



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«МИРЭА – Российский технологический университет»  
РТУ МИРЭА

---

---

Институт кибернетики  
Кафедра программного обеспечения систем радиоэлектронной аппаратуры

---

**ЛАБОРАТОРНАЯ РАБОТА № 4**

Вариант № 12

Студент группы: КМБО-02-19 Курса: 3

*С. А. Минеев*

Руководитель практики \_\_\_\_\_  
(Должность)

*А. А. Липатов*

«Отчет представлен  
к рассмотрению»

«\_\_\_» \_\_\_\_\_ 2022 г.

\_\_\_\_\_  
(подпись студента)

«Отчет утвержден.  
Допущен к защите»

«\_\_\_» \_\_\_\_\_ 2022 г.

\_\_\_\_\_  
(подпись руководителя)

Москва 2022

## Оглавление

1	ЗАДАНИЕ № 1.....	3
1.1	Текст задания.	3
1.2	Теоретическая часть.	3
1.3	Практическая часть.(Результаты работы программы.)	17
1.4	Приложение № 1 – Примеры работы программы.	19
1.5	Приложение № 2 – Листинг программы.	20
2	ЗАДАНИЕ № 2.....	25
2.1	Текст задания.	25
2.2	Теоретическая часть.	25
2.3	Практическая часть. (Результаты работы программы.)	36
2.4	Приложение № 1 – Примеры работы программы.	39
2.5	Приложение № 2 – Листинг программы.	40
3	ЗАДАНИЕ № 3.....	50
3.1	Текст задания.	50
3.2	Теоретическая часть.	50
3.3	Практическая часть. (Результаты работы программы.)	56
3.4	Приложение № 1 – Примеры работы программы.	59
3.5	Приложение № 2 – Листинг программы.	61

# 1 Задание № 1

## 1.1 Текст задания

Используя **первую** *или* **вторую** формулы Ньютона, вычислить значения функции, заданной таблично для заданных значений аргумента.

Таблица значений функции для варианта № 12	
X	Y
0.101	1.26183
0.106	1.27644
0.111	1.29122
0.116	1.30617
0.121	1.32130
0.126	1.33660
0.131	1.35207
0.136	1.36773
0.141	1.38357
0.146	1.39959
0.151	1.41579
0.156	1.42683
0.161	1.43356

№ Вар.	Таблица значений аргумента
12	$x = 0.1485 - 0.002 * n$ $n$ - Номер варианта.

## 1.2 Теоретическая часть

1.2.1 Рассмотрим случай, когда для определённой  $y = f(x)$  функции, непрерывной на некотором отрезке  $[a, b]$  известен некоторый набор(совокупность)  $n + 1$  аргументов  $x_i$  и соответственным им значения функции  $y_i : y_i = f(x_i)$ .

Особенностью в данном задании является тот факт, что аргументы  $x_i$ , которые называются **интерполяционными/аппроксимационными узлами** функции  $y = f(x)$ , находятся на некотором **фиксированном** расстоянии друг от друга, т. е.  $x_{i+1} - x_i = h$ .

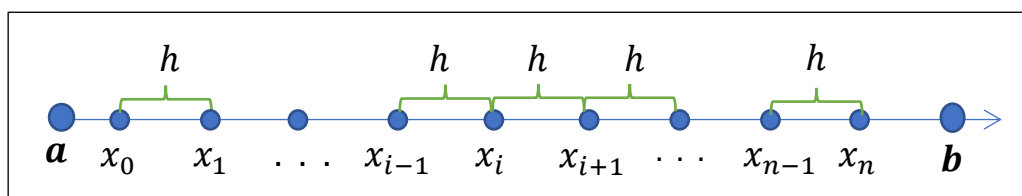


Рис. 1 – Иллюстрация расположения равноотстоящих интерполяционных узлов.

Если взять **упорядоченный** набор аргументов  $x_i$  и пронумеровать по индексу  $i = \overline{0, n}$ , то зная начальный элемент набора  $x_0$  можно однозначно определить  $x_i$  по следующей формуле:

$$x_i = x_0 + i \cdot h, \quad i = \overline{0, n}.$$

Рассмотрим далее понятие *аппроксимации* и *интерполяции* функций.

1.2.2 Во-первых, согласно аппроксимационной теореме Вейерштрасса [1885г.] для любой **непрерывной** на отрезке  $[x_0, x_n]$  функции  $y = f(x)$  можно подобрать последовательность многочленов, равномерно сходящихся к этой функции на данном отрезке.

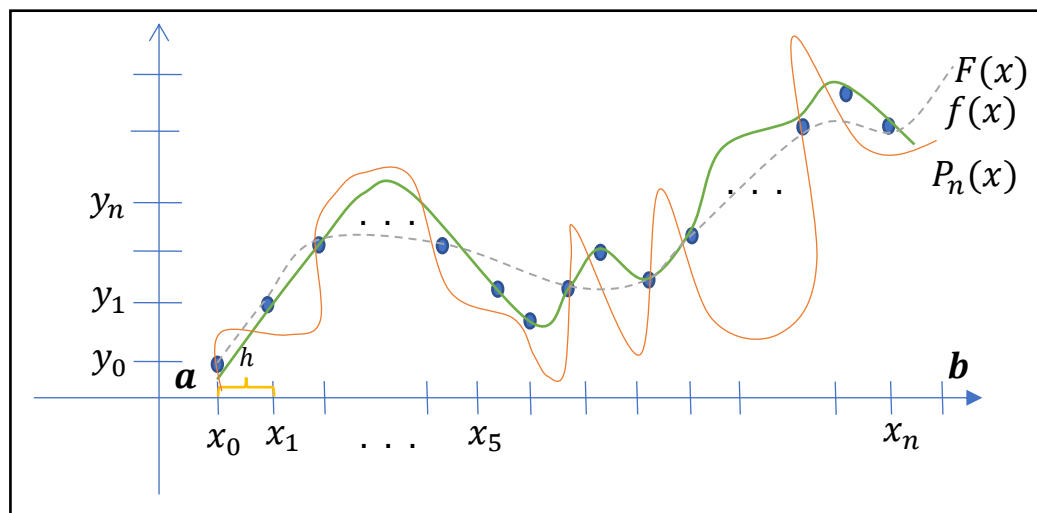


Рис. 2 – Иллюстрация  $P_n(x)$  – Интерполяционная функция.  
 $F(x)$  – Аппроксимационная функция.  
 $f(x)$  – оригинальная функция.

Именно на основе этой аппроксимационной теоремы Вейштрасса в математике в конце 70-х годов зародился метод, связанный с нахождением для сложных математических объектов, таких, например, как функции, более простых объектов, а метод был назван аппроксимацией.

### Определение № 1:

**Аппроксимация** (от лат. *proxima* — ближайшая, или приближение) — численный метод, состоящий в замене одних объектов (например,

тригонометрических функций) другими, в каком-то смысле близкими к исходным, но более простыми (например, алгебраическими функциями).

1.2.3 Во-вторых, из Рис. 2 видно, что при аппроксимации приходится “платить” точностью вычислений при упрощении функции т. к. аппроксимирующая функция строится не по узлам, а по отрезкам связывающие эти узлы, при этом могут иметься потери в конкретных узлах. В связи с этим были разработаны методы, которые строятся непосредственно по узлам функции и носят название методы интерполирования. Методы интерполирования позволяют учитывать все известные интерполирующие узлы и производить оценку точности вычислений при конкретных значениях функции.

### Определение № 2:

**Интерполирование** (от лат. *inter-polis* — «разглаженный, подновлённый, обновлённый; преобразованный») – численный метод, состоящий в нахождении промежуточных значений величины по имеющемуся дискретному набору известных значений.

### Задачи интерполяции:

- Научиться вычислять значение функции в любой наперед заданной точке.
- Построение алгебраических многочленов(полиномов)  $P_n(x)$  степени не выше, чем  $n$ , принимающих в точках  $x_i$  значения  $y_i$ .

Интерполяция иногда делится на **два** вида:

- 1)  $x \in [x_0, x_n]$  - собственная *интерполяция*. (Интерполяция в узком смысле.)
- 2)  $x \notin [x_0, x_n]$  - *экстраполяция*. (Интерполяция в широком смысле.)

**Факт № 1:** Термин «*интерполяция*» впервые употребил Джон Валлис в своём трактате «Арифметика бесконечных» (1656).

Стоит отметить, что, как и при аппроксимации, так и при интерполяции, имеется подвох в том, что мы не знаем, как поведет себя функция за пределами отрезка, на котором определена функция. В общем случае получается, что чем

далее аргумент (слева или справа от отрезка, на котором строится полином), тем менее точный результат вычислений получается. Поэтому на практике при интерполяции/аппроксимации делается приоритет тем таблицам функции, которые содержат наиболее точные данные и по возможности полностью покрывающие отрезок, на котором проводятся вычисления функции.

Перейдет теперь к непосредственной теории для интерполяции.

#### 1.2.4 Конечная разность

Когда интерполирующий многочлен – это полином, то вводится такое понятие как конечная разность.

Рассмотрим по подробнее.

Пусть есть какая-то функция  $y = f(x)$  и имеется приращение  $\Delta x = h$  – шаг.

$$\Delta y = \Delta f(x) = f(x + \Delta x) - f(x); \quad (1)$$

$\Delta y$  – Первая конечная разность.

В случае, когда  $x = x_0$  – конкретная точка, то:

$$\begin{cases} \Delta x = x - x_0; \\ \Delta f(x) = f(x_0 + \Delta x) - f(x_0); \\ \Delta f(x) = f(x) - f(x_0). \end{cases}$$

– Приращения функции для конкретной точки.

$$\begin{aligned} \Delta^2 y &= \Delta(\Delta f(x)) = \Delta(f(x + \Delta x) - f(x)) = \\ &= [f(x + 2 * \Delta x) - f(x + \Delta x)] - [f(x + \Delta x) - f(x)] = \\ &= f(x + 2 * \Delta x) - 2 * f(x + \Delta x) + f(x). \end{aligned}$$

$\Delta^2 y$  – Вторая конечная разность.

В общем случае:

Общая формула:  $\Delta^n y = \Delta(\Delta^{n-1} y)$ ; –  $n$  - ая конечная разность,  $n = \overline{1, n}$ .

Начальная формула:  $\Delta y = \Delta f(x) = f(x + \Delta x) - f(x)$ .

Пример № 1:

Дано:

$$P_3(x) = x^3, \quad \Delta x = 1;$$

Найти:

- Все конечные разности.

Решение:

$$1) \Delta P_3(x) = P_3(x+\Delta x) - P_3(x) = (x+\Delta x)^3 - x^3 = (x+1)^3 - x^3 = x^3 + 3x^2 + 3x + 1 - x^3 = 3x^2 + 3x + 1. \text{ — Значение первой конечной разности.}$$

$$\begin{aligned} 2) \Delta^2 P_3(x) &= \Delta(\Delta P_3(x)) = \Delta(3x^2 + 3x + 1) \\ &= [3 \cdot (x + \Delta x)^2 + 3(x + \Delta x) + 1] - [3x^2 + 3x + 1] \\ &= [3 \cdot (x^2 + 2 \cdot x \cdot \Delta x + \Delta^2 x) + 3x + 3\Delta x] - [3x^2 + 3x] \\ &= [3 \cdot (x^2 + 2 \cdot x + 1) + 3x + 3] - [3x^2 + 3x] \\ &= [3x^2 + 6 \cdot x + 3 + 3x + 3] - [3x^2 + 3x] = 3x^2 + 9x + 6 - 3x^2 - 3x \\ &= 6x + 6; \end{aligned}$$

$$3) \Delta^3 P_3(x) = \Delta(\Delta^2 P_3(x)) = \Delta(6x + 6) = [6(x + \Delta x) + 6] - [6x + 6] = 6; \text{ — Значение второй конечной разности.}$$

$$4) \Delta^4 P_3(x) = \Delta(\Delta^3 P_3(x)) = \Delta(6) = [6] - [6] = 0;$$

$$5) \Delta^n P_3(x) = 0, \text{ при } n > 3 \text{ конечные разности равны нулю.}$$

Ответ:

- $\Delta P_3(x) = 3x^2 + 3x + 1$ ; — Значение **первой** конечной разности.
- $\Delta^2 P_3(x) = 6x + 6$ ; — Значение **второй** конечной разности.
- $\Delta^3 P_3(x) = 6$ ; — Значение **третьей** конечной разности.
- $\Delta^n P_3(x) = 0$ , при  $n \geq 4$ ;

-----  
 $\Delta$  — оператор.  $f$  — функция.

**Свойства  $\Delta$ :**

- 1)  $\Delta[u + v] = \Delta u + \Delta v$ ;
- 2)  $\Delta(C \cdot u) = C \cdot \Delta u, C = \text{Const}$ ;
- 3)  $\Delta^m(\Delta^n u) = \Delta^{m+n} u$ ;
- 4)  $\Delta^0 y = y$ .

**Свойства  $f$ :**

- 1)  $f(x + \Delta x) = f(x) + \Delta f(x)$ ;
- 2)  $f(x + \Delta x) = (1 + \Delta) \cdot f(x)$ ;
- 3)  $n$  — раз:  
 $f(x + n \cdot \Delta x) = (1 + \Delta)^n f(x)$ ;  
 $f(x + n \cdot \Delta x) = \sum_{k=0}^n C_n^k \cdot \Delta^k f(x)$ .

### 1.2.5 Таблицы конечных разностей

Пусть нам известна некоторая функция  $y = f(x)$  и упорядоченная совокупность пар  $\{(x_i, y_i)\} : y_i = f(x_i)$ .

Что мы знаем ?

Если  $x_i$  – равноотстоящие точки, то мы можем ввести понятие  $\Delta x_i$ :

$$\begin{aligned}\Delta x_i &= x_{i+1} - x_i = h; \\ \Delta^1 y_i &= y_{i+1} - y_i; \\ \Delta^2 y_i &= \Delta(\Delta^1 y_i) = \Delta(y_{i+1} - y_i) = \Delta y_{i+1} - \Delta y_i; \\ \Delta^n y_i &= \Delta(\Delta^{n-1} y_i) = \Delta^{n-1}(y_{i+1} - y_i) = \Delta^{n-1} y_{i+1} - \Delta^{n-1} y_i;\end{aligned}$$

Эти формулы дают нам возможность построение конечных разностей в виде двух видов таблиц конечных разностей:

- **Горизонтальные таблицы разностей.**
- **Диагональные таблицы разностей.**

Рассмотрим каждую из таблиц по подробнее.

