Ход выполнения лабораторной работы

- Пусть  $\sum_{n=2}^{\infty} a_n$  - ряд,  $n$ -й член равен  $a_n = \frac{1}{n(n+2)}$ . Построили слагаемые и частичные суммы для  $2 \leq n \leq 11$  на графике

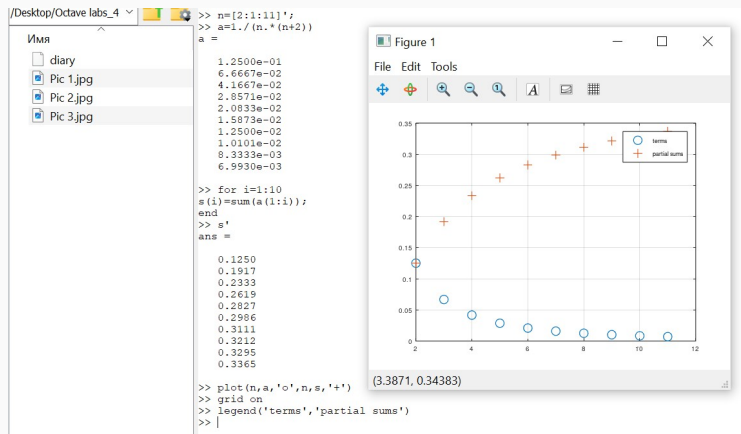


Figure 2: Рис.2: Частичные суммы

- Нашли сумму первых 1000 членов гармонического ряда:  $\sum_{n=1}^{1000} \frac{1}{n}$

```
>> n=[1:1:1000];  
>> a=1./n;  
>> sum(a)  
ans = 7.4855  
>> |
```

Figure 3: Рис.3: Сумма ряда

- Вычислили интеграл:  $\int_0^{\frac{\pi}{2}} e^{x^2} \cos(x) dx$

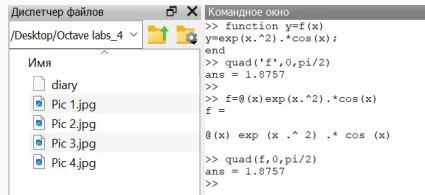
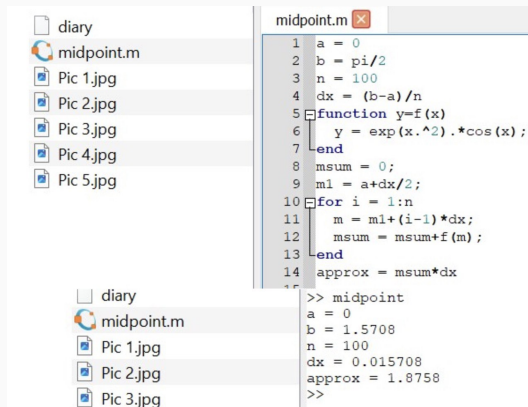


Figure 4: Рис.4: Вычисление интеграла

## Ход выполнения лабораторной работы

- Вычислили указанный ранее интеграл по правилу средней точки для  $n=100$  (классический код)

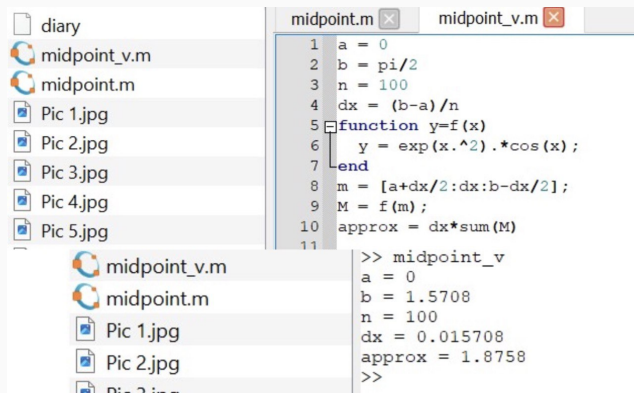


```
midpoint.m
1 a = 0
2 b = pi/2
3 n = 100
4 dx = (b-a)/n
5 function y=f(x)
6     y = exp(x.^2).*cos(x);
7 end
8 msum = 0;
9 m1 = a+dx/2;
10 for i = 1:n
11     m = m1+(i-1)*dx;
12     msum = msum+f(m);
13 end
14 approx = msum*dx
15
>> midpoint
a = 0
b = 1.5708
n = 100
dx = 0.015708
approx = 1.8758
>>
```

Figure 5: Рис.5: Аппроксимирование суммами

## Ход выполнения лабораторной работы

- Вычислили указанный ранее интеграл по правилу средней точки для  $n=100$  (векторизованный код)



```
diary
midpoint_v.m
midpoint.m
Pic 1.jpg
Pic 2.jpg
Pic 3.jpg
Pic 4.jpg
Pic 5.jpg
midpoint_v.m
midpoint.m
Pic 1.jpg
Pic 2.jpg
Pic 3.jpg

midpoint.m
1 a = 0
2 b = pi/2
3 n = 100
4 dx = (b-a)/n
5 function y=f(x)
6     y = exp(x.^2).*cos(x);
7 end
8 m = [a+dx/2:dx:b-dx/2];
9 M = f(m);
10 approx = dx*sum(M)
11

midpoint_v.m
>> midpoint_v
a = 0
b = 1.5708
n = 100
dx = 0.015708
approx = 1.8758
>>
```

Figure 6: Рис.7: Аппроксимирование суммами - векторизованный код



- Сравнивали результаты и время выполнения программ










Имя	
 diary	<code>&gt;&gt; tic; midpoint; toc</code>
 midpoint_v.m	<code>a = 0</code>
 midpoint.m	<code>b = 1.5708</code>
 Pic 1.jpg	<code>n = 100</code>
 Pic 2.jpg	<code>dx = 0.015708</code>
 Pic 3.jpg	<code>approx = 1.8758</code>
 Pic 4.jpg	<code>Elapsed time is 0.00568295 seconds.</code>
 Pic 5.jpg	<code>&gt;&gt; tic; midpoint_v; toc</code>
 Pic 6.jpg	<code>a = 0</code>
	<code>b = 1.5708</code>
	<code>n = 100</code>
	<code>dx = 0.015708</code>
	<code>approx = 1.8758</code>
	<code>Elapsed time is 0.00294685 seconds.</code>
	<code>&gt;&gt;</code>

Figure 7: Рис.8: Сравнение кодов

- В ходе выполнения данной лабораторной работы я изучила в Octave методы расчета пределов, частичных сумм, суммы ряда, а также методы вычисления интегралов и аппроксимирования суммами