«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет программной инженерии и компьютерной техники  
Направление подготовки 09.03.04 Программная инженерия  
Дисциплина «Вычислительная математика»

Отчет

По лабораторной работе №5

Вариант 1

Студент:

Алхимовици А.

Р3210

Преподаватель:

Наумова Н. А.

Санкт-Петербург, 2025 г.

**Цель лабораторной работы**:

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

# **1 Вычислительная реализация задачи:**

1. Выбрать таблицу 𝑦 = 𝑓(𝑥):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблица 1.1 | X | y | вариант | X1 | X2 |
| 0.25 | 1.2557 | 1 | 0.251 | 0.402 |
| 0.30 | 2.1764 |
| 0.35 | 3.1218 |
| 0.40 | 4.0482 |
| 0.45 | 5.9875 |
| 0.50 | 6.9195 |
| 0.55 | 7.8359 |

2. Построить таблицу конечных разностей:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | **x** | **y** | **Δ1** | **Δ2** | **Δ3** | **Δ4** | **Δ5** | **Δ6** |
| 0 | 0.25 | 1.2557 | 0.9207 | 0.0247 | -0.0437 | 1.0756 | -4.1277 | 10.1917 |
| 1 | 0.30 | 2.1764 | 0.9454 | -0.0190 | 1.0319 | -3.0521 | 6.0640 |  |
| 2 | 0.35 | 3.1218 | 0.9264 | 1.0129 | -2.0202 | 3.0119 |  |  |
| 3 | 0.40 | 4.0482 | 1.9393 | -1.0073 | 0.9917 |  |  |  |
| 4 | 0.45 | 5.9875 | 0.9320 | -0.0156 |  |  |  |  |
| 5 | 0.50 | 6.9195 | 0.9164 |  |  |  |  |  |
| 6 | 0.55 | 7.8359 |  |  |  |  |  |  |

3. Вычислить значения функции для аргумента 𝑋1 используя интерполяционную формулу Ньютона

Воспользуемся формулой Ньютона для интерполирования вперед(первая формула), так как X1 = 0.251 лежит в левой половине отрезка

Для X1 = 0.251:

4. **Вычислить значения функции для аргумента 𝑋2**, используя интерполяционную формулу **Гаусса**:

Воспользуемся первой формулой Гауса, так как X2 = 0.402 > a = 0.4

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | **x** | **y** | **Δ1** | **Δ2** | **Δ3** | **Δ4** | **Δ5** | **Δ6** |
| -3 | 0.25 | 1.2557 | 0.9207 | 0.0247 | -0.0437 | 1.0756 | -4.1277 | 10.1917 |
| -2 | 0.30 | 2.1764 | 0.9454 | -0.0190 | 1.0319 | -3.0521 | 6.0640 |  |
| -1 | 0.35 | 3.1218 | 0.9264 | 1.0129 | -2.0202 | 3.0119 |  |  |
| 0 | 0.40 | 4.0482 | 1.9393 | -1.0073 | 0.9917 |  |  |  |
| 1 | 0.45 | 5.9875 | 0.9320 | -0.0156 |  |  |  |  |
| 2 | 0.50 | 6.9195 | 0.9164 |  |  |  |  |  |
| 3 | 0.55 | 7.8359 |  |  |  |  |  |  |

=4.0482+0.04\*(1.9393)+1.0129\*0.04\*(-0.96)/2-2.0202\*0.04\*0.96\*1.004/6-3.0521\*1.96\*0.96\*1.04\*0.04/24 +6.0640\*0.04\*0.96\*1.004\*1.96\*2.004/120 - 10.1917\*0.04\*0.96\*1.004\*1.96\*2.004\*2.96/720≈4.084

# **2. Программная реализация**

# **Блок схемы**

Многочлен Ньютона с разделенными разностями

A diagram of a flowchart

AI-generated content may be incorrect.

Формула прямых конечных разностей

A diagram of a flowchart

AI-generated content may be incorrect.

Формула обратных конечных разностей

A diagram of a algorithm

AI-generated content may be incorrect.

Интерполяция Лагранжа

A diagram of a flowchart

AI-generated content may be incorrect.

Стирлинг:

A diagram of a computer flowchart

AI-generated content may be incorrect.

Бессел:

A diagram of a computer program

AI-generated content may be incorrect.

