**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,   
МЕХАНИКИ И ОПТИКИ»**

**Факультет Программной Инженерии и Компьютерной Техники**

**Дисциплина:**

**«*Вычислительная математика*»**

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5  
*«Интерполяция функции»***

***Вариант 3***

**Выполнил:**

Студент гр. P32151 *Горинов Даниил Андреевич*

**Проверил:**

*Машина Екатерина Алексеевна*

Санкт-Петербург

2023г.

**Цель лабораторной работы:**

Решить задачу интерполяции, найти значения функции при заданных значениях аргумента, отличных от узловых точек

**Порядок выполнения лабораторной работы:**

***Обязательное задание (до 80 баллов)***

*Вычислительная реализация задачи:*

1. Выбрать из табл. 1 заданную по варианту таблицу (таблица 1.1 – таблица 1.5);
2. Построить таблицу конечных разностей для заданной таблицы. Таблицу отразить в отчете;
3. Вычислить значения функции для аргумента (см. табл.1), используя первую или вторую интерполяционную формулу Ньютона. Обратить внимание какой конкретно формулой необходимо воспользоваться;
4. Вычислить значения функции для аргумента (см. табл. 1), используя первую или вторую интерполяционную формулу Гаусса. Обратить внимание какой конкретно формулой необходимо воспользоваться;
5. Подробные вычисления привести в отчете.

*Программная реализация задачи:*

1. Исходные данные задаются тремя способами:
2. в виде набора данных (таблицы x, y), пользователь вводит значения с клавиатуры;
3. в виде сформированных в файле данных (подготовить не менее трех тестовых вариантов);
4. на основе выбранной функции, из тех, которые предлагает программа, например, . Пользователь выбирает уравнение, исследуемый интервал и количество точек на интервале (не менее двух функций).
5. Сформировать и вывести таблицу конечных разностей;
6. Вычислить приближенное значение функции для заданного значения аргумента, введенного с клавиатуры, указанными методами (см. табл. 5.2). Сравнить полученные значения;
7. Построить графики заданной функции с отмеченными узлами интерполяции и интерполяционного многочлена Ньютона/Гаусса (разными цветами);
8. Программа должна быть протестирована на различных наборах данных, в том числе и некорректных.
9. Проанализировать результаты работы программы.

***Необязательное задание (до 20 баллов)***

1. Реализовать в программе вычисление значения функции для заданного значения аргумента, введенного с клавиатуры, используя схемы Стирлинга;
2. Реализовать в программе вычисление значения функции для заданного значения аргумента, введенного с клавиатуры, используя схемы Бесселя.

**Рабочие формулы методов:**

*Формула для полинома Лагранжа:*

*Формула Ньютона для не равностоящих узлов:*

,где – разделенные разности.

*Вторая интерполяционная формула Ньютона для интерполирования назад всех равностоящих узлов:*

, где и

*Формула интерполяции полинома Гаусса для :*

Изображение выглядит как текст, рукописный текст, Шрифт, линия

Автоматически созданное описание

Формула интерполяционного полинома Стирлинга:

Изображение выглядит как текст, Шрифт, рукописный текст, линия

Автоматически созданное описание

, где и

Формула интерполяционного полинома Бесселя:

