ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

Пределы, последовательности и ряды

Аминов Зулфикор Мирзокаримович

Содержание

[Ход работы: 1](#_Toc83487544)

[Пределы, последовательности и ряды 1](#_Toc83487545)

[Частичные суммы 2](#_Toc83487546)

[Сумма ряда 5](#_Toc83487547)

[Частичные интегрирование 6](#_Toc83487548)

[Вычисление интегралов 6](#_Toc83487549)

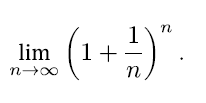
[Аппроксимирование суммами 6](#_Toc83487550)

[Вывод 9](#_Toc83487551)

# Ход работы:

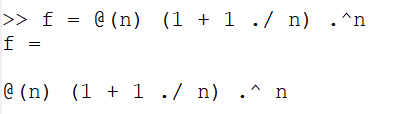
# Пределы, последовательности и ряды

Рассмотрим предел:



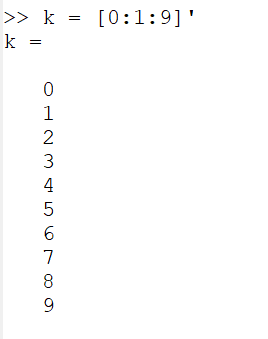
рисунка 1

Оценим это выражение с методом анонимной функцией. Это хороший способ быстро определить простую функйию.



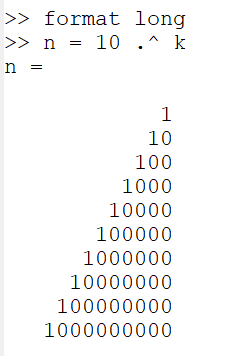
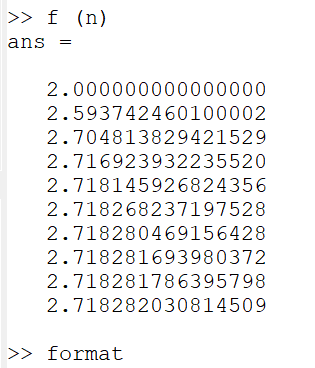
рисунка 2

Далее мы создаём индексную переменную, состоящую из целых чисел от 0 до 9:



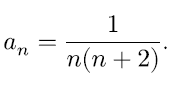
рисунка 3

Теперь мы возьмём степени 10, которые будут входными значениями, а затем оценим f(n).

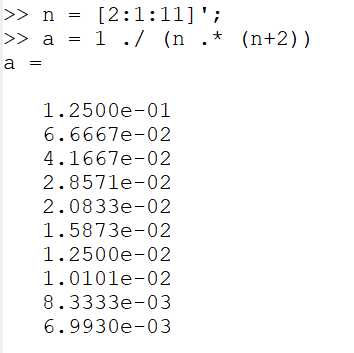
# Частичные суммы

Пусть рисунка 6 -ряд, n-й член равен



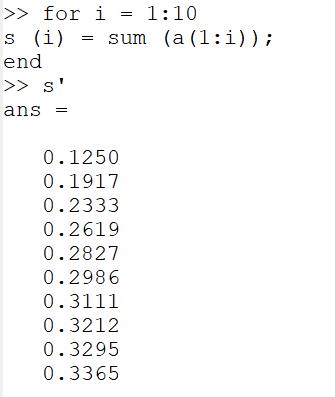
рисунка 7

Для этого мы определим индксный вектор n от 2 до 11, а затем вычислим члены.



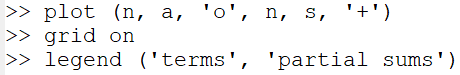
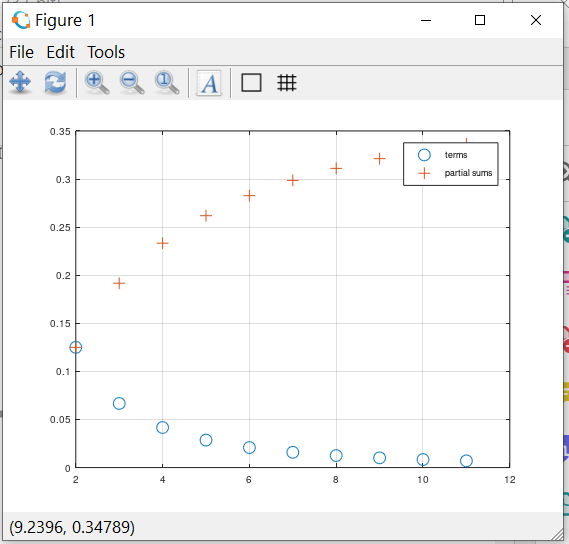
рисунка 8

Если мы хотим частичную сумму, нам нужно только написать sum(a). Если мы хотим получить последовательность частичных сумм, нам нужно использовать цикл. Мы будем использоватьцикл for с индексом i от 10. Для каждого i мы получим частичную сумму последовательность a\_n от первого слагаемого до i-го слагаемого. На выходе получается 10-элементный вектор этих частичных сумм.