#### 1.2.1.5.1 Горизонтальные таблицы разностей

Общий вид горизонтальной таблицы конечных разностей:

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$x_0$	$y_0$	$\Delta^1 y_0$	$\Delta^2 y_0$	...	$\Delta^n y_0$
...	...	...	...	...	...
$x_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	...	$\Delta^n y_m$

Конкретный вид горизонтальной таблицы конечных разностей:

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$x_0$	$y_0$	$\Delta^1 y_0 = y_1 - y_0$	$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0$	...	$\Delta^n y_0 = C$
$x_1$	$y_1$	$\Delta^1 y_1 = y_2 - y_1$	$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1$	...	$\Delta^n y_1 = \emptyset$
...	...	...	...	...	...
$x_{m-1}$	$y_{m-1}$	$\Delta^1 y_{m-1} = y_m - y_{m-1}$	$\Delta^2 y_{m-1} = \emptyset$	...	$\Delta^n y_{m-1} = \emptyset$
$x_m$	$y_m$	$\Delta^1 y_m = \emptyset$	$\Delta^2 y_m = \emptyset$	...	$\Delta^n y_m = \emptyset$

#### Пример № 2:

Дано:

•  $y = 2x^3 - 2x^2 + 3x - 1, h = 1, x \in [0; 5];$

Задание:

- *Составить горизонтальную таблицу конечных разностей.*

Ответ:

❖ *Горизонтальная таблица:*



$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$C = 12$
$x_0 = 0$	$y_0 = 2(0)^3 - 2(0)^2 + 3(0) - 1 = (-1)$	$\Delta^1 y_0 = y_1 - y_0 = (2) - (-1) = 3$	$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0 = (11) - (3) = 8$	$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0 = (20) - (8) = 12$	
$x_1 = 1$	$y_1 = 2(1)^3 - 2(1)^2 + 3(1) - 1 = (2)$	$\Delta^1 y_1 = y_2 - y_1 = (13) - (2) = 11$	$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1 = (31) - (11) = 20$	$\Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1 = (32) - (20) = 12$	
$x_2 = 2$	$y_2 = 2(2)^3 - 2(2)^2 + 3(2) - 1 = (13)$	$\Delta^1 y_2 = y_3 - y_2 = (44) - (13) = 31$	$\Delta^2 y_2 = \Delta^1 y_3 - \Delta^1 y_2 = (63) - (31) = 32$	$\Delta^3 y_2 = \Delta^2 y_3 - \Delta^2 y_2 = (44) - (32) = 12$	
$x_3 = 3$	$y_3 = 2(3)^3 - 2(3)^2 + 3(3) - 1 = (44)$	$\Delta^1 y_3 = y_4 - y_3 = (107) - (44) = 63$	$\Delta^2 y_3 = \Delta^1 y_4 - \Delta^1 y_3 = (107) - (63) = 44$	$\Delta^3 y_3 = 12$	
$x_4 = 4$	$y_4 = 2(4)^3 - 2(4)^2 + 3(4) - 1 = (107)$	$\Delta^1 y_4 = y_5 - y_4 = (214) - (107) = 107$	$\Delta^2 y_4 = 44 + 12 = 56$	$\Delta^3 y_4 = 12$	
$x_5 = 5$	$y_5 = 2(5)^3 - 2(5)^2 + 3(5) - 1 = (214)$	$\Delta^1 y_5 = 107 + 56 = 163$	$\Delta^2 y_5 = 56 + 12 = 68$	$\Delta^3 y_5 = 12$	

### 1.2.1.5.2 Диагональные таблицы разностей

Общий вид диагональной таблицы конечных разностей.

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$x_0$	$y_0$				
		$\Delta^1 y_0$			
$x_1$	$y_1$		$\Delta^2 y_0$		
		$\Delta^1 y_1$		...	
$x_2$	$y_2$		$\Delta^2 y_1$		...
		$\Delta^1 y_2$		...	
$x_3$	$y_3$		$\Delta^2 y_2$		...
		$\Delta^1 y_3$		...	
$x_4$	$y_4$		$\Delta^2 y_3$		...
...	...	...	...	...	...
$x_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	...	$\Delta^n y_m$

Конкретный вид диагональной таблицы конечных разностей.

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$x_0$	$y_0$				
		$\Delta^1 y_0 = y_1 - y_0$			
$x_1$	$y_1$		$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0$		
		$\Delta^1 y_1 = y_2 - y_1$		...	
$x_2$	$y_2$		$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1$		...
		$\Delta^1 y_2 = y_3 - y_2$		...	
$x_3$	$y_3$		$\Delta^2 y_2 =$		...
		$\Delta^1 y_3 = y_4 - y_3$		...	
$x_4$	$y_4$		$\Delta^2 y_3$		...
...	...	...	...	...	...

$x_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	...	$\Delta^n y_m$
-------	-------	----------------	----------------	-----	----------------

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$x_0$	$y_0$	$\Delta^1 y_0 = y_1 - y_0$	$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0$	...	$\Delta^n y_0 = C$
$x_1$	$y_1$	$\Delta^1 y_1 = y_2 - y_1$	$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1$	...	$\Delta^n y_1 = \emptyset$
...	...	...	...	...	...
$x_{m-1}$	$y_{m-1}$	$\Delta^1 y_{m-1} = y_m - y_{m-1}$	$\Delta^2 y_{m-1} = 0$	...	$\Delta^n y_{m-1} = 0$
$x_m$	$y_m$	$\Delta^1 y_m = 0$	$\Delta^2 y_m = 0$	...	$\Delta^n y_m = 0$

### Пример № 3:

Дано:

•  $y = 2x^3 - 2x^2 + 3x - 1$ ,  $h = 1$ ,  $x \in [0; 5]$ ;

Задание:

• Составить горизонтальную таблицу конечных разностей.

Ответ:

❖ Горизонтальная таблица:

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$C = 12$
$x_0 = 0$	$y_0 = 2(0)^3 - 2(0)^2 + 3(0) - 1 = (-1)$	$\Delta^1 y_0 = y_1 - y_0 = (2) - (-1) = 3$	$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0 = (11) - (3) = 8$	$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0 = (20) - (8) = 12$	
$x_1 = 1$	$y_1 = 2(1)^3 - 2(1)^2 + 3(1) - 1 = (2)$	$\Delta^1 y_1 = y_2 - y_1 = (13) - (2) = 11$	$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1 = (31) - (11) = 20$	$\Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1 = (32) - (20) = 12$	
$x_2 = 2$	$y_2 = 2(2)^3 - 2(2)^2 + 3(2) - 1 = (13)$	$\Delta^1 y_2 = y_3 - y_2 = (44) - (13) = 31$	$\Delta^2 y_2 = \Delta^1 y_3 - \Delta^1 y_2 = (63) - (31) = 32$	$\Delta^3 y_2 = \Delta^2 y_3 - \Delta^2 y_2 = (44) - (32) = 12$	
$x_3 = 3$	$y_3 = 2(3)^3 - 2(3)^2 + 3(3) - 1 = (44)$	$\Delta^1 y_3 = y_4 - y_3 = (107) - (44) = 63$	$\Delta^2 y_3 = \Delta^1 y_4 - \Delta^1 y_3 = (107) - (63) = 44$	$\Delta^3 y_3 = 12$	
$x_4 = 4$	$y_4 = 2(4)^3 - 2(4)^2 + 3(4) - 1 = (107)$	$\Delta^1 y_4 = y_5 - y_4 = (214) - (107) = 107$	$\Delta^2 y_4 = 44 + 12 = 56$	$\Delta^3 y_4 = 12$	
$x_5 = 5$	$y_5 = 2(5)^3 - 2(5)^2 + 3(5) - 1 = (214)$	$\Delta^1 y_5 = 107 + 56 = 163$	$\Delta^2 y_5 = 56 + 12 = 68$	$\Delta^3 y_5 = 12$	

❖ Диагональная таблица:

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$C = 12$
$x_0 = 0$	$y_0 = 2(0)^3 - 2(0)^2 + 3(0) - 1 = (-1)$				
		$\Delta^1 y_0 = y_1 - y_0 = (2) - (-1) = 3$			
$x_1 = 1$	$y_1 = 2(1)^3 - 2(1)^2 + 3(1) - 1 = (2)$		$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0 = (11) - (3) = 8$		
		$\Delta^1 y_1 = y_2 - y_1 = (13) - (2) = 11$		$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0 = (20) - (8) = 12$	
$x_2 = 2$	$y_2 = 2(2)^3 - 2(2)^2 + 3(2) - 1 = (13)$		$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1 = (31) - (11) = 20$		
		$\Delta^1 y_2 = y_3 - y_2 = (44) - (13) = 31$		$\Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1 = (32) - (20) = 12$	
$x_3 = 3$	$y_3 = 2(3)^3 - 2(3)^2 + 3(3) - 1 = (44)$		$\Delta^2 y_2 = \Delta^1 y_3 - \Delta^1 y_2 = (63) - (31) = 32$		
		$\Delta^1 y_3 = y_4 - y_3 = (107) - (44) = 63$		$\Delta^3 y_2 = \Delta^2 y_3 - \Delta^2 y_2 = (44) - (32) = 12$	

$x_4 = 4$	$y_4 = 2(4)^3 - 2(4)^2 + 3(4) - 1 = (107)$		$\Delta^2 y_3 = \Delta^1 y_4 - \Delta^1 y_3 = (107) - (63) = 44$		
		$\Delta^1 y_4 = y_5 - y_4 = (214) - (107) = 107$		$\Delta^3 y_3 = 12$	
$x_5 = 5$	$y_5 = 2(5)^3 - 2(5)^2 + 3(5) - 1 = (214)$		$\Delta^2 y_4 = 44 + 12 = 56$		
		$\Delta^1 y_5 = 107 + 56 = 163$		$\Delta^3 y_4 = 12$	
			$\Delta^2 y_5 = 56 + 12 = 68$		
				$\Delta^3 y_5 = 12$	

### 1.2.6 Обобщенная степень.

Пусть имеется:

$n$  — обобщенная степень,

$x$  — число,

$h$  — фиксированный шаг;

#### Определение № 4:

**Обобщенной степенью** порядка  $n$  для числа  $x$  называется произведение  $n$  сомножителей, первый из которых равен  $x$ , а каждый следующий на шаг  $h$  меньше предыдущего.

$$x^{[n]} = \prod_{i=0}^{n-1} (x - i \cdot h) = x \cdot (x - h) \cdot (x - 2h) \cdot (x - 3h) \cdot \dots \cdot (x - (n - 1) \cdot h);$$

Полагают, что:

$$x^{[0]} = 1;$$

Если  $h = 0$  :

Обобщенная степень **совпадает** с обычной степенью, т. е.  $x^{[n]} = x^n$ ;

Если  $h = 1$  :

то мы получаем факториал, т. е.  $x^{[n]} = (x - n)!;$

Положим  $\Delta x = h$ ;

Тогда:

$$\Delta x^{[n]} = (x + h)^{[n]} - x^{[n]} = (x + h) \cdot x \cdot (x - h) \cdot (x - 2h) \cdot \dots \cdot (x - (n - 1)h) \cdot (x - h) \cdot (x - 2h) \cdot \dots \cdot (x - (n - 1)h) = x \cdot$$

$$(x-h)(x-2h) \cdot \dots \cdot (x-(n-2)h) \cdot [(x+h) - (x-(n-1)h)] = \\ x \cdot (x-h) \cdot (x-2h) \cdot \dots \cdot (x-(n-2)h) \cdot [nh] = n \cdot h \cdot x^{[n-1]};$$

$$\Delta^2 x^{[n]} = n \cdot (n-1) \cdot h^2 \cdot x^{[n-2]};$$

...

$$\Delta^k x^{[n]} = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot (n-(k-1)) \cdot h^k \cdot x^{[n-k]},$$

$$k = 1, 2, \dots;$$

Причем, если  $k > n$ , то такая конечная разность – это есть 0.  $\Delta^k x^{[n]} = 0$ .

-----

## 1.2.7 Интерполяционные формулы Ньютона.

### 1.2.1.7.1 I Интерполяционная формула Ньютона.[I И. Ф. Н.]

Имеем:

- Некоторая функция  $f(x)$ , известны узлы  $x_i$  – равноотстоящие узлы.
  - $y_i = f(x_i)$ ;
  - $x_i = x_0 + i \cdot h$ ;
  - Требуется подобрать полином  $P_n(x)$ , который
- $$P_n(x_i) = y_i; (1)$$

Это условие можно записать эквивалентным выражением в конечных разностях:

$$\Delta^m P_n(x_0) = \Delta^m y_0, \quad m = 1, 2, \dots, n;$$

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) + \dots + a_n(x - x_0)(x - x_1)(x - x_2) \cdot \dots \cdot (x - x_{n-1}) = a_0 + \\ a_1(x - x_0)^{[1]} + a_2(x - x_0)^{[2]} + a_3(x - x_0)^{[3]} + \dots + a_n(x - x_0)^{[n]}; (2)$$

- Надо найти коэффициенты:  $a_0, a_1, \dots, a_n$ ;
- Положим, что  $x = x_0$ ;
- Получим:

$$P_n(x_0) = a_0 = \{\text{При обращении к формуле (1)}\} = y_0;$$

- Положим, что  $x = x_1$ .

$$\Delta^1 P_n(x) = a_1 \cdot h + 2a_2 \cdot (x - x_0)^{[1]} \cdot h + 3a_3 \cdot (x - x_0)^{[2]} \cdot h + \dots + na_n(x - x_0)^{[n-1]} \cdot h;$$

$$\Delta^1 P_n(x_0) = a_1 h = \Delta y_0; \rightarrow a_1 = \frac{\Delta y_0}{h};$$

$$\Delta^2 P_n(x_0) = 2h^2 a_2 + 2 \cdot 3 \cdot a_3 (x - x_0)^{[1]} \cdot h^2 + 4 \cdot 3 \cdot a_4 (x - x_0)^{[2]} \cdot h^2 + \dots + (n-1) \cdot n \cdot h^2 \cdot a_n (x - x_0)^{[n-2]};$$

$$\Delta^2 P_n(x_0) = 2h^2 a_2; \rightarrow a_2 = \frac{\Delta^2 P_n(x_0)}{2h^2} = \frac{\Delta^2 y_0}{2! \cdot h^2};$$

...

$$a_i = \frac{\Delta^i P_n(x_0)}{2h^2} = \frac{\Delta^i y_0}{i! \cdot h^i}; (3), i = 0, 1, \dots, n;$$

стоит отметить, что  $\Delta^0 y_0 = y_0$ ;

• Подставим (3) в формулу (2):

• Получим:

$$P_n(x) = y_0 + \frac{\Delta y_0}{h} (x - x_0)^{[1]} + \frac{\Delta^2 y_0}{2! \cdot h^2} (x - x_0)^{[2]} + \dots + \frac{\Delta^n y_0}{n! \cdot h^n} (x - x_0)^{[n]};$$

– Первая интерполяционная формула Ньютона.

В целом такие расчеты довольно трудоёмки для облегчения использует новую переменную:

$$q = \frac{x - x_0}{h};$$

И полином принимает вид:

$$P_n(x) = y_0 + q \cdot \Delta y_0 + \frac{q \cdot (q-1)}{2!} \Delta^2 y_0 + \dots + \frac{q \cdot (q-1) \cdot (q-2) \cdot (q-3) \cdot \dots \cdot (q-n+1)}{n!} \Delta^n y_0; \quad (4)$$

- Первая формула Ньютона, которую используют для расчетов.

Пример № 2:

Дано:

$$y = e^x, \quad h = 0.05; \quad [a; b] = [3.5; 3.7];$$

Задание:

• Построить интерполирующий многочлен.

Решение:

1) Если мы построим на формулу (4), то у нас здесь присутствуют конечные разности. Если говорить по правильному, то степень многочлена должна быть такой сколько у нас конечных разностей. Степень многочлена  $n$  должна совпадать с тем количеством точек, которые у нас присутствуют в таблице. Но зачастую это не оправдано. И в общем случае делают многочлен с той степенью сколько у нас конечных разностей. Например, если у нас получается многочлен с тремя конечными разностями, то и будет у нас полином третьей степени. Количество суммируемых коэффициентов в многочлене определяется из таблицы разностей. Поэтому сначала строим таблицу конечных разностей.