Изображение выглядит как текст, Шрифт, линия, рукописный текст

Автоматически созданное описание

, где и

Вычисление значений функции

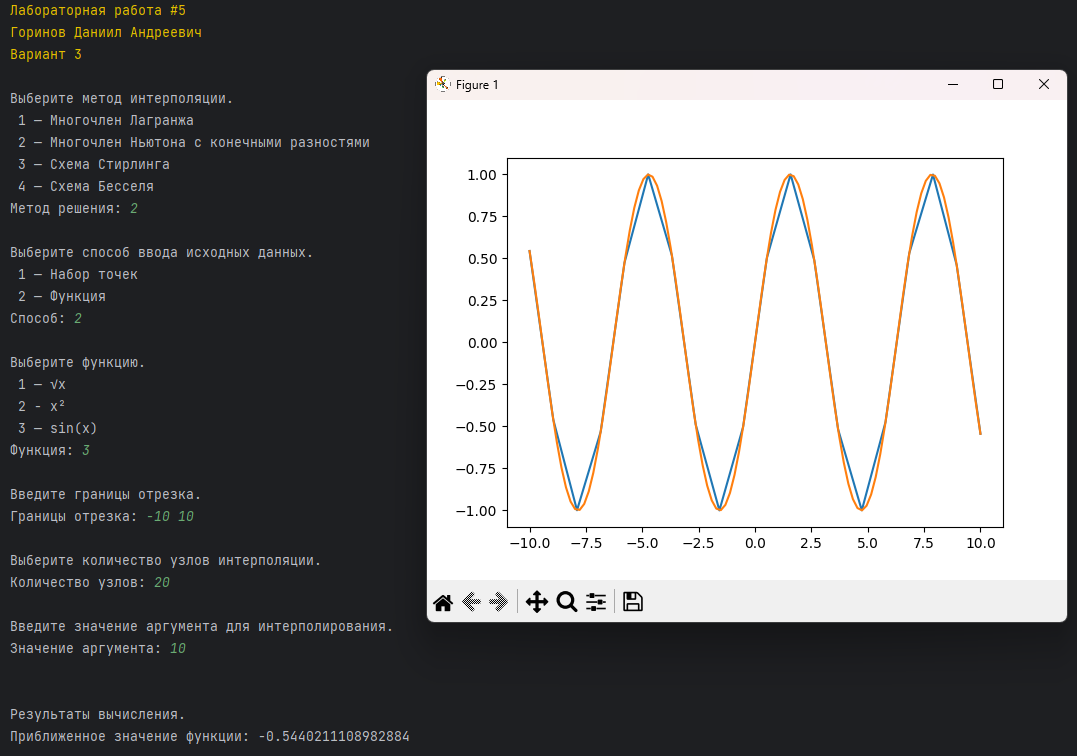
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i |  |  |  |  |  |  |  |  |
| 0 | 1,1 | 0,2234 | 1,0204 | 0,0002 | 0,0132 | -0,0268 | 0,0262 | 0,0187 |
| 1 | 1,25 | 1,2438 | 1,0206 | 0,0134 | -0,0136 | -0,0006 | 0,0449 |  |
| 2 | 1,4 | 2,2644 | 1,034 | -0,0002 | -0,0142 | 0,0443 |  |  |
| 3 | 1,55 | 3,2984 | 1,0338 | -0,0144 | 0,0301 |  |  |  |
| 4 | 1,7 | 4,3322 | 1,0194 | 0,0157 |  |  |  |  |
| 5 | 1,85 | 5,3516 | 1,0351 |  |  |  |  |  |
| 6 | 2 | 6,3867 |  |  |  |  |  |  |

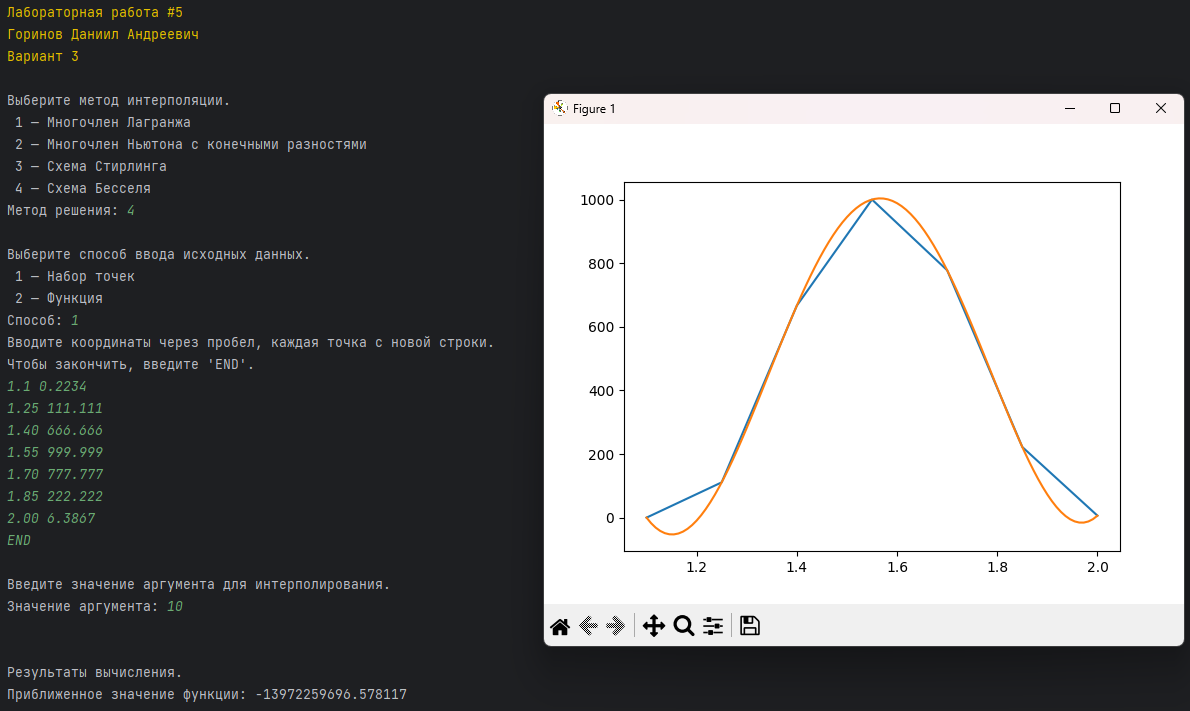
*Используем первую формулу Ньютона:*

*Используем вторую формулу Гаусса:*

Листинг программы:

1. def lagrange\_polynomial(dots, x):
2. result = 0
3. n = len(dots)
4. for i in range(n):
5. c1 = c2 = 1
6. for j in range(n):
7. if i != j:
8. c1 \*= x - dots[j][0]
9. c2 \*= dots[i][0] - dots[j][0]
10. result += dots[i][1] \* c1 / c2
11. return result
12. def newton\_polynomial(dots, x):
13. y\_arr = [i[1] for i in dots]
14. x\_arr = [i[0] for i in dots]
15. diff = [y\_arr]
16. for i in range(len(x\_arr)):
17. tmp\_dif = []
18. for j in range(len(x\_arr) - i - 1):
19. tmp\_dif.append((diff[-1][j + 1] - diff[-1][j]) / (x\_arr[j + i + 1] - x\_arr[j]))
20. diff.append(tmp\_dif)
21. mul = 1
22. answer = y\_arr[0]
23. for i in range(len(x\_arr) - 1):
24. mul \*= (x - x\_arr[i])
25. answer += diff[i + 1][0] \* mul
26. return answer
27. def sterling\_scheme(dots, x):
28. n = len(dots)
29. def divided\_differences(dots):
30. n = len(dots)
31. divided\_diff = [[y for \_, y in dots]]
32. for i in range(1, n):
33. diff = [(divided\_diff[i - 1][j + 1] - divided\_diff[i - 1][j]) / (dots[j + i][0] - dots[j][0]) for j in
34. range(n - i)]
35. divided\_diff.append(diff)
36. return divided\_diff
37. def sterling\_polynomial(dots, divided\_diff, x):
38. n = len(dots)
39. y = divided\_diff[0][0]
40. p = 1
41. for i in range(1, n):
42. p \*= (x - dots[i - 1][0])
43. y += divided\_diff[i][0] \* p
44. return y
45. divided\_diff = divided\_differences(dots)
46. result = sterling\_polynomial(dots, divided\_diff, x)
47. return result
48. def bessel\_scheme(dots, x):
49. n = len(dots)
50. result = 0.0
51. for i in range(n):
52. term = dots[i][1]
53. for j in range(n):
54. if i != j:
55. term \*= (x - dots[j][0]) / (dots[i][0] - dots[j][0])
56. result += term
57. return result

Результаты выполнения программы:

Вывод:

Интерполяция функций представляет собой значительно более сложный процесс по сравнению с аппроксимацией. Кроме того, выражения, используемые для получения интерполяционных полиномов, обычно являются достаточно сложными для визуального восприятия и понимания. Непонятно, почему существует так много способов построения полинома для интерполяции, хотя итоговый полином должен быть единственным. Тем не менее, поставленная задача была успешно решена.