

# Лабораторная работа №6

Научное программирование

---

Полиенко Анастасия Николаевна

19 сентября 2023

Российский университет дружбы народов, Москва, Россия

НПМмд-02-23

# **Пределы, последовательности и ряды и Численное интегрирование**

---

## Цель лабораторной работы

Изучить работу с пределами, последовательностями и рядами и численным интегрированием в GNU Octave.

## Задачи лабораторной работы

1. Изучить работу с пределами, последовательностями и рядами
2. Изучить численное интегрирование

## **Ход лабораторной работы**

---

# Предел функции

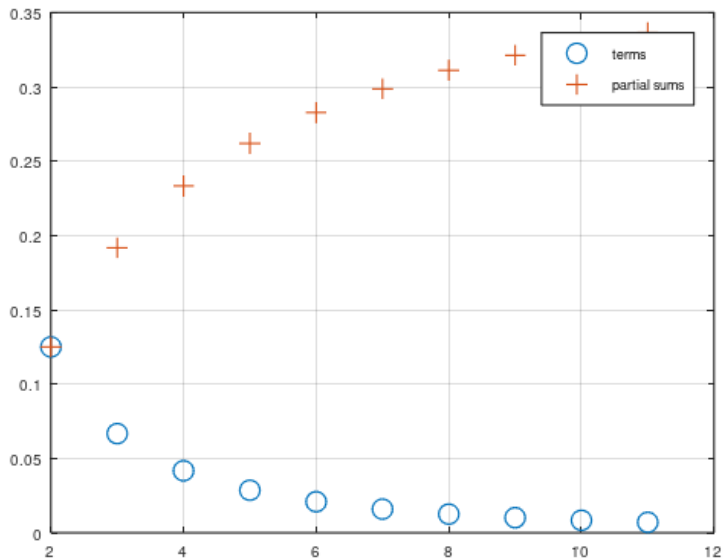
Посчитаем значения функции  $f(n)$ . Можем наблюдать, как они сходятся к числу  $e$ .

```
>> f(n)
ans =

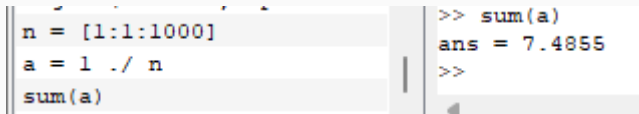
2.0000000000000000
2.593742460100002
2.704813829421529
2.716923932235520
2.718145926824356
2.718268237197528
2.718280469156428
2.718281693980372
2.718281786395798
2.718282030814509
```

Рис. 1: Значения функции

## Частичные суммы



Создаём индексную переменную, генерируем на основе неё вектор значений и считаем их сумму.

A screenshot of a MATLAB script editor and its command window. The script editor on the left contains three lines of code: `n = [1:1:1000]`, `a = 1 ./ n`, and `sum(a)`. The command window on the right shows the execution results: `>> sum(a)` followed by `ans = 7.4855` and a prompt `>>`.

```
n = [1:1:1000]
a = 1 ./ n
sum(a)
```

```
>> sum(a)
ans = 7.4855
>>
```

**Рис. 3:** Сумма ряда



# Численное интегрирование

Создадим функцию и вычислим определённый интеграл с помощью встроенной команды `quad`.

```
>> function y = f(x)
y = exp(x.^2) .* cos(x)
end
>> quad('f',0,pi/2)
y = 1.3103
y = 1.0002
y = 0.2267
y = 1.0056
y = 0.9042
y = 1.0319
y = 1.4191
y = 1.1003
y = 1.5288
y = 1.2269
y = 1.3991
y = 1.0000
y = 0.039792
y = 1.0015
y = 0.5458
y = 1.0149
y = 1.2115
```

# Аппроксимация суммами

Напишем код для аппроксимирования суммами с циклами и векторами и сравним два алгоритма по времени

```
>> tic; midpoint; toc
m1 = 7.8540e-03
approx = 1.8759
Elapsed time is 0.00720716 seconds.
>> tic; midpoint_v; toc
M =

Columns 1 through 9:

    1.000031    1.000278    1.000771    1.001512    1.002499    1.003734    1.005217    1.006947    1.008926

Columns 10 through 18:

    1.011154    1.013632    1.016359    1.019336    1.022564    1.026045    1.029777    1.033762    1.038001

Columns 19 through 27:

    1.042493    1.047241    1.052244    1.057502    1.063017    1.068788    1.074816    1.081101    1.087643

Columns 28 through 36:

    1.094441    1.101496    1.108807    1.116374    1.124194    1.132267    1.140591    1.149165    1.157985

Columns 37 through 45:

    1.167048    1.176352    1.185891    1.195662    1.205659    1.215876    1.226306    1.236941    1.247771

Columns 46 through 54:

    1.258788    1.269579    1.281332    1.292833    1.304466    1.316214    1.328057    1.339973    1.351940

Columns 55 through 63:

    1.363930    1.375916    1.387864    1.399740    1.411505    1.423117    1.434529    1.445691    1.456547

Columns 64 through 72:

    1.467034    1.477087    1.486632    1.495689    1.503870    1.511380    1.518015    1.523660    1.528192

Columns 73 through 81:

    1.531474    1.533360    1.533688    1.532283    1.528963    1.523491    1.515670    1.505243    1.491941

Columns 82 through 90:

    1.475473    1.455519    1.431733    1.403739    1.371125    1.333447    1.290218    1.240910    1.184949
```

Научилась работе с системами линейных алгебраических уравнений в Octave.