2) Построение таблицы разностей:

$x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$C = 0.003$ $-0.002$
$x_0 = 3.5$	$y_0 \sim 33.115$	$\Delta^1 y_0 = y_1 - y_0 = 1.698$	$\Delta^2 y_0 = \Delta^1 y_1 - \Delta^1 y_0 = 0.087$	$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0 = 0.005$	$n = 3$
$x_1 = 3.55$	$y_1 \sim 34.813$	$\Delta^1 y_1 = y_2 - y_1 = 1.785$	$\Delta^2 y_1 = \Delta^1 y_2 - \Delta^1 y_1 = 0.092$	$\Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1 = 0.003$	-
$x_2 = 3.6$	$y_2 \sim 36.598$	$\Delta^1 y_2 = y_3 - y_2 = 1.877$	$\Delta^2 y_2 = \Delta^1 y_3 - \Delta^1 y_2 = 0.095$	-	-
$x_3 = 3.65$	$y_3 \sim 38.475$	$\Delta^1 y_3 = y_4 - y_3 = 1.972$	-	-	-
$x_4 = 3.7$	$y_4 \sim 40.447$	-	-	-	-

$$\begin{aligned}
 n &= 3; \rightarrow P_3(x) \\
 &= y_0 + q \cdot \Delta y_0 + \frac{q \cdot (q-1)}{2!} \Delta^2 y_0 + \frac{q \cdot (q-1) \cdot (q-2)}{3!} \Delta^3 y_0; \\
 q &= \frac{x - x_0}{h} = \frac{x - 3.5}{0.05}; \\
 P_3(x) &= y_0 + q \cdot \Delta y_0 + \frac{q \cdot (q-1)}{2!} \Delta^2 y_0 + \frac{q \cdot (q-1) \cdot (q-2)}{3!} \Delta^3 y_0 = 33.115 + \frac{x-3.5}{0.05} \cdot \\
 &1.698 + \frac{x-3.5}{0.05} \cdot \left( \frac{x-3.5}{0.05} - 1 \right) \cdot \frac{1}{2} \cdot 0.087 + \frac{x-3.5}{0.05} \cdot \left( \frac{x-3.5}{0.05} - 1 \right) \cdot \left( \frac{x-3.5}{0.05} - 2 \right) \cdot \\
 &\frac{1}{6} \cdot 0.005;
 \end{aligned}$$

Посчитаем значение в точке  $x = 3 \notin [3.5; 3.7]$ ;

$$\begin{aligned}
 x = 3: P_3(x) &= 33.115 + (-10) \cdot 1.698 + (-10) \cdot (-10 - 1) \cdot \frac{1}{2} \cdot 0.087 + \\
 &+ (-10) \cdot (-10 - 1) \cdot (-10 - 2) \cdot \frac{1}{6} \cdot 0.005 = 20,8189.
 \end{aligned}$$

Получим значение приближённое.

При сравнении с данными из таблицы имеем:

$$3 < 3.5; \rightarrow 20,8189 < 33,115;$$

$$\text{Ответ: } P_3(x) = 20,8189;$$

## Факт № 2:

- Эти интерполяционные формулы позволяют вне диапазона этих интерполяционных узлов. Чем дальше от таблицы разностей, тем менее точное значение мы можем получить.

### 1.2.1.7.2 II Интерполяционная формула Ньютона. [II И. Ф. Н.]

В целом I И. Ф. Н. нужна в основном в том случае, когда надо интерполировать в начале таблицы конечных разностей. Поскольку мы постоянно берем из таблицы  $y_0, \Delta y_0, \dots, \Delta^n y_0$ , а если интерполировать в конце таблицы, то она уже не очень хорошо применима т. к. у нас нет в конце таблицы конечных разностей и в этом случае нужно до заполнять таблицу неизвестными конечными разностями, что не очень удобно. Поэтому была выведена вторая интерполяционная формула Ньютона, как раз для интерполирования в конце таблицы.

Рассмотрим метод интерполирования по II И. Ф. Н.

Имеется набор значений  $y_i = f(x_i)$ ,  $x_i$  – равноотстоящие узлы т.е. их можно посчитать по формуле:

$$x_i = x_0 + i \cdot h, \quad i = 0, 1, \dots, n;$$

Имеется шаг:  $h$ ;

Выпишем интерполяционный многочлен в виде:

$$P_n(x) = a_0 + a_1(x - x_n) + a_2(x - x_n)(x - x_{n-1}) + a_3(x - x_0)(x - x_1)(x - x_2) + \dots + a_n(x - x_0)(x - x_1)(x - x_2) \cdot \dots \cdot (x - x_{n-n+1}) = a_0 + a_1(x - x_n)^{[1]} + a_2(x - x_{n-1})^{[2]} + a_3(x - x_{n-2})^{[3]} + \dots + a_n(x - x_1)^{[n]}; \quad (2)$$

$$P_n(x_i) = y_i;$$

$$\Delta^i P_n(x_{n-i}) = \Delta^i y_{n-i};$$

Положим  $x = x_n$ ;

$$P_n(x_n) = y_n = a_0; \Rightarrow a_0 = y_n;$$

$$\Delta^1 P_n(x) = a_1 h + 2a_2 \cdot (x - x_{n-1})^{[1]} \cdot h + 3a_3 \cdot (x - x_{n-2})^{[2]} \cdot h + \dots + n \cdot a_n (x - x_1)^{[n-1]} \cdot h;$$

Положим  $x = x_{n-1}$ ;

$$\Delta^1 P_n(x_{n-1}) = \Delta y_{n-1} = a_1 h; \Rightarrow a_1 = \frac{\Delta y_{n-1}}{h};$$

...

$$a_i = \frac{\Delta^i y_{n-i}}{i! \cdot h^i}, \quad i = 0, 1, \dots, n;$$

В итоге имеем:

$$\begin{aligned} P_n(x) = & y_n + \frac{\Delta y_{n-1}}{h} \cdot (x - x_n) + \frac{\Delta^2 y_{n-2}}{2! \cdot h^2} \cdot (x - x_n)(x - x_{n-1}) + \frac{\Delta^3 y_{n-3}}{3! \cdot h^3} \cdot (x - x_n)(x - x_{n-1})(x - x_{n-2}) + \dots + \\ & \frac{\Delta^n y_0}{n! \cdot h^n} \cdot (x - x_n) \cdot (x - x_{n-1}) \dots (x - x_1) = \\ & y_n + \frac{\Delta y_{n-1}}{h} \cdot (x - x_n)^{[1]} + \frac{\Delta^2 y_{n-2}}{2! \cdot h^2} \cdot (x - x_{n-1})^{[2]} + \frac{\Delta^3 y_{n-3}}{3! \cdot h^3} \cdot (x - x_{n-2})^{[3]} + \\ & \dots + \frac{\Delta^n y_0}{n! \cdot h^n} \cdot (x - x_1)^{[n]}; \end{aligned}$$

Сделаем замену:

$$q = \frac{x - x_n}{h};$$

$$\frac{x - x_{n-1}}{h} = \frac{x - x_n + h}{h} = q + 1;$$

$$\frac{x - x_{n-i}}{h} = \frac{x - x_n + ih}{h} = q + i;$$

Получим:

$$\begin{aligned} P_n(x) = & y_n + q \cdot \Delta y_{n-1} + \frac{q(q+1)}{2!} \Delta^2 y_{n-2} + \frac{q(q+1)(q+2)}{3!} \cdot \Delta^3 y_{n-3} + \dots + \\ & \frac{q(q+1)(q+2) \cdot \dots \cdot (q+n-1)}{n!} \cdot \Delta^n y_0; \end{aligned}$$

- Вторая интерполяционная формула Ньютона.

Пример № 29:

Дано:

$$y = f(x) = \lg(x); \quad h = 10;$$



Задание:

- Найти:  $\lg(1044)$  - ?

Решение:

1) Выписываем конечные разности.

$x_i \backslash \Delta^n y_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$C = 0$	$\Delta^n y_i$
$x_0 = 1000$	$y_0 = 3$	0,0043	0	...	$\Delta^n y_0$
$x_1 = 1010$	$y_1 = 3.0043$	0,0043	-0,001~0	...	...
$x_2 = 1020$	$y_2 = 3.0086$	0,042	0	...	-
$x_3 = 1030$	$y_3 = 3.0128$	0,042	0	-	-
$x_4 = 1040$	$y_4 = 3.0170$	0,042	-	-	-
$x_5 = 1050$	$y_5 = 3.0212$	-	-	-	-
$x_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	...	$\Delta^n y_m$

$$x_n = x_5 = 1050; h = 10; x = 1044; n = 2;$$

$$q = \frac{x - x_n}{h} = \frac{x - 1050}{10} = \frac{1044 - 1050}{10} = -\frac{6}{10} = (-0.6);$$

$$P_{n=5}(x = 1044) = y_n + q \cdot \Delta y_{n-1} + \frac{q(q+1)}{2!} \Delta^2 y_{n-2} = y_n + q \cdot \Delta y_{n-1} = 3.0212 + (-0.6) \cdot 0,042 = 3.01868;$$

Ответ:  $P_2(1044) = 3.01868$ ;

### 1.3 Практическая часть.(Результаты работы программы.)

Принятые данные :

А :

x	y
0.101	1.26183
0.106	1.27644
0.111	1.29122
0.116	1.30617
0.121	1.3213
0.126	1.3366
0.131	1.35207
0.136	1.36773
0.141	1.38357

0.146	1.39959
0.151	1.41579
0.156	1.42683
0.161	1.43356

**X<sub>0</sub> :**

0.1245

**Точность:**  $\varepsilon = 0.00001$

**Вычисления :**

**newVec**

0.121000	1.321300
0.126000	1.336600
0.131000	1.352070
0.136000	1.367730
0.141000	1.383570
0.146000	1.399590
0.151000	1.415790
0.156000	1.426830
0.161000	1.433560

**Вычисления :**

**newVec :**

0.121000	1.321300	0.015300	0.000170	0.000020	-0.000030	0.000040	-0.000050	-0.005280	0.027480
0.126000	1.336600	0.015470	0.000190	-0.000010	0.000010	-0.000010	-0.005330	0.022200	0
0.131000	1.352070	0.015660	0.000180	-0.000000	0.000000	-0.005340	0.016870	0	0
0.136000	1.367730	0.015840	0.000180	0.000000	-0.005340	0.011530	0	0	0
0.141000	1.383570	0.016020	0.000180	-0.005340	0.006190	0	0	0	0
0.146000	1.399590	0.016200	-0.005160	0.000850	0	0	0	0	0
0.151000	1.415790	0.011040	-0.004310	0	0	0	0	0	0
0.156000	1.426830	0.006730	0	0	0	0	0	0	0
0.161000	1.433560	0	0	0	0	0	0	0	0

**Предварительное выделение элементов для 1 И. Ф. Н. :**

**newVec :**

<b>0.12100</b>	<b>1.32130</b>	<b>0.01530</b>	<b>0.00017</b>	<b>0.00002</b>	<b>-0.00003</b>	<b>0.00004</b>	<b>-0.00005</b>	<b>-0.00528</b>	<b>0.02748</b>
0.12600	1.33660	0.01547	0.00019	-0.00001	0.00001	-0.00001	-0.00533	0.02220	0
0.13100	1.35207	0.01566	0.00018	-0.00000	0.00000	-0.00534	0.01687	0	0
0.13600	1.36773	0.01584	0.00018	0.00000	-0.00534	0.01153	0	0	0
0.14100	1.38357	0.01602	0.00018	-0.00534	0.00619	0	0	0	0
0.14600	1.39959	0.01620	-0.00516	0.00085	0	0	0	0	0
0.15100	1.41579	0.01104	-0.00431	0	0	0	0	0	0
0.15600	1.42683	0.00673	0	0	0	0	0	0	0
0.16100	1.43356	0	0	0	0	0	0	0	0

**Вычисления :**

$x = 0.1245$ ;  $x_0 = 0.121$ ;  $h = 0.005$ ;  $q = (x - x_0) / h = (0.1245 - 0.121) / 0.005 = 0.7$ ;

**Вычисления :**

$P_n(x) = P_8(x = 0.1245) = y_0 + q \Delta y_0 + q(q-1)/2! \Delta^2 y_0 + q(q-1)(q-2)/3! \Delta^3 y_0 + q(q-1)(q-2)(q-3)/4! \Delta^4 y_0 + q(q-1)(q-2)(q-3)(q-4)/5! \Delta^5 y_0 + q(q-1)(q-2)(q-3)(q-4)(q-5)/6! \Delta^6 y_0 + q(q-1)(q-2)(q-3)(q-4)(q-5)(q-6)/7! \Delta^7 y_0 + q(q-1)(q-2)(q-3)(q-4)(q-5)(q-6)(q-7)/8! \Delta^8 y_0 = 1.321736$  ;

Время работы: 0.146 ;

## 1.4 Приложение № 1 – Примеры работы программы.

MainWindow

Загрузка данных

Введите количество известных узлов таблицы: 13

X\_0 = 0.1245

Вычисление

Точность  $\epsilon = 0.00001$

Таблица значений функции для варианта № 12

x	y
0.101	1.26183
0.106	1.27644
0.111	1.29122
0.116	1.30617
0.121	1.3213
0.126	1.3366
0.131	1.35207
0.136	1.36773
0.141	1.38357
0.146	1.39959
0.151	1.41579
0.156	1.42683
0.161	1.43356

**Принятые данные :**

**A :**

x	y
0.101	1.26183
0.106	1.27644
0.111	1.29122
0.116	1.30617
0.121	1.3213
0.126	1.3366
0.131	1.35207
0.136	1.36773
0.141	1.38357
0.146	1.39959
0.151	1.41579
0.156	1.42683
0.161	1.43356

**X\_0 :**

0.1245

Номер варианта: N = 0.00001;

**Вычисления :**

**newVec :**

0.121000	1.321300
0.126000	1.336600
0.131000	1.352070
0.136000	1.367730
0.141000	1.383570
0.146000	1.399590
0.151000	1.415790
0.156000	1.426830
0.161000	1.433560

**Вычисления :**

**newVec :**

0.121000	1.321300	0.015300	0.000170	0.000020	-0.000030	0.000040	-0.000050	-0.005280	0.027480
0.126000	1.336600	0.015470	0.000190	-0.000010	0.000010	-0.000010	-0.005330	0.022200	0
0.131000	1.352070	0.015660	0.000180	-0.000000	0.000000	-0.005340	0.016870	0	0
0.136000	1.367730	0.015840	0.000180	0.000000	-0.005340	0.011530	0	0	0
0.141000	1.383570	0.016020	0.000180	-0.005340	0.006190	0	0	0	0
0.146000	1.399590	0.016200	-0.005160	0.000850	0	0	0	0	0
0.151000	1.415790	0.011040	-0.004310	0	0	0	0	0	0
0.156000	1.426830	0.006730	0	0	0	0	0	0	0
0.161000	1.433560	0	0	0	0	0	0	0	0

**Предварительное выделение элементов для 1 И. Ф. Н. :**

**newVec :**

0.12100	1.32130	0.01530	0.00017	0.00002	-0.00003	0.00004	-0.00005	-0.00528	0.02748
0.12600	1.33660	0.01547	0.00019	-0.00001	0.00001	-0.00001	-0.00533	0.02220	0
0.13100	1.35207	0.01566	0.00018	-0.00000	0.00000	-0.00534	0.01687	0	0
0.13600	1.36773	0.01584	0.00018	0.00000	-0.00534	0.01153	0	0	0
0.14100	1.38357	0.01602	0.00018	-0.00534	0.00619	0	0	0	0
0.14600	1.39959	0.01620	-0.00516	0.00085	0	0	0	0	0
0.15100	1.41579	0.01104	-0.00431	0	0	0	0	0	0
0.15600	1.42683	0.00673	0	0	0	0	0	0	0
0.16100	1.43356	0	0	0	0	0	0	0	0

**Вычисления :**

$x = 0.1245$ ;  $x_0 = 0.121$ ;  $h = 0.005$ ;  $q = (x - x_0) / h = (0.1245 - 0.121) / 0.005 = 0.7$ ;