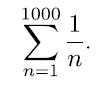
рисунка 9

Наконец, мы построим слагаемые и частичные суммы для 2<= n <= 11.

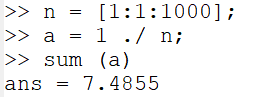
# Сумма ряда

Найдем сумму первых 1000 членов гармонического ряда:



рисунка 12

Нам нужно только сгенировать члены как ряда вектор, а затем взять их сумму.

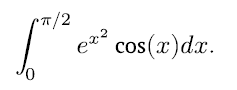


рисунка 13

# Частичные интегрирование

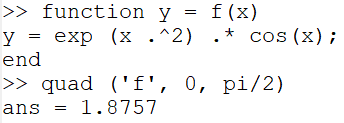
# Вычисление интегралов

С помощью команду quar вычислим интеграл:



рисунка 14

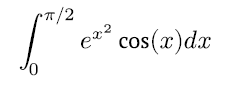
Синтаксис команды -quar(‘f’, a, b). Нам нужно сначала определить функйию.



рисунка 15

# Аппроксимирование суммами

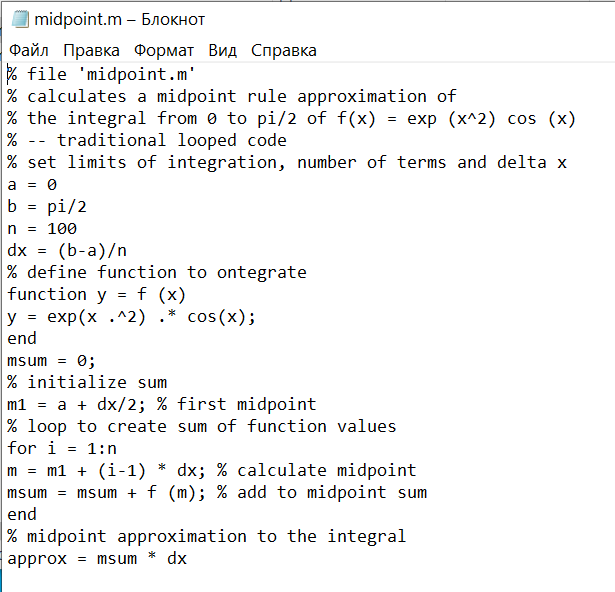
Напишем скрипт, чтобы вычислить интеграл



рисунка 16

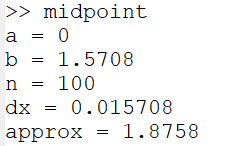
по правилу средней точки для n = 100.

Введим код в текстовом файле и назавем его midpoint.m.



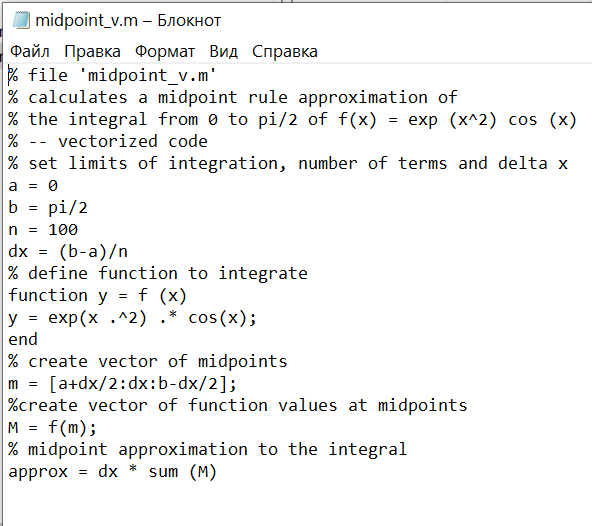
рисунка 17

Набрав midpoint в командной строке запустим скрипт



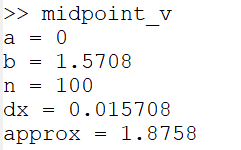
рисунка 18

Создадим вектор x-координат средних точек. Затем мы оцениваем f по этому вектору средней точки, чтобы получить вектор значений функции. Аппроксимация средней точки -это сумма компонент вектора, умноженная на делтаX.



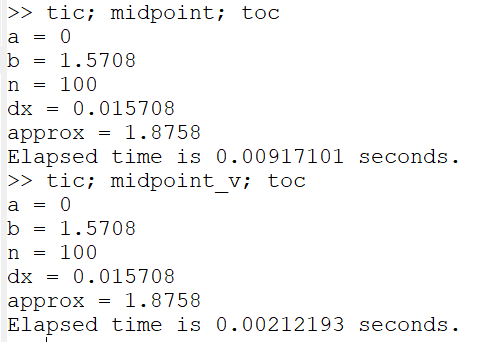
рисунка 19

Запустим его.



рисунка 20

Сравниваем результаты и сравниваем время выполнения для каждой реализации.



рисунка 21

# Вывод

Научился работать в Octave с пределамы, последовательностами и с рядамы.