# **Листинг программы**

<https://github.com/senya-2011/Vu4Math/tree/main/lab5>

|  |
| --- |
| def lagrange(xs, ys, x):  *"""  Интерполяция Лагранжа.  """* n = len(xs)  total = 0.0  for i in range(n):  term = ys[i]  for j in range(n):  if j != i:  term \*= (x - xs[j]) / (xs[i] - xs[j])  total += term  return total  # Многочлен Ньютона с разделёнными разностями def newton\_divided(xs, ys, x):  *"""  Интерполяция Ньютона (разделённые разности).  """* n = len(xs)  coeff = list(ys)  for j in range(1, n):  for i in range(n - 1, j - 1, -1):  coeff[i] = (coeff[i] - coeff[i - 1]) / (xs[i] - xs[i - j])  result = coeff[0]  prod = 1.0  for i in range(1, n):  prod \*= (x - xs[i - 1])  result += coeff[i] \* prod  return result  # Многочлен Ньютона с конечными разностями (прямая формула) def newton\_forward(xs, ys, x):  *"""Формула прямых конечных разностей. Только для равноотстоящих узлов."""* h = xs[1] - xs[0]  table = finite\_diff\_table(ys)  t = (x - xs[0]) / h  result = ys[0]  for k in range(1, len(xs)):  coeff = 1.0  for j in range(k):  coeff \*= (t - j)  result += coeff \* table[k][0] / factorial(k)  return result  # Многочлен Ньютона с конечными разностями (обратная формула) def newton\_backward(xs, ys, x):  *"""Формула обратных конечных разностей. Только для равноотстоящих узлов."""* h = xs[1] - xs[0]  table = finite\_diff\_table(ys)  n = len(xs)  t = (x - xs[-1]) / h  result = ys[-1]  for k in range(1, n):  coeff = 1.0  for j in range(k):  coeff \*= (t + j)  result += coeff \* table[k][n - k - 1] / factorial(k)  return result  def stirling(xs, ys, x):  m = len(xs)   h = xs[1] - xs[0]  N = m - 1   diffs = finite\_diff\_table(ys)   alpha = N // 2  x0 = xs[alpha]  t = (x - x0) / h   res = diffs[0][alpha]   prod\_even = 1.0  prod\_odd = t   for k in range(1, N + 1):  if k % 2 == 1:  j = (k - 1) // 2  if j > 0:  prod\_odd \*= (t\*t - j\*j)  idx = alpha - j  if idx > 0:  d\_center = 0.5 \* (diffs[k][idx] + diffs[k][idx-1])  else:  d\_center = diffs[k][idx]  term = (prod\_odd \* d\_center) / factorial(k)  else:  j = k // 2  if j > 0:  prod\_even \*= (t\*t - (j-1)\*(j-1))  idx = alpha - j  term = (prod\_even \* diffs[k][idx]) / factorial(k)   res += term   return res   def bessel(xs, ys, x):  m = len(xs)   h = xs[1] - xs[0]  N = m - 1   diffs = finite\_diff\_table(ys)   i1 = (N // 2) + 1  i0 = i1 - 1   x0 = xs[i0]  t = (x - x0) / h   res = 0.5 \* (diffs[0][i0] + diffs[0][i1])   res += (t - 0.5) \* diffs[1][i0]   for k in range(2, N + 1):  if k % 2 == 0:  j = k // 2  prod = 1.0  for shift in range(-j, j):  prod \*= (t + shift)  idx = i0 - j + 1  diff\_avg = 0.5 \* (diffs[k][idx] + diffs[k][idx-1])  term = prod \* diff\_avg / factorial(k)  else:  j = (k - 1) // 2  prod = (t - 0.5)  for shift in range(-j, j):  prod \*= (t + shift)  idx = i0 - j  diff\_val = diffs[k][idx]  term = prod \* diff\_val / factorial(k)   res += term   return res |

## 

## **Примеры и результаты работы программы**

A graph on a screen

AI-generated content may be incorrect.

# **Вывод**:

В ходе выполнения лабораторной работы я изучил и практично применил интерполяционные методы Ньютона и Гаусса для обработки табличных данных. Эти методы позволили аппроксимировать значения функции в точках, отсутствующих в исходном наборе.

Разработанная программа обеспечила вычисление значений функции в заданных точках с использованием обоих методов. Проведённый сравнительный анализ показал совпадение результатов, что свидетельствует о правильной реализации и работоспособности выбранных алгоритмов.