**Вычисления :**

$P_n(x) = P_8(x = 0.1245) = y_0 + q \Delta y_0 + q(q-1)/2! \Delta^2 y_0 + q(q-1)(q-2)/3! \Delta^3 y_0 + q(q-1)(q-2)(q-3)/4! \Delta^4 y_0 + q(q-1)(q-2)(q-3)(q-4)/5! \Delta^5 y_0 + q(q-1)(q-2)(q-3)(q-4)(q-5)/6! \Delta^6 y_0 + q(q-1)(q-2)(q-3)(q-4)(q-5)(q-6)/7! \Delta^7 y_0 + q(q-1)(q-2)(q-3)(q-4)(q-5)(q-6)(q-7)/8! \Delta^8 y_0 = 1.321736$ ;

Время работы: 0.077;

## 1.5 Приложение № 2 – Листинг программы.

```
// Определение границ элемента x в матрицы A относительно строк.
int Ga, Gb;
double step_a, step_b;
for(size_t i = 0; i < CN; i++){
    if(i == 0){
        step_a = abs(A[i][0] - B[0]);
        Ga = i;
    }else if ( (step_a > (A[i][0] - B[0])) && (A[i][0] <= B[0])){
        step_a = abs(A[i][0] - B[0]);
        Ga = i;
    }

    if(i == 0){
        step_b = abs(A[(CN-1)][0] - B[0]);
        Gb = (CN - 1);
    }else if ( (step_b > (A[(CN-1) - i][0] - B[0])) && (A[(CN-1) - i][0] >= B[0])){
        //qDebug() << step_b;
        step_b = abs(A[(CN-1) - i][0] - B[0]);
        Gb = (CN-1) - i;
    }
}

// Метод вывода данных из матриц.
auto print_general = [&](QVector<QVector<double>>& vec) -> void {
    strHtml_A.clear();
    strHtml_A.append("<b style=\"color: red;\">Вычисления </b><b style=\"color: black;\">:</b><br>");
    strHtml_A.append("<b>newVec :</b><table border=1 style=\"border-collapse: collapse;\">");

    foreach(auto str, vec){
        strHtml_A.append("<tr>");
        foreach(auto elem, str){
            elem == 0 ? strHtml_A.append("<td align=center> " +
                QString::number(elem) + " </td>")
                : strHtml_A.append("<td align=center> " +
                QString::number(elem, 'f', 6) + " </td>");
        }
        strHtml_A.append("</tr>");
    }
    strHtml_A.append("</table>");
    ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() +
    strHtml_A + "<p></p>");
};

switch(Gb - Ga){
case(1):
    while(Ga != 0){ A.removeFirst(); Ga--; Gb--; CN--;}
    qDebug() << Ga;
    qDebug() << Gb;
    break;
}

print_general(A);
```

```

    auto print_general_N_1 = [&](QVector<QVector<double>>& vec, size_t Ga,
QVector<double>& c) -> void {
        strHtml_A.clear();
        strHtml_A.append("<b style=\"color: green;\">Предварительное выделение
элементов для 1 И. Ф. Г. </b><b style=\"color: black;\">:</b><br>");
        strHtml_A.append("<b>newVec :</b><table border=1 style=\"border-
collapse: collapse;\">");

        for(size_t i = 0; i < CN; i++){
            strHtml_A.append("<tr>");
            for(size_t j = 0; j < vec[i].count(); j++){
                if(i == Ga){
                    vec[i][j] == 0 ? strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 0) +
"</b></td>")
                                : strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 5) +
"</b></td>");
                    c.push_front(vec[i][j]);
                }else{
                    vec[i][j] == 0 ? strHtml_A.append("<td align=center> " +
QString::number(vec[i][j]) + " </td>")
                                : strHtml_A.append("<td align=center> " +
QString::number(vec[i][j], 'f', 5) + " </td>");
                }
            }
            strHtml_A.append("</tr>");
        }
        strHtml_A.append("</table>");
        ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
"<p></p>");
    };

    auto general_i = [&]() -> void {
        size_t n = 3;
        auto FL = [CN = CN] (QVector<QVector<double>> NewVec, size_t N, double
EPS) -> bool {
            bool FLAG = false;
            for(size_t i = 0; i < CN; i++){
                FLAG = (abs(NewVec[i][N-1]) > EPS) ? (true) : ((FLAG ==
true) ? (true) : (false));
            }
            return FLAG;
        };
    };

    uint64_t COUNT_R = 0;

```

```

    auto general = [&, CN = CN, print_general = print_general, eps = eps, FL
= FL, ui=ui, B=B, print_general_N_1=print_general_N_1](auto GEN, int64_t N,
QVector<QVector<double>> A) -> void {
    QVector<QVector<double>> newVec(CN, QVector<double>(N, 0));
    for(size_t i = 0; i < CN; i++){
        for(size_t j = 0; j < A[i].count(); j++){
            newVec[i][j] = A[i][j];

            if(j == 1){
                uint64_t COUNT = 0;
                std::string number = std::to_string(A[i][j]); //A[i][j]
number = number.substr(0,
number.find_last_not_of('0')); // Убираем последние нули
                size_t dotFound;
                std::stoi(number, &dotFound);

                COUNT = std::string(number).substr(dotFound).size();

                if(COUNT > COUNT_R){
                    COUNT_R = COUNT;
                }
            }
        }
    }

    bool FLAG = true;

    for(size_t i = 0; i < CN - (N - 2); i++){
        newVec[i][N-1] = (double)(newVec[i+1][N - 2] - newVec[i][N -
2]);

        uint64_t COUNT = 0;
        std::string number = std::to_string(newVec[i][N - 1]);
        number = number.substr(0, number.find_last_not_of('0')); //
Убираем последние нули;
        size_t dotFound;
        std::stoi(number, &dotFound);

        COUNT = std::string(number).substr(dotFound).size();

        double x = newVec[i][N - 1] - floor(newVec[i][N - 1]);

        double fractpart, // дробная часть
            intpart; // целая часть

        fractpart = modf( newVec[i][N - 1], &intpart);
        fractpart = QString::number(fractpart, 'f',
COUNT_R).toDouble();

        double f = QString::number(fractpart, 'f', COUNT_R -
1).toDouble();
        double f_i = fractpart - f;

        if(fractpart - f_i != 0){
            FLAG = false;
        }
    }
}

```

```

        if( FL(newVec, N, eps) && (N <= CN) && !FLAG){
            GEN(GEN, N + 1, newVec);
        }else{
            print_general(newVec); // print one, end iteration;
            print_general_N_1(newVec, Ga, X);

            QString strHtml;
            strHtml.clear();
            strHtml.append("<b style=\"color: red;\">Вычисления </b><b  

style=\"color: black;\">:</b><br>");
            strHtml.append("<b style=\"color: green;\" >x = </b> " +
QString::number(B[0]) + "; ");
            strHtml.append("<b style=\"color: green;\" >x_0 = </b> " +
QString::number(newVec[0][0]) + "; ");
            strHtml.append("<b style=\"color: green;\" >h = </b> " +
QString::number(newVec[1][0] - newVec[0][0]) + "; ");
            double q = (B[0] - newVec[0][0]) / (newVec[1][0] -
newVec[0][0]);
            strHtml.append("<b style=\"color: blue;\" >q = (x - x_0) / h =  

</b> (" + QString::number(B[0]) + " - " + QString::number(newVec[0][0]) + " )/  

" + QString::number(newVec[1][0] - newVec[0][0])  

            + " = " + QString::number(q) + "; ");
            ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() +
strHtml + "<p></p>");

            strHtml.clear();
            strHtml.append("<b style=\"color: red;\">Вычисления </b><b  

style=\"color: black;\">:</b><br>");

            auto P_n = [&]() -> double {

                auto ITER = [&](auto iter, size_t i) -> double {

                    if(i >= newVec[0].count() - 2/*COLUMN: x, y;*/ ){
                        return 0;
                    }

                    auto factorial = [&](auto fact, size_t i) -> double {
                        return (i == 1 || i == 0) ? i/*База факториала*/ :
i*fact(fact, i - 1);
                    };

                    auto Q = [&](auto Q, size_t i) -> double {
                        if (i == 0){
                            return q;
                        }else if(i < 0){
                            return 1;
                        }else{
                            return (q-i)*Q(Q, i-1);
                        }
                    };

                    return newVec[0][2/*COLUMN: x, y;*/ + i]*Q(Q,i-1) /
factorial(factorial, i) + iter(iter, i + 1);
                };

                return newVec[0][1] + ITER(ITER, 1);
            };

```

```

switch(N - 2/*COLUMN: x, y;*/){
    case(8):
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(N - 2/*COLUMN: x, y;*/)+
"</sub>(x = " + QString::number(B[0]) + ") = y<sub>0</sub> "
            "+ q*Δy<sub>0</sub> + q(q-1)/2! *
Δ<sup>2</sup>y<sub>0</sub> + q(q-1)(q-2)/3! * Δ<sup>3</sup>y<sub>0</sub>
+ q(q-1)(q-2)(q-3)/4! * Δ<sup>4</sup>y<sub>0</sub> "
            "+ q(q-1)(q-2)(q-3)(q-4)/5! *
Δ<sup>5</sup>y<sub>0</sub> + q(q-1)(q-2)(q-3)(q-4)(q-5)/6! *
Δ<sup>6</sup>y<sub>0</sub> "
            "+ q(q-1)(q-2)(q-3)(q-4)(q-5)(q-6)/7!
* Δ<sup>7</sup>y<sub>0</sub> + q(q-1)(q-2)(q-3)(q-4)(q-5)(q-6)(q-7)/8! *
Δ<sup>8</sup>y<sub>0</sub>"
            " </b> = " + QString::number(P_n(),
'f', 6) + " ; ");
        break;
    case(2):
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(N - 2/*COLUMN: x, y;*/)+
"</sub>(x = " + QString::number(B[0]) + ") = y<sub>0</sub> "
            "+ q*Δy<sub>0</sub> + q(q-1)/2! *
Δ<sup>2</sup>y<sub>0</sub> "
            " </b> = " + QString::number(P_n(),
'f', 6) + " ; ");
        break;
    case(1):
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(N - 2/*COLUMN: x, y;*/)+
"</sub>(x = " + QString::number(B[0]) + ") = y<sub>0</sub> "
            "+ q*Δy<sub>0</sub> </b> = " +
QString::number(P_n(), 'f', 6) + " ; ");
        break;
    default:
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(N - 2/*COLUMN: x, y;*/)+
"</sub>(x = " + QString::number(B[0]) + ") = y<sub>0</sub> "
            "+ q*Δy<sub>0</sub> </b> = " +
QString::number(P_n(), 'f', 6) + " ; ");
        break;
};

ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() +
strHtml + "<p></p>");
}
};
general(general, n, A);
};
general_i();

```

Ссылка на git-Hub: [https://github.com/MineevS/CHM\\_3\\_6.git](https://github.com/MineevS/CHM_3_6.git)



## 2 Задание № 2

### 2.1 Текст задания.

Используя интерполяционные формулы *Гаусса*, *Стирлинга* и *Бесселя*, вычислить приближенные значения функции  $Y = F(X)$  при заданных значениях аргумента, если исходные значения функции  $Y = F(X)$  представлены таблицей.:

Таблица значений функции  $Y = F(X)$ .

X	Y	X	Y	X	Y
1.50	15.132	1.75	32.812	2.00	59.158
1.55	17.422	1.80	37.857	2.05	64.817
1.60	20.393	1.85	43.189	2.10	69.550
1.65	23.994	1.90	48.689	2.15	74.782
1.70	28.160	1.95	54.225	2.20	79.548

Значения аргумента для интерполяционных формул.

Гаусса	Стирлинга	Бесселя
$x = 1.60 + 0.006 * N$	$x = 1.725 + 0.002 * N$	$x = 1.83 + 0.003 * N$
N – номер варианта		
$x = 1.6 + 0.006 * 12 = 1.672$	$x = 1.725 + 0.002 * 12 = 1.749$	$x = 1.83 + 0.003 * 12 = 1.83 + 0.036n$

### 2.2 Теоретическая часть.

#### 2.2.1 Особенности использования интерполяционных формул Ньютона.

- Условия использования Интерполяционных формул Ньютона:
- ✓ Если  $x < x_0$ ,  $x$  **близко** к  $x_0$ , то  $q = \frac{x - x_0}{h}; < 0$ ; – Используем I ИФН.
- ✓ Если  $x > x_n$ ,  $x$  **близко** к  $x_n$ ,  $q = \frac{x - x_n}{h}; > 0$ ; – Используем II ИФН.

В частных случаях может возникнуть ситуация, когда мы выходим за пределы таблицы:

$\Delta^n y_i$ $x_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
--	-	-	-	-	-
$x_0$	$y_0$	$\Delta^1 y_0$	$\Delta^2 y_0$	...	$\Delta^n y_0$

...	...	...	...	...	...
$x_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	...	$\Delta^n y_m$
--	-	-	-	-	-

(--) - находимся близко к значениям таблицы.

Если аргумент функции находится близко к таблице конечных разностей по первому столбцу, то значения И. Ф. Н. будут достаточно точными. Но чем дальше от таблицы требуется найти значение, тем менее точное получается значение для заданной функции. В этом случае прибегают к методу экстраполирования. Процесс нахождения значений полинома за пределами заданного интервала *экстраполированием*.

#### Определение № 4:

**Экстраполирование(Экстраполяция)** – распространение результатов, полученных из наблюдений над одной частью некоторого явления, на другую его часть.

**Экстраполяция функции** – продолжение функции за пределы её области определения, при котором продолженная функция (как правило, аналитическая) принадлежит заданному классу функций.

В случае, когда уходим в бесконечность от значений таблицы, то значения интерполяционного многочлена распадается.

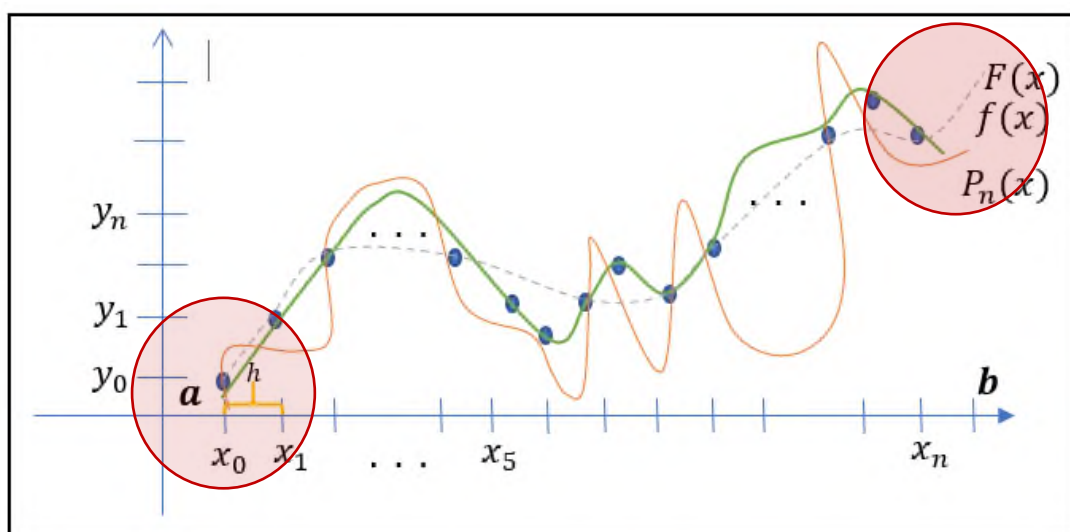


Рис. 2 – Иллюстрация  $P_n(x)$  – Интерполяционная функция.  
 $F(x)$  – Аппроксимационная функция.

$f(x)$  – оригинальная функция.

**Итог:** Интерполяционные формулы Ньютона очень удобно использовать, когда значения ищутся либо в начале таблицы, либо в конце.

В случае, когда надо искать значения где-то в середине таблицы, то использование интерполяционных формул Ньютона не очень подходит, хотя и позволяет найти значение с определенной точностью. Но в целом для этого случая имеются другие интерполяционные формулы, которые делают это более быстро и более просто. Для того, чтобы их рассмотреть нужно ввести такое понятие как таблица центральных разностей.

### 2.2.2 Таблицы центральных разностей.

Часто, когда интерполирование происходит где-то в центре таблицы конечных разностей, удобно использовать не первые и последние значения, а близкие к тем, которые нам надо получить, но обычно по обе стороны.

Имеем:

- Таблицу конечных разностей:

$x_i \backslash \Delta^n y_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$x_0 = 1000$	$y_0 = 3$	$\Delta^1 y_0$	$\Delta^2 y_0$	...	$\Delta^n y_0$
$x_1 = 1010$	$y_1 = 3.0043$	...	...	...	...
$x_2 = 1020$	$y_2 = 3.0086$	...	...	...	...
$x_3 = 1030$	$y_3 = 3.0128$	...	...	...	...
$x_4 = 1040$	$y_4 = 3.017$	...	...	...	...
$x_4 = 1050$	$y_5 = 3.0212$	...	...	...	...
$x_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	...	$\Delta^n y_m$

- Таблица центральных разностей:

$z_i \backslash \Delta^n y_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	...	$\Delta^n y_i$
$z_{-3}$	$y_{-3}$	$\Delta^1 y_{-2}$	$\Delta^2 y_{-1}$	...	$\Delta^n y_0$
$z_{-2}$	$y_{-2}$	$\Delta^1 y_{-1}$	$\Delta^2 y_0$	...	...
$z_{-1}$	$y_{-1}$	$\Delta^1 y_0$	$\Delta^2 y_1$		
$z_0$	$y_0$	$\Delta^1 y_1$	$\Delta^2 y_2$		
$z_1$	$y_1$	$\Delta^1 y_2$	$\Delta^2 y_3$		

$x_2$	$y_2$	$\Delta^1 y_3$	$\Delta^2 y_4$		
$z_m$	$y_m$	$\Delta^1 y_m$	$\Delta^2 y_m$	$\dots$	$\Delta^n y_m$

Появляются отрицательные индексы. Которые помогают получить местоположение узла  $z_0$ .

$$\Delta^1 y_{-1}, \Delta^1 y_0, \Delta^2 y_{-1}, \Delta^2 y_0, \dots;$$

Формулы для интерполирования на середины: I, II интерполяционные формулы Гаусса, Интерполяционная формула Стирлинга, интерполяционная формула Бесселя.

### 2.2.3 Интерполяционные формулы Гаусса.

#### 2.2.3.1 I Интерполяционная формула Гаусса. [I И.Ф. Г.]

- Имеется  $2n + 1$  (нечетное) равноотстоящих узлов.

$$x_{-n}, x_{-(n-1)}, \dots, x_{-1}, x_0, x_1, \dots, x_{n-1}, x_n;$$

$$\Delta x_i = x_{i+1} - x_i = h;$$

- Имеется функция:  $y = f(x)$ ;
- Известны значения функции в узлах:  $y_i = f(x_i)$ ;
- Требуется построить интерполяционный полином  $P_n(x_i) = y_i$ , причем этот полином не выше степени  $2n$ .

(Потому что у нас  $2n + 1$  узел).

$$\text{Отсюда: } \Delta^k P_n(x_i) = \Delta^k y_i;$$

Распишем полином:

$$\begin{aligned} P_n(x) &= a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_{-1})(x - x_0)(x - x_1) + \dots + \\ &+ a_{2n-1}(x - x_{-(n-1)}) \cdot \dots \cdot (x - x_{-1})(x - x_0)(x - x_1) \cdot \dots \cdot \\ &\cdot (x - x_{n-1}) + a_{2n}(x - x_{-(n-1)}) \cdot \dots \cdot (x - x_{-1})(x - x_0)(x - x_1) \cdot \dots \cdot \\ &\cdot (x - x_n) = a_0 + a_1(x - x_0)^{[1]} + a_2(x - x_0)^{[2]} + a_3(x - x_{n-1})^{[3]} + \dots + \\ &+ a_{2n}(x - x_{-(n-1)})^{[2n]}; \end{aligned}$$

Предположим,  $x = x_0$ ;

$$a_0 = y_0;$$

$$\begin{aligned}
a_1 &= \frac{\Delta y_0}{h}; \\
a_2 &= \frac{\Delta^2 y_0}{2! h^2}; \\
a_3 &= \frac{\Delta^3 y_{-1}}{3! h^3}; \\
a_4 &= \frac{\Delta^4 y_{-2}}{4! h^4}; \\
&\dots \\
a_{2n-1} &= \frac{\Delta^{2n-1} y_{-(n-1)}}{(2n-1)! h^{2n-1}}; \\
a_{2n} &= \frac{\Delta^{2n} y_{-n}}{(2n)! h^{2n}};
\end{aligned}$$

Если подставим в полином и введем:

$$q = \frac{x - x_0}{h};$$

$$\begin{aligned}
P_n(x) = & y_0 + q\Delta y_0 + \frac{q(q-1)}{2!} \Delta^2 y_{-1} + \frac{(q+1)q(q-1)}{3!} \Delta^3 y_{-1} \\
& + \frac{(q+1)q(q-1)(q-2)}{4!} \Delta^4 y_{-2} + \\
& \dots + \frac{(q+n-1) \cdot \dots \cdot (q+1)q(q-1)(q-2) \cdot \dots \cdot (q-n+1)}{(2n-1)!} \Delta^{2n-1} y_{-(n-1)} \\
& + \frac{(q+n-1) \cdot \dots \cdot (q+1)q(q-1) \cdot \dots \cdot (q-n)}{(2n)!} \Delta^{2n} y_{-n};
\end{aligned}$$

- Первая интерполяционная формула Гаусса.[Г И.Ф.Г.]

Первая интерполяционная формула гаусса использует следующие центральные разности:

$$\Delta y_0, \Delta^2 y_{-1}, \Delta^3 y_{-1}, \Delta^4 y_{-2}, \Delta^5 y_{-2}, \dots, \Delta^{2n-1} y_{-(n-1)}, \Delta^{2n} y_{-n}.$$

Интерполяционный многочлен идет **вверх в начало** таблицы.

$\Delta^n y_i$ $z_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$\Delta^4 y_i$	$\Delta^5 y_i$	...	$\Delta^n y_i$
...	...	...	...	...	...	...	...	...
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-2}$	...	$\Delta^n y_{n-7}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-2}$	$\Delta^5 y_{-1}$	...	$\Delta^n y_{n-6}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-1}$	$\Delta^5 y_0$	...	$\Delta^n y_{n-5}$
$z_{-4}$	$y_{-4}$	$\Delta^1 y_{-3}$	$\Delta^2 y_{-2}$	$\Delta^3 y_{-1}$	$\Delta^4 y_0$	$\Delta^5 y_1$	...	$\Delta^n y_{n-4}$

$z_{-3}$	$y_{-3}$	$\Delta^1 y_{-2}$	$\Delta^2 y_{-1}$	$\Delta^3 y_0$	$\Delta^4 y_1$	$\Delta^5 y_2$	...	$\Delta^n y_{n-3}$
$z_{-2}$	$y_{-2}$	$\Delta^1 y_{-1}$	$\Delta^2 y_0$	$\Delta^3 y_1$	$\Delta^4 y_2$	$\Delta^5 y_3$	...	$\Delta^n y_{n-2}$
$z_{-1}$	$y_{-1}$	$\Delta^1 y_0$	$\Delta^2 y_1$	$\Delta^3 y_2$	$\Delta^4 y_3$	$\Delta^5 y_4$	...	$\Delta^n y_{n-1}$
$z_0$	$y_0$	$\Delta^1 y_1$	$\Delta^2 y_2$	$\Delta^3 y_3$	$\Delta^4 y_4$	$\Delta^5 y_5$	...	$\Delta^n y_{n+0}$
$z_1$	$y_1$	$\Delta^1 y_2$	$\Delta^2 y_3$	$\Delta^3 y_4$	$\Delta^4 y_5$	$\Delta^5 y_6$	...	$\Delta^n y_{n+1}$
$x_2$	$y_2$	$\Delta^1 y_3$	$\Delta^2 y_4$	$\Delta^3 y_5$	$\Delta^4 y_6$	$\Delta^5 y_7$	...	$\Delta^n y_{n+2}$
...	...	...	...	...	...	...	...	...
$z_m$	$y_m$	$\Delta^1 y_{m+1}$	$\Delta^2 y_{m+2}$	$\Delta^3 y_{m+3}$	$\Delta^4 y_{m+4}$	$\Delta^5 y_{m+5}$	...	$\Delta^n y_{n+n}$

### 2.2.3.2 II Интерполяционная формула Гаусса. [II И.Ф. Г.]

Можно построить ещё одну интерполяционную формулу Гаусса, но на следующих центральных разностях:

$$\Delta y_{-1}, \Delta^2 y_{-1}, \Delta^3 y_{-2}, \Delta^4 y_{-2}, \Delta^5 y_{-3}, \Delta^6 y_{-3}, \dots, \Delta^{2n-1} y_{-(n-1)}, \Delta^{2n} y_{-n}.$$

$\Delta^n y_i$ $z_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$\Delta^4 y_i$	$\Delta^5 y_i$	$\Delta^6 y_i$	...	$\Delta^n y_i$
...	...	...	...	...	...	...	...	...	...
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-3}$	...	$\Delta^n y_{n-9}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-2}$	...	$\Delta^n y_{n-8}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-2}$	$\Delta^5 y_{-1}$	...	$\Delta^n y_{n-7}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-2}$	$\Delta^5 y_{-1}$	$\Delta^5 y_0$	...	$\Delta^n y_{n-6}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-1}$	$\Delta^5 y_0$	$\Delta^5 y_1$	...	$\Delta^n y_{n-5}$
$z_{-4}$	$y_{-4}$	$\Delta^1 y_{-3}$	$\Delta^2 y_{-2}$	$\Delta^3 y_{-1}$	$\Delta^4 y_0$	$\Delta^5 y_1$	$\Delta^5 y_2$	...	$\Delta^n y_{n-4}$
$z_{-3}$	$y_{-3}$	$\Delta^1 y_{-2}$	$\Delta^2 y_{-1}$	$\Delta^3 y_0$	$\Delta^4 y_1$	$\Delta^5 y_2$	$\Delta^5 y_3$	...	$\Delta^n y_{n-3}$
$z_{-2}$	$y_{-2}$	$\Delta^1 y_{-1}$	$\Delta^2 y_0$	$\Delta^3 y_1$	$\Delta^4 y_2$	$\Delta^5 y_3$	$\Delta^5 y_4$	...	$\Delta^n y_{n-2}$
$z_{-1}$	$y_{-1}$	$\Delta^1 y_0$	$\Delta^2 y_1$	$\Delta^3 y_2$	$\Delta^4 y_3$	$\Delta^5 y_4$	$\Delta^5 y_5$	...	$\Delta^n y_{n-1}$
$z_0$	$y_0$	$\Delta^1 y_1$	$\Delta^2 y_2$	$\Delta^3 y_3$	$\Delta^4 y_4$	$\Delta^5 y_5$	$\Delta^5 y_6$	...	$\Delta^n y_{n+0}$
$z_1$	$y_1$	$\Delta^1 y_2$	$\Delta^2 y_3$	$\Delta^3 y_4$	$\Delta^4 y_5$	$\Delta^5 y_6$	$\Delta^5 y_7$	...	$\Delta^n y_{n+1}$
$x_2$	$y_2$	$\Delta^1 y_3$	$\Delta^2 y_4$	$\Delta^3 y_5$	$\Delta^4 y_6$	$\Delta^5 y_7$	$\Delta^6 y_8$	...	$\Delta^n y_{n+2}$
...	...	...	...	...	...	...	...	...	...
$z_m$	$y_m$	$\Delta^1 y_{m+1}$	$\Delta^2 y_{m+2}$	$\Delta^3 y_{m+3}$	$\Delta^4 y_{m+4}$	$\Delta^5 y_{m+5}$	$\Delta^5 y_{m+5}$	...	$\Delta^n y_{n+n}$

В отличие от I И.Ф.Г. центральные разности начинаются не с  $\Delta y_0$ , а с  $\Delta y_{-1}$ , т. е. II И. Ф. Г. еще быстрее приходит в начало таблицы.

Будем иметь:

$$\begin{aligned}
 P_n(x) = & y_0 + q\Delta y_{-1} + \frac{(q+1)q}{2!}\Delta^2 y_{-1} + \frac{(q+1)q(q-1)}{3!}\Delta^3 y_{-2} \\
 & + \frac{(q+2)(q+1)q(q-1)}{4!}\Delta^4 y_{-2} + \dots \\
 & + \frac{(q+n-1)\dots(q+1)q(q-1)(q-2)\dots(q-n+1)}{(2n-1)!}\Delta^{2n-1} y_{-n} \\
 & + \frac{(q+n)\dots(q+1)q(q-1)\dots(q-n+1)}{(2n)!}\Delta^{2n-1(2n)} y_{-n};
 \end{aligned}$$

- Вторая интерполяционная формула Гаусса.

#### 2.2.4 Интерполяционная формула Стирлинга. [И. Ф. С.]

Кроме интерполяционных формул Гаусса, которые все равно требуют интерполирования или подсчета центральных разностей в начале таблицы, поскольку индексы все время уходят в минус, то мы все время приближаемся в начало таблицы. Это не всегда удобно. Для этого существуют еще несколько интерполяционных формул. Одна из них интерполяционная формула Стирлинга. Она представляет собой как среднее арифметическое I и II И. Ф. Г. .

А именно:

$$[\text{И. Ф. С.}] = \frac{[\text{I И. Ф. Г.}] + [\text{II И. Ф. Г.}]}{2};$$

$$\begin{aligned}
 P_n(x) = & y_0 + q \frac{\Delta y_{-1} + \Delta y_0}{2} + \frac{q^2}{2!} \Delta^2 y_{-1} + \frac{q(q^2-1^2)}{3!} \cdot \frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2} \\
 & + \frac{q^2(q^2-1^2)}{4!} \Delta^4 y_{-2} + \frac{q(q^2-1^2)(q^2-2^2)}{5!} \cdot \frac{\Delta^5 y_{-3} + \Delta^5 y_{-2}}{2} \\
 & + \frac{q^2(q^2-1^2)(q^2-2^2)}{5!} \Delta^6 y_{-3} \dots + \frac{q(q^2-1^2)\dots(q^2-(n-1)^2)}{(2n-1)!} \\
 & \cdot \frac{\Delta^{2n-1} y_{-n} + \Delta^{2n-1} y_{-(n-1)}}{2} + \frac{q^2(q^2-1^2)\dots(q^2-(n-1)^2)}{(2n)!} \Delta^{2n} y_{-n};
 \end{aligned}$$

- Интерполяционная формула Стирлинга.

Интерполяционная формула Стирлинга использует следующие конечные разности:

$$\begin{aligned}
 & \frac{\Delta y_{-1} + \Delta y_0}{2}, \Delta^2 y_{-1}, \frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2}, \Delta^4 y_{-2}, \frac{\Delta^5 y_{-3} + \Delta^5 y_{-2}}{2}, \Delta^6 y_{-3}, \dots, \\
 & \dots, \frac{\Delta^{2n-1} y_{-n} + \Delta^{2n-1} y_{-(n-1)}}{2}, \Delta^{2n} y_{-n}.
 \end{aligned}$$

Таблица выделенных элементов по И. Ф. С.:

$\Delta^n y_i$ $z_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$\Delta^4 y_i$	$\Delta^5 y_i$	$\Delta^6 y_i$	...	$\Delta^n y_i$
...	...	...	...	...	...	...	...	...	...
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-3}$	...	$\Delta^n y_{n-9}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-2}$	...	$\Delta^n y_{n-8}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-2}$	$\Delta^5 y_{-1}$	...	$\Delta^n y_{n-7}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-2}$	$\Delta^5 y_{-1}$	$\Delta^5 y_0$	...	$\Delta^n y_{n-6}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-1}$	$\Delta^5 y_0$	$\Delta^5 y_1$	...	$\Delta^n y_{n-5}$
$z_{-4}$	$y_{-4}$	$\Delta^1 y_{-3}$	$\Delta^2 y_{-2}$	$\Delta^3 y_{-1}$	$\Delta^4 y_0$	$\Delta^5 y_1$	$\Delta^5 y_2$	...	$\Delta^n y_{n-4}$
$z_{-3}$	$y_{-3}$	$\Delta^1 y_{-2}$	$\Delta^2 y_{-1}$	$\Delta^3 y_0$	$\Delta^4 y_1$	$\Delta^5 y_2$	$\Delta^5 y_3$	...	$\Delta^n y_{n-3}$
$z_{-2}$	$y_{-2}$	$\Delta^1 y_{-1}$	$\Delta^2 y_0$	$\Delta^3 y_1$	$\Delta^4 y_2$	$\Delta^5 y_3$	$\Delta^5 y_4$	...	$\Delta^n y_{n-2}$
$z_{-1}$	$y_{-1}$	$\Delta^1 y_0$	$\Delta^2 y_1$	$\Delta^3 y_2$	$\Delta^4 y_3$	$\Delta^5 y_4$	$\Delta^5 y_5$	...	$\Delta^n y_{n-1}$
$z_0$	$y_0$	$\Delta^1 y_1$	$\Delta^2 y_2$	$\Delta^3 y_3$	$\Delta^4 y_4$	$\Delta^5 y_5$	$\Delta^5 y_6$	...	$\Delta^n y_{n+0}$
$z_1$	$y_1$	$\Delta^1 y_2$	$\Delta^2 y_3$	$\Delta^3 y_4$	$\Delta^4 y_5$	$\Delta^5 y_6$	$\Delta^5 y_7$	...	$\Delta^n y_{n+1}$
$x_2$	$y_2$	$\Delta^1 y_3$	$\Delta^2 y_4$	$\Delta^3 y_5$	$\Delta^4 y_6$	$\Delta^5 y_7$	$\Delta^6 y_8$	...	$\Delta^n y_{n+2}$
...	...	...	...	...	...	...	...	...	...
$z_m$	$y_m$	$\Delta^1 y_{m+1}$	$\Delta^2 y_{m+2}$	$\Delta^3 y_{m+3}$	$\Delta^4 y_{m+4}$	$\Delta^5 y_{m+5}$	$\Delta^5 y_{m+5}$	...	$\Delta^n y_{n+n}$

## 2.2.5 Интерполяционная формула Бесселя.[И. Ф. Б.]

- И.Ф. Б. берется  $2n + 2$  равноотстоящих узла.

$$x_{-n}, x_{-(n-1)}, \dots, x_0, x_1, \dots, x_n, x_{n+1};$$

- $h$  - шаг.
- Известны значения функции в узлах:  $y_i = f(x_i)$ ;
- Требуется построить интерполяционный полином  $P_n(x_i) = y_i$ , причем этот полином не выше степени  $2n + 1$ . (Потому, что у нас  $2n + 2$  узел).

$$\text{Откуда: } \Delta^k P_n(x_i) = \Delta^k y_i;$$

Распишем полином:

$$\begin{aligned}
 P_n(x) &= a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_{-1})(x - x_0)(x - x_1) + \dots \\
 &\quad + a_{2n-1}(x - x_{-(n-1)}) \cdot \dots \cdot (x - x_{-1})(x - x_0)(x - x_1) \cdot \dots \cdot (x - x_{n-1}) + \\
 &\quad + a_{2n}(x - x_{-(n-1)}) \cdot \dots \cdot (x - x_{-1})(x - x_0)(x - x_1) \cdot \dots \cdot (x - x_n) \\
 &= a_0 + a_1(x - x_0)^{[1]} + a_2(x - x_0)^{[2]} + a_3(x - x_{n-1})^{[3]} \\
 &\quad + \dots + a_{2n}(x - x_{-(n-1)})^{[2n]};
 \end{aligned}$$



$$\begin{aligned}
P_n(x) = & y_0 + q\Delta y_{-1} + \frac{(q+1)q}{2!}\Delta^2 y_{-1} + \frac{(q+1)q(q-1)}{3!}\Delta^3 y_{-2} \\
& + \frac{(q+2)(q+1)q(q-1)}{4!}\Delta^4 y_{-2} + \dots + \\
& + \frac{(q+n-1)\dots(q+1)q(q-1)(q-2)\dots(q-n+1)}{(2n-1)!}\Delta^{2n-1} y_{-n} \\
& + \frac{(q+n)\dots(q+1)q(q-1)\dots(q-n+1)}{(2n)!}\Delta^{2n} y_{-n}; - II \text{ ИФГ}
\end{aligned}$$

Возьмем за начальное значение не  $x = x_0; y = y_0$ , а  $x = x_1; y = y_1$ .

Получим:

$$\frac{x-x_1}{h} = \frac{x-(x_0+h)}{h} = q-1; \text{ т. е. в полином подставляем не } q, \text{ а } q-1.$$

Все индексы возрастают на единицу (+1). Получим вспомогательную интерполяционную формулу, которая примет следующий вид:

$$\begin{aligned}
P_n(x) = & y_1 + (q-1)\Delta y_0 + \frac{q(q-1)}{2!}\Delta^2 y_0 \\
& + \frac{q(q-1)(q-2)}{3!}\Delta^3 y_{-1} + \frac{(q+1)q(q-1)(q-2)}{4!}\Delta^4 y_{-1} + \dots \\
& + \frac{(q+n-2)\dots(q-n)}{(2n-1)!}\Delta^{2n-1} y_{-(n-1)} + \frac{(q+n-1)\dots(q-n)}{(2n)!}\Delta^{2n} y_{-(n-1)};
\end{aligned}$$

- Вспомогательный интерполяционный формула Бесселя[В ИФБ].

Если теперь взять среднее арифметическое от предыдущего полинома и вспомогательного, то получим интерполяционный многочлен Бесселя.

$$[\text{И. Ф. Б.}] = \frac{[\text{II И. Ф. Г}] + [\text{В И. Ф. Б.}]}{2};$$

Причем, это будет полином, который совпадает с  $2n+2$  Точек(узлов).

$$\begin{aligned}
P_n(x) = & \frac{y_0 + y_1}{2} + \left(q - \frac{1}{2}\right)\Delta y_0 + \frac{q(q-1)}{2!} \cdot \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} \\
& + \frac{\left(q - \frac{1}{2}\right)q(q-1)}{3!}\Delta^3 y_{-1} + \frac{q(q-1)(q+1)(q-2)}{4!} \cdot \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \dots + \\
& + \frac{q(q-1)(q+1)\dots(q-n)(q+n-1)}{(2n)!} \cdot \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n-1)}}{2} \\
& + \frac{\left(q - \frac{1}{2}\right)q(q-1)\dots(q-n)(q+n-1)}{(2n+1)!}\Delta^{2n+1} y_{-n};
\end{aligned}$$

- Интерполяционный полином Бесселя.

$$q = \frac{x-x_0}{h}; - \text{Стандартный вид.}$$

Для формулы Бесселя можно заметить, что коэффициенты нечетного порядка имеют коэффициент:

$$\left(q - \frac{1}{2}\right).$$

Если  $q = \frac{1}{2}$ , то вычисления значительно упрощаются.

$$\begin{aligned} P_n\left(\frac{x_0 + x_1}{2}\right) &= \frac{y_0 + y_1}{2} + \left(q - \frac{1}{2}\right) \cdot \Delta y_0 + \frac{1}{8} \cdot \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + \frac{\left(q - \frac{1}{2}\right) q(q-1)}{3!} \Delta^3 y_{-1} \\ &\quad + \frac{q(q-1)(q+1)(q-2)}{4!} \cdot \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \dots \\ &\quad + \frac{q(q-1)(q+1) \cdot \dots \cdot (q-n)(q+n-1)}{(2n)!} \cdot \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n-1)}}{2} \\ &\quad + \frac{\left(q - \frac{1}{2}\right) q(q-1) \cdot \dots \cdot (q-n)(q+n-1)}{(2n+1)!} \Delta^{2n+1} y_{-n} \\ &= \frac{y_0 + y_1}{2} + \frac{1}{8} \cdot \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + \frac{3}{128} \cdot \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} - \frac{5}{1024} \cdot \frac{\Delta^6 y_{-3} + \Delta^6 y_{-2}}{2} \\ &\quad + \dots + (-1)^n \frac{[(2n-1)!!]^2}{2^{2n}} \cdot \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n+1)}}{2}; \end{aligned}$$

–Интерполяционная формула Бесселя.

• Интерполяционная формула Бесселя использует следующие конечные разности:

$$\begin{aligned} \frac{y_0 + y_1}{2}, \Delta y_0, \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2}, \Delta^3 y_{-1}, \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2}, \Delta^5 y_{-2}, \dots, \\ \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n-1)}}{2}, \Delta^{2n+1} y_{-n+1}; \end{aligned}$$

Таблица выделенных элементов по И. Ф. Б.:

$\Delta^n y_i$ $z_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$\Delta^4 y_i$	$\Delta^5 y_i$	$\Delta^6 y_i$	...	$\Delta^n y_i$
...	...	...	...	...	...	...	...	...	...
$z_{-9}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-3}$	...	$\Delta^n y_{n-9}$
$z_{-8}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-2}$	...	$\Delta^n y_{n-8}$
$z_{-7}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-2}$	$\Delta^5 y_{-1}$	...	$\Delta^n y_{n-7}$
$z_{-6}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-2}$	$\Delta^5 y_{-1}$	$\Delta^5 y_0$	...	$\Delta^n y_{n-6}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-1}$	$\Delta^5 y_0$	$\Delta^5 y_1$	...	$\Delta^n y_{n-5}$
$z_{-4}$	$y_{-4}$	$\Delta^1 y_{-3}$	$\Delta^2 y_{-2}$	$\Delta^3 y_{-1}$	$\Delta^4 y_0$	$\Delta^5 y_1$	$\Delta^5 y_2$	...	$\Delta^n y_{n-4}$
$z_{-3}$	$y_{-3}$	$\Delta^1 y_{-2}$	$\Delta^2 y_{-1}$	$\Delta^3 y_0$	$\Delta^4 y_1$	$\Delta^5 y_2$	$\Delta^5 y_3$	...	$\Delta^n y_{n-3}$
$z_{-2}$	$y_{-2}$	$\Delta^1 y_{-1}$	$\Delta^2 y_0$	$\Delta^3 y_1$	$\Delta^4 y_2$	$\Delta^5 y_3$	$\Delta^5 y_4$	...	$\Delta^n y_{n-2}$
$z_{-1}$	$y_{-1}$	$\Delta^1 y_0$	$\Delta^2 y_1$	$\Delta^3 y_2$	$\Delta^4 y_3$	$\Delta^5 y_4$	$\Delta^5 y_5$	...	$\Delta^n y_{n-1}$
$z_0$	$y_0$	$\Delta^1 y_1$	$\Delta^2 y_2$	$\Delta^3 y_3$	$\Delta^4 y_4$	$\Delta^5 y_5$	$\Delta^5 y_6$	...	$\Delta^n y_{n+0}$
$z_1$	$y_1$	$\Delta^1 y_2$	$\Delta^2 y_3$	$\Delta^3 y_4$	$\Delta^4 y_5$	$\Delta^5 y_6$	$\Delta^5 y_7$	...	$\Delta^n y_{n+1}$
$x_2$	$y_2$	$\Delta^1 y_3$	$\Delta^2 y_4$	$\Delta^3 y_5$	$\Delta^4 y_6$	$\Delta^5 y_7$	$\Delta^6 y_8$	...	$\Delta^n y_{n+2}$

...	...	...	...	...	...	...	...	...	...
$z_m$	$y_m$	$\Delta^1 y_{m+1}$	$\Delta^2 y_{m+2}$	$\Delta^3 y_{m+3}$	$\Delta^4 y_{m+4}$	$\Delta^5 y_{m+5}$	$\Delta^5 y_{m+5}$	...	$\Delta^n y_{n+n}$

Так же для общего случая можно тоже упростить формулу, а именно:

Сделаем замену:  $p = q - \frac{1}{2} = \frac{1}{h} \cdot \left( x - \frac{x_0 + x_1}{2} \right)$ ;

$$\begin{aligned}
 P_n(x) = & \frac{y_0 + y_1}{2} + p\Delta y_0 + \frac{\left(p^2 - \frac{1}{4}\right)}{2!} \cdot \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + \frac{p\left(p^2 - \frac{1}{4}\right)}{3!} \Delta^3 y_{-1} + \frac{\left(p^2 - \frac{1}{4}\right)\left(p^2 - \frac{9}{4}\right)}{4!} \\
 & \cdot \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \frac{p\left(p^2 - \frac{1}{4}\right)\left(p^2 - \frac{9}{4}\right)}{5!} \cdot \Delta^5 y_{-2} + \dots \\
 & + \frac{\left(p^2 - \frac{1}{4}\right)\left(p^2 - \frac{9}{4}\right) \cdot \dots \cdot \left(p^2 - \frac{(2n-1)^2}{4}\right)}{(2n)!} \cdot \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n-1)}}{2} \\
 & + \frac{p\left(p^2 - \frac{1}{4}\right)\left(p^2 - \frac{9}{4}\right) \cdot \dots \cdot \left(p^2 - \frac{(2n-1)^2}{4}\right)}{(2n+1)!} \cdot \Delta^{2n+1} y_{-n+1};
 \end{aligned}$$

- Интерполяционный полином Бесселя. [Упрощенная].

• [Упрощенная] Интерполяционная формула Бесселя использует следующие конечные разности:

$$\begin{aligned}
 & \frac{y_0 + y_1}{2}, \Delta y_0, \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2}, \Delta^3 y_{-1}, \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2}, \Delta^5 y_{-2}, \dots, \\
 & , \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n-1)}}{2}, \Delta^{2n+1} y_{-n+1};
 \end{aligned}$$

Таблица выделенных элементов по И. Ф. Б.:

$\Delta^n y_i$ $z_i$	$y_i$	$\Delta^1 y_i$	$\Delta^2 y_i$	$\Delta^3 y_i$	$\Delta^4 y_i$	$\Delta^5 y_i$	$\Delta^6 y_i$	...	$\Delta^n y_i$
...	...	...	...	...	...	...	...	...	...
$z_{-9}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-3}$	...	$\Delta^n y_{n-9}$
$z_{-8}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-2}$	...	$\Delta^n y_{n-8}$
$z_{-7}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-2}$	$\Delta^5 y_{-1}$	...	$\Delta^n y_{n-7}$
$z_{-6}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-2}$	$\Delta^5 y_{-1}$	$\Delta^5 y_0$	...	$\Delta^n y_{n-6}$
$z_{-5}$	$y_{-5}$	$\Delta^1 y_{-4}$	$\Delta^2 y_{-3}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-1}$	$\Delta^5 y_0$	$\Delta^5 y_1$	...	$\Delta^n y_{n-5}$
$z_{-4}$	$y_{-4}$	$\Delta^1 y_{-3}$	$\Delta^2 y_{-2}$	$\Delta^3 y_{-1}$	$\Delta^4 y_0$	$\Delta^5 y_1$	$\Delta^5 y_2$	...	$\Delta^n y_{n-4}$
$z_{-3}$	$y_{-3}$	$\Delta^1 y_{-2}$	$\Delta^2 y_{-1}$	$\Delta^3 y_0$	$\Delta^4 y_1$	$\Delta^5 y_2$	$\Delta^5 y_3$	...	$\Delta^n y_{n-3}$
$z_{-2}$	$y_{-2}$	$\Delta^1 y_{-1}$	$\Delta^2 y_0$	$\Delta^3 y_1$	$\Delta^4 y_2$	$\Delta^5 y_3$	$\Delta^5 y_4$	...	$\Delta^n y_{n-2}$
$z_{-1}$	$y_{-1}$	$\Delta^1 y_0$	$\Delta^2 y_1$	$\Delta^3 y_2$	$\Delta^4 y_3$	$\Delta^5 y_4$	$\Delta^5 y_5$	...	$\Delta^n y_{n-1}$
$z_0$	$y_0$	$\Delta^1 y_1$	$\Delta^2 y_2$	$\Delta^3 y_3$	$\Delta^4 y_4$	$\Delta^5 y_5$	$\Delta^5 y_6$	...	$\Delta^n y_{n+0}$
$z_1$	$y_1$	$\Delta^1 y_2$	$\Delta^2 y_3$	$\Delta^3 y_4$	$\Delta^4 y_5$	$\Delta^5 y_6$	$\Delta^5 y_7$	...	$\Delta^n y_{n+1}$
$x_2$	$y_2$	$\Delta^1 y_3$	$\Delta^2 y_4$	$\Delta^3 y_5$	$\Delta^4 y_6$	$\Delta^5 y_7$	$\Delta^6 y_8$	...	$\Delta^n y_{n+2}$
...	...	...	...	...	...	...	...	...	...
$z_m$	$y_m$	$\Delta^1 y_{m+1}$	$\Delta^2 y_{m+2}$	$\Delta^3 y_{m+3}$	$\Delta^4 y_{m+4}$	$\Delta^5 y_{m+5}$	$\Delta^5 y_{m+5}$	...	$\Delta^n y_{n+n}$

## 2.2.6 Особенности использования интерполяционных формул Стирлинга и Бесселя. + Общая таблица использования Интерполяционных формул для равноотстоящих узлов.

$$q = \frac{x - x_0}{h};$$

Если  $|q| \leq \frac{1}{4}$ ; то тогда применяется формула Стирлинга.

Если  $\frac{1}{4} \leq |q| \leq \frac{3}{4}$ ; то тогда применяется формула Бесселя.

Общая таблица использования Интерполяционных формул:

№	Формула	Условия использования	Примечания
1	I И. Ф. Н.	Интерполирование в начале таблицы.	$q = \frac{x - x_0}{h}$
2	II И. Ф. Н.	Интерполирование в конце таблицы.	$q = \frac{x - x_n}{h}$
3	I И. Ф. Г.	Используется в И. Ф. С.	$q = \frac{x - x_0}{h}$
4	II И. Ф. Г.	Используется в И. Ф. С.	
5	И. Ф. С.	$q \leq 0.25 = \left[\frac{1}{4}\right];$	
6	И. Ф. Б.	$\left[\frac{1}{4}\right] = 0.25 < q < 0.75 = \left[\frac{3}{4}\right];$	

## 2.3 Практическая часть. (Результаты работы программы.)

**Принятые данные :**

**A:**

x	y
1.50	15.132
1.55	17.422
1.60	20.393
1.65	23.994
1.70	28.160
1.75	32.812
1.80	37.857
1.85	43.189
1.90	48.689
1.95	54.225
2.00	59.158
2.05	64.817
2.10	69.550
2.15	74.782
2.20	79.548

**x\_G, x\_S, x\_B :**

1.672	1.749	1.866
-------	-------	-------

Точность :  $\varepsilon = 0.00001$ ;

**Вычисления :**

**newVec :**

1.500	15.132
1.550	17.422
1.600	20.393
1.650	23.994
1.700	28.160
1.750	32.812
1.800	37.857
1.850	43.189
1.900	48.689
1.950	54.225
2.000	59.158
2.050	64.817
2.100	69.550
2.150	74.782
2.200	79.548

**Вычисления :**

**newVec :**

1.500	15.132	2.290	0.681	-0.051	-0.014	-0.000	0.000	0.001	-0.003	0.006	-0.504	5.454	-32.669	142.642	-508.333
1.550	17.422	2.971	0.630	-0.065	-0.014	0.000	0.001	-0.002	0.003	-0.498	4.950	-27.215	109.973	-365.691	0
1.600	20.393	3.601	0.565	-0.079	-0.014	0.001	-0.001	0.001	-0.495	4.452	-22.265	82.758	-255.718	0	0
1.650	23.994	4.166	0.486	-0.093	-0.013	0.000	-0.000	-0.494	3.957	-17.813	60.493	-172.960	0	0	0
1.700	28.160	4.652	0.393	-0.106	-0.013	-0.000	-0.494	3.463	-13.856	42.680	-112.467	0	0	0	0
1.750	32.812	5.045	0.287	-0.119	-0.013	-0.494	2.969	-10.393	28.824	-69.787	0	0	0	0	0
1.800	37.857	5.332	0.168	-0.132	-0.507	2.475	-7.424	18.431	-40.963	0	0	0	0	0	0
1.850	43.189	5.500	0.036	-0.639	1.968	-4.949	11.007	-22.532	0	0	0	0	0	0	0
1.900	48.689	5.536	-0.603	1.329	-2.981	6.058	-11.525	0	0	0	0	0	0	0	0
1.950	54.225	4.933	0.726	-1.652	3.077	-5.467	0	0	0	0	0	0	0	0	0
2.000	59.158	5.659	-0.926	1.425	-2.390	0	0	0	0	0	0	0	0	0	0
2.050	64.817	4.733	0.499	-0.965	0	0	0	0	0	0	0	0	0	0	0
2.100	69.550	5.232	-0.466	0	0	0	0	0	0	0	0	0	0	0	0
2.150	74.782	4.766	0	0	0	0	0	0	0	0	0	0	0	0	0
2.200	79.548	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Предварительное выделение элементов для 1 И. Ф. Г. :**

**newVec :**

1.500	15.132	2.290	0.681	-0.051	-0.014	-0.000	0.000	0.001	-0.003	0.006	-0.504	5.454	-32.669	142.642	-508.333
1.550	17.422	2.971	0.630	-0.065	-0.014	0.000	0.001	-0.002	0.003	-0.498	4.950	-27.215	109.973	-365.691	0
1.600	20.393	3.601	0.565	-0.079	-0.014	0.001	-0.001	0.001	-0.495	4.452	-22.265	82.758	-255.718	0	0
1.650	23.994	4.166	0.486	-0.093	-0.013	0.000	-0.000	-0.494	3.957	-17.813	60.493	-172.960	0	0	0
1.700	28.160	4.652	0.393	-0.106	-0.013	-0.000	-0.494	3.463	-13.856	42.680	-112.467	0	0	0	0
1.750	32.812	5.045	0.287	-0.119	-0.013	-0.494	2.969	-10.393	28.824	-69.787	0	0	0	0	0
1.800	37.857	5.332	0.168	-0.132	-0.507	2.475	-7.424	18.431	-40.963	0	0	0	0	0	0
1.850	43.189	5.500	0.036	-0.639	1.968	-4.949	11.007	-22.532	0	0	0	0	0	0	0
1.900	48.689	5.536	-0.603	1.329	-2.981	6.058	-11.525	0	0	0	0	0	0	0	0
1.950	54.225	4.933	0.726	-1.652	3.077	-5.467	0	0	0	0	0	0	0	0	0
2.000	59.158	5.659	-0.926	1.425	-2.390	0	0	0	0	0	0	0	0	0	0
2.050	64.817	4.733	0.499	-0.965	0	0	0	0	0	0	0	0	0	0	0
2.100	69.550	5.232	-0.466	0	0	0	0	0	0	0	0	0	0	0	0
2.150	74.782	4.766	0	0	0	0	0	0	0	0	0	0	0	0	0
2.200	79.548	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Предварительное выделение элементов для И. Ф. С. :**

**newVec :**

1.500	15.132	2.290	0.681	-0.051	-0.014	-0.000	0.000	0.001	-0.003	0.006	-0.504	5.454	-32.669	142.642	-508.333
1.550	17.422	2.971	0.630	-0.065	-0.014	0.000	0.001	-0.002	0.003	-0.498	4.950	-27.215	109.973	-365.691	0
1.600	20.393	3.601	0.565	-0.079	-0.014	0.001	-0.001	0.001	-0.495	4.452	-22.265	82.758	-255.718	0	0
1.650	23.994	4.166	0.486	-0.093	-0.013	0.000	-0.000	-0.494	3.957	-17.813	60.493	-172.960	0	0	0
1.700	28.160	4.652	0.393	-0.106	-0.013	-0.000	-0.494	3.463	-13.856	42.680	-112.467	0	0	0	0
1.750	32.812	5.045	0.287	-0.119	-0.013	-0.494	2.969	-10.393	28.824	-69.787	0	0	0	0	0
1.800	37.857	5.332	0.168	-0.132	-0.507	2.475	-7.424	18.431	-40.963	0	0	0	0	0	0
1.850	43.189	5.500	0.036	-0.639	1.968	-4.949	11.007	-22.532	0	0	0	0	0	0	0
1.900	48.689	5.536	-0.603	1.329	-2.981	6.058	-11.525	0	0	0	0	0	0	0	0
1.950	54.225	4.933	0.726	-1.652	3.077	-5.467	0	0	0	0	0	0	0	0	0
2.000	59.158	5.659	-0.926	1.425	-2.390	0	0	0	0	0	0	0	0	0	0
2.050	64.817	4.733	0.499	-0.965	0	0	0	0	0	0	0	0	0	0	0
2.100	69.550	5.232	-0.466	0	0	0	0	0	0	0	0	0	0	0	0
2.150	74.782	4.766	0	0	0	0	0	0	0	0	0	0	0	0	0
2.200	79.548	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Вычисления :**

$x = 1.749$ ;  $x_0 = 1.7$ ;  $h = 0.05$ ;  $q = (x - x_0) / h = (1.749 - 1.7) / 0.05 = 0.98$ ;

**Вычисления :**

$P_n(x) = P_6(x = 1.749) = 31.855051$  ;

**Предварительное выделение элементов для И. Ф. В. :**

**newVec :**

1.500	15.132	2.290	0.681	-0.051	-0.014	-0.000	0.000	0.001	-0.003	0.006	-0.504	5.454	-32.669	142.642	-508.333
1.550	17.422	2.971	0.630	-0.065	-0.014	0.000	0.001	-0.002	0.003	-0.498	4.950	-27.215	109.973	-365.691	0
1.600	20.393	3.601	0.565	-0.079	-0.014	0.001	-0.001	0.001	-0.495	4.452	-22.265	82.758	-255.718	0	0
1.650	23.994	4.166	0.486	-0.093	-0.013	0.000	-0.000	-0.494	3.957	-17.813	60.493	-172.960	0	0	0
1.700	28.160	4.652	0.393	-0.106	-0.013	-0.000	-0.494	3.463	-13.856	42.680	-112.467	0	0	0	0
1.750	32.812	5.045	0.287	-0.119	-0.013	-0.494	2.969	-10.393	28.824	-69.787	0	0	0	0	0
1.800	37.857	5.332	0.168	-0.132	-0.507	2.475	-7.424	18.431	-40.963	0	0	0	0	0	0
1.850	43.189	5.500	0.036	-0.639	1.968	-4.949	11.007	-22.532	0	0	0	0	0	0	0
1.900	48.689	5.536	-0.603	1.329	-2.981	6.058	-11.525	0	0	0	0	0	0	0	0
1.950	54.225	4.933	0.726	-1.652	3.077	-5.467	0	0	0	0	0	0	0	0	0
2.000	59.158	5.659	-0.926	1.425	-2.390	0	0	0	0	0	0	0	0	0	0
2.050	64.817	4.733	0.499	-0.965	0	0	0	0	0	0	0	0	0	0	0
2.100	69.550	5.232	-0.466	0	0	0	0	0	0	0	0	0	0	0	0
2.150	74.782	4.766	0	0	0	0	0	0	0	0	0	0	0	0	0
2.200	79.548	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Вычисления :**

$x = 1.866$ ;  $x_0 = 1.85$ ;  $h = 0.05$ ;  $q = (x - x_0) / h = (1.866 - 1.85) / 0.05 = 0.32$ ;

**Вычисления :**

$P_n(x) = P_7(x = 1.866) = 46.579585$  ;

---

**Время работы: 1.072 ;**

## 2.4 Приложение № 1 – Примеры работы программы.

MainWindow

Загрузка данных

Введите количество известных узлов таблицы: 15

Вычисление

Точность  $\epsilon =$  0.00001

$X_G, X_S, X_B =$  1.672 1.749 1.866

Таблица значений функции для варианта № 12

x	y
1.50	15.132
1.55	17.422
1.60	20.393
1.65	23.994
1.70	28.160
1.75	32.812
1.80	37.857
1.85	43.189
1.90	48.689
1.95	54.225
2.00	59.158
2.05	64.817
2.10	69.550
2.15	74.782
2.20	79.548

Принятые данные:

x

y

1.500 15.132

1.550 17.422

1.600 20.393

1.650 23.994

1.700 28.160

1.750 32.812

1.800 37.857

1.850 43.189

1.900 48.689

1.950 54.225

2.000 59.158

2.050 64.817

2.100 69.550

2.150 74.782

2.200 79.548

$X_G, X_S, X_B:$

1.672 1.749 1.866

Точность  $\epsilon = 0.00001$ ;

Вычисления:

newVec:

1.500 15.132 2.290 0.681 -0.051 -0.014 -0.000 0.000 0.001 -0.003 0.006 -0.504 5.454 -32.669 142.642 -508.333

1.550 17.422 2.971 0.630 -0.065 -0.014 0.000 0.001 -0.002 0.003 -0.498 4.950 -27.215 109.973 -365.691 0

1.600 20.393 3.601 0.565 -0.079 -0.014 0.001 -0.001 0.001 -0.495 4.452 -22.265 82.758 -255.718 0 0

1.650 23.994 4.166 0.486 -0.093 -0.013 0.000 -0.000 -0.494 3.957 -17.813 60.493 -172.960 0 0 0

1.700 28.160 4.652 0.393 -0.106 -0.013 -0.000 -0.494 3.463 -13.856 42.680 -112.467 0 0 0 0

1.750 32.812 5.045 0.287 -0.119 -0.013 -0.494 2.969 -10.393 28.824 -69.787 0 0 0 0 0

1.800 37.857 5.332 0.168 -0.132 -0.507 2.475 -7.424 18.431 -40.963 0 0 0 0 0 0

1.850 43.189 5.500 0.036 -0.639 1.968 -4.949 11.007 -22.532 0 0 0 0 0 0 0

1.900 48.689 5.536 -0.603 1.329 -2.981 6.058 -11.525 0 0 0 0 0 0 0 0

1.950 54.225 4.933 0.726 -1.652 3.077 -5.467 0 0 0 0 0 0 0 0 0

2.000 59.158 5.659 -0.926 1.425 -2.390 0 0 0 0 0 0 0 0 0 0

2.050 64.817 4.733 0.499 -0.965 0 0 0 0 0 0 0 0 0 0 0

2.100 69.550 5.232 -0.466 0 0 0 0 0 0 0 0 0 0 0 0

2.150 74.782 4.766 0 0 0 0 0 0 0 0 0 0 0 0 0

2.200 79.548 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Предварительное выделение элементов для 1 И. Ф. Г.:

newVec:

1.500 15.132 2.290 0.681 -0.051 -0.014 -0.000 0.000 0.001 -0.003 0.006 -0.504 5.454 -32.669 142.642 -508.333

1.550 17.422 2.971 0.630 -0.065 -0.014 0.000 0.001 -0.002 0.003 -0.498 4.950 -27.215 109.973 -365.691 0

1.600 20.393 3.601 0.565 -0.079 -0.014 0.001 -0.001 0.001 -0.495 4.452 -22.265 82.758 -255.718 0 0

1.650 23.994 4.166 0.486 -0.093 -0.013 0.000 -0.000 -0.494 3.957 -17.813 60.493 -172.960 0 0 0

1.700 28.160 4.652 0.393 -0.106 -0.013 -0.000 -0.494 3.463 -13.856 42.680 -112.467 0 0 0 0

1.750 32.812 5.045 0.287 -0.119 -0.013 -0.494 2.969 -10.393 28.824 -69.787 0 0 0 0 0

1.800 37.857 5.332 0.168 -0.132 -0.507 2.475 -7.424 18.431 -40.963 0 0 0 0 0 0

1.850 43.189 5.500 0.036 -0.639 1.968 -4.949 11.007 -22.532 0 0 0 0 0 0 0

1.900 48.689 5.536 -0.603 1.329 -2.981 6.058 -11.525 0 0 0 0 0 0 0 0

1.950 54.225 4.933 0.726 -1.652 3.077 -5.467 0 0 0 0 0 0 0 0 0

2.000 59.158 5.659 -0.926 1.425 -2.390 0 0 0 0 0 0 0 0 0 0

2.050 64.817 4.733 0.499 -0.965 0 0 0 0 0 0 0 0 0 0 0

2.100 69.550 5.232 -0.466 0 0 0 0 0 0 0 0 0 0 0 0

2.150 74.782 4.766 0 0 0 0 0 0 0 0 0 0 0 0 0

2.200 79.548 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Предварительное выделение элементов для И. Ф. С.:

newVec:

1.500 15.132 2.290 0.681 -0.051 -0.014 -0.000 0.000 0.001 -0.003 0.006 -0.504 5.454 -32.669 142.642 -508.333

1.550 17.422 2.971 0.630 -0.065 -0.014 0.000 0.001 -0.002 0.003 -0.498 4.950 -27.215 109.973 -365.691 0

1.600 20.393 3.601 0.565 -0.079 -0.014 0.001 -0.001 0.001 -0.495 4.452 -22.265 82.758 -255.718 0 0

1.650 23.994 4.166 0.486 -0.093 -0.013 0.000 -0.000 -0.494 3.957 -17.813 60.493 -172.960 0 0 0

1.700 28.160 4.652 0.393 -0.106 -0.013 -0.000 -0.494 3.463 -13.856 42.680 -112.467 0 0 0 0

1.750 32.812 5.045 0.287 -0.119 -0.013 -0.494 2.969 -10.393 28.824 -69.787 0 0 0 0 0

1.800 37.857 5.332 0.168 -0.132 -0.507 2.475 -7.424 18.431 -40.963 0 0 0 0 0 0

1.850 43.189 5.500 0.036 -0.639 1.968 -4.949 11.007 -22.532 0 0 0 0 0 0 0

1.900 48.689 5.536 -0.603 1.329 -2.981 6.058 -11.525 0 0 0 0 0 0 0 0

1.950 54.225 4.933 0.726 -1.652 3.077 -5.467 0 0 0 0 0 0 0 0 0

2.000 59.158 5.659 -0.926 1.425 -2.390 0 0 0 0 0 0 0 0 0 0

2.050 64.817 4.733 0.499 -0.965 0 0 0 0 0 0 0 0 0 0 0

2.100 69.550 5.232 -0.466 0 0 0 0 0 0 0 0 0 0 0 0

2.150 74.782 4.766 0 0 0 0 0 0 0 0 0 0 0 0 0

2.200 79.548 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Вычисления:

$x = 1.749, x_0 = 1.7; h = 0.05; q = (x - x_0) / h = (1.749 - 1.7) / 0.05 = 0.98;$

Вычисления:

$P(x) = P(x = 1.749) = 31.855051;$

Предварительное выделение элементов для И. Ф. В.:

newVec:

1.500 15.132 2.290 0.681 -0.051 -0.014 -0.000 0.000 0.001 -0.003 0.006 -0.504 5.454 -32.669 142.642 -508.333

1.550 17.422 2.971 0.630 -0.065 -0.014 0.000 0.001 -0.002 0.003 -0.498 4.950 -27.215 109.973 -365.691 0

1.600 20.393 3.601 0.565 -0.079 -0.014 0.001 -0.001 0.001 -0.495 4.452 -22.265 82.758 -255.718 0 0

1.650 23.994 4.166 0.486 -0.093 -0.013 0.000 -0.000 -0.494 3.957 -17.813 60.493 -172.960 0 0 0

1.700 28.160 4.652 0.393 -0.106 -0.013 -0.000 -0.494 3.463 -13.856 42.680 -112.467 0 0 0 0

1.750 32.812 5.045 0.287 -0.119 -0.013 -0.494 2.969 -10.393 28.824 -69.787 0 0 0 0 0

1.800 37.857 5.332 0.168 -0.132 -0.507 2.475 -7.424 18.431 -40.963 0 0 0 0 0 0

1.850 43.189 5.500 0.036 -0.639 1.968 -4.949 11.007 -22.532 0 0 0 0 0 0 0

1.900 48.689 5.536 -0.603 1.329 -2.981 6.058 -11.525 0 0 0 0 0 0 0 0

1.950 54.225 4.933 0.726 -1.652 3.077 -5.467 0 0 0 0 0 0 0 0 0

2.000 59.158 5.659 -0.926 1.425 -2.390 0 0 0 0 0 0 0 0 0 0

2.050 64.817 4.733 0.499 -0.965 0 0 0 0 0 0 0 0 0 0 0

2.100 69.550 5.232 -0.466 0 0 0 0 0 0 0 0 0 0 0 0

2.150 74.782 4.766 0 0 0 0 0 0 0 0 0 0 0 0 0

2.200 79.548 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Вычисления:

$x = 1.866, x_0 = 1.85; h = 0.05; q = (x - x_0) / h = (1.866 - 1.85) / 0.05 = 0.32;$

Вычисления:

$P(x) = P(x = 1.866) = 46.579585;$

Время работы: 0.525;

## 2.5 Приложение № 2 – Листинг программы.

```
int Ga, Gb;
double step_a, step_b;

auto print_general = [&](QVector<QVector<double>>& vec) -> void {
    strHtml_A.clear();
    strHtml_A.append("<b style=\"color: red;\">Вычисления </b><b style=\"color: black;\">:</b><br>");
    strHtml_A.append("<b>newVec :</b><table border=1 style=\"border-collapse: collapse;\">");
    foreach(auto str, vec){
        strHtml_A.append("<tr>");
        foreach(auto elem, str){
            elem == 0 ? strHtml_A.append("<td align=center> " +
QString::number(elem) + " </td>")
                : strHtml_A.append("<td align=center> " +
QString::number(elem,'f',3) + " </td>");
        }
        strHtml_A.append("</tr>");
    }
    strHtml_A.append("</table>");
    ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
"<p></p>");
};

auto print_general_G_1 = [&](QVector<QVector<double>>& vec, size_t Ga,
QVector<double>& c) -> void {
    strHtml_A.clear();
    strHtml_A.append("<b style=\"color: green;\">Предварительное выделение
элементов для 1 И. Ф. Г. </b><b style=\"color: black;\">:</b><br>");
    strHtml_A.append("<b>newVec :</b><table border=1 style=\"border-collapse: collapse;\">");

    for(size_t i = 0; i < CN; i++){
        strHtml_A.append("<tr>");
        for(size_t j = 0; j < vec[i].count(); j++){
            if((j == (2*Ga + 1) - 2*i || j == (2*Ga + 2) - 2*i) && (i <= Ga)) ||
(i == Ga && (j == 0 || j == 1)) ){
                vec[i][j] == 0 ? strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 0) + "</b></td>")
                    : strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 3) + "</b></td>");
                if (j != 0 && i == 0){
                    c.push_front(vec[i][j]);
                }else if(i > 0 && j != 0){
                    static QVector<double> st;
                    st.push_front(vec[i][j]);
                    if(st.count() == 2){
                        std::copy(st.begin(), st.end(), std::back_inserter(c));
                        st.clear();
                    }
                }
            }else{
                vec[i][j] == 0 ? strHtml_A.append("<td align=center> " +
QString::number(vec[i][j]) + " </td>")
                    : strHtml_A.append("<td align=center> " +
QString::number(vec[i][j], 'f', 3) + " </td>");
            }
        }
        strHtml_A.append("</tr>");
    }
    strHtml_A.append("</table>");
    ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
"<p></p>");
};
```



```

    auto print_general_S = [&](QVector<QVector<double>>& vec, size_t Ga,
QVector<double>& c) -> void {
        strHtml_A.clear();
        strHtml_A.append("<b style=\"color: green;\">Предварительное выделение
элементов для И. Ф. С. </b><b style=\"color: black;\">:</b><br>");
        strHtml_A.append("<b>newVec :</b><table border=1 style=\"border-collapse:
collapse;\">");
        for(size_t i = 0; i < CN; i++){
            strHtml_A.append("<tr>");
            for(size_t j = 0; j < vec[i].count(); j++){
                if(((j % 2 == 0) && (j != 0)) && ( i == (Ga + 1 - 3*j/2) || i == (2*Ga
+ 4 - 3*j)/2 )) || ((j % 2 == 1) && ( i == (2*Ga+3*(1 - j))/2 ))){
                    vec[i][j] == 0 ? strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 0) + "</b></td>")
                    : strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 3) + "</b></td>");
                    if (j != 0 && i == 0) { c.push_back(vec[i][j]);}
                    }else if(i > 0 && j != 0) { c.push_back(vec[i][j]);}
                    }else{
                        (i == Ga && j == 0)
                        ? strHtml_A.append("<td align=center><b style=\"color:
DarkCyan;\"> " + QString::number(vec[i][j], 'f', 3) + "</b></td>")
                        : vec[i][j] == 0
                        ? strHtml_A.append("<td align=center> " +
QString::number(vec[i][j]) + " </td>")
                        : strHtml_A.append("<td align=center> " +
QString::number(vec[i][j], 'f', 3) + " </td>");}}
                    strHtml_A.append("</tr>");}
            strHtml_A.append("</table>");
            ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
"<p></p>");
        };
        auto print_general_B_1 = [&](QVector<QVector<double>>& vec, size_t Ga,
QVector<double>& c) -> void {
            strHtml_A.clear();
            strHtml_A.append("<b style=\"color: green;\">Предварительное выделение
элементов для И. Ф. В. </b><b style=\"color: black;\">:</b><br>");
            strHtml_A.append("<b>newVec :</b><table border=1 style=\"border-collapse:
collapse;\">");
            for(size_t i = 0; i < CN; i++){
                strHtml_A.append("<tr>");
                for(size_t j = 0; j < vec[i].count(); j++){
                    if( ((j % 2 == 1) && (j != 0)) && ( i == ( (2*Ga + 3*(1 - j))/2 ) || i
== (2*Ga + 5 - 3*j)/2 ))
                    || ( ((j % 2 == 0) && (j != 0)) && ( i == ((2*(Ga + 2) - 3*j)/2 ) ))){
                        vec[i][j] == 0 ? strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 0) + "</b></td>")
                        : strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 3) +
"</b></td>");
                        if (j != 0 && i == 0) {c.push_back(vec[i][j]);}
                        }else if(i > 0 && j != 0) {c.push_back(vec[i][j]);}
                        }else{
                            (i == Ga && j == 0)
                            ? strHtml_A.append("<td align=center><b style=\"color:
DarkCyan;\"> " + QString::number(vec[i][j], 'f', 3) + "</b></td>")
                            : vec[i][j] == 0
                            ? strHtml_A.append("<td align=center> " +
QString::number(vec[i][j]) + " </td>")
                            : strHtml_A.append("<td align=center> " +
QString::number(vec[i][j], 'f', 3) + " </td>");
                            }
                        }
                strHtml_A.append("</tr>");} strHtml_A.append("</table>");
                ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
"<p></p>");
            };

```

```

    auto general_i = [&]() -> void {
        size_t n = 3;
        auto FL = [CN = CN] (QVector<QVector<double>> NewVec, size_t N, double
EPS) -> bool {
            bool FLAG = false;
            for(size_t i = 0; i < CN; i++){
                FLAG = (abs(NewVec[i][N-1]) > EPS) ? (true) : ((FLAG ==
true) ? (true) : (false));
            } return FLAG;
        };
        auto general = [&](auto GEN, int64_t N, QVector<QVector<double>> A,
QVector<double>& C) -> void {
            QVector<QVector<double>> newVec(CN, QVector<double>(N, 0));
            for(size_t i = 0; i < CN; i++){
                for(size_t j = 0; j < A[i].count(); j++){
                    newVec[i][j]=A[i][j];
                }
            }
            for(size_t i = 0; i < CN - (N - 2); i++){
                newVec[i][N - 1] = (double)(newVec[i + 1][N - 2] - newVec[i][N - 2]);
            }

            if( FL(newVec, N, eps) && (N <= CN)){
                GEN(GEN, N + 1, newVec, C);
            }else{
                print_general(newVec); // print one, end iteration;

                // I M. Φ. Γ.:
                {
                    // #:
                    for(size_t i = 0; i < CN; i++){
                        if(i == 0){ step_a = abs(A[i][0] - B[0]); Ga = i;
                        }else if ( (step_a > (A[i][0] - B[0])) && (A[i][0] <= B[0])){
                            step_a = abs(A[i][0] - B[0]); Ga = i;
                        }
                    }

                    if(i == 0){ step_b = abs(A[(CN-1)][0] - B[0]); Gb = (CN - 1);
                    }else if ( (step_b > (A[(CN-1) - i][0] - B[0])) && (A[(CN-1) -
i][0] >= B[0])){ step_b = abs(A[(CN-1) - i][0] - B[0]); Gb = (CN-1) - i; }
                }

                print_general_G_1(newVec, Ga, C);
                if(!C.empty()) && (C[C.count() - 1] == A[Ga][1]){ QString
strHtml; strHtml.clear(); strHtml.append("<b style=\"color: red;\">Вычисления
</b><b style=\"color: black;\">:</b><br>");
                    strHtml.append("<b style=\"color: green;\" >x =
</b> " + QString::number(B[0]) + "; ");
                    strHtml.append("<b style=\"color: green;\" >x_0 =
</b> " + QString::number(A[Ga][0]) + "; ");
                    strHtml.append("<b style=\"color: green;\" >h =
</b> " + QString::number(newVec[1][0] - newVec[0][0]) + "; ");
                    double q = (B[0] - A[Ga][0]) / (newVec[1][0] -
newVec[0][0]);
                    strHtml.append("<b style=\"color: blue;\" >q = (x -
x_0) / h = </b> (" + QString::number(B[0]) + " - "
                        + QString::number(A[Ga][0]) + " ) / " +
                        QString::number(newVec[1][0] - newVec[0][0])
                        + " = " + QString::number(q) + "; ");
                    ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml()
+ strHtml + "<p></p>"); strHtml.clear();
                    strHtml.append("<b style=\"color: red;\">Вычисления </b><b style=\"color:
black;\">:</b><br>");

```

```

        auto P_n = [&]() -> double {
            auto ITER = [&](auto iter, size_t i) -> double {
                if(i >= C.count() - 1/*COLUMN: y;*/) return
0;

                auto factorial = [](auto fact, size_t i) ->
double {
                    return (i == 1 || i == 0) ? i/*База
факториала*/ : i*fact(fact, i - 1);
                };
                auto Q = [&](auto Q, size_t i) -> double {
                    return (i == 0) ? q
                        : (i % 2 == 0) ? (q + (i
/ 2)) * Q(Q, i - 1)
                        : (q - ((i
+ 1) / 2)) * Q(Q, i - 1);
                };
                return C[C.count() - 1 - i] * Q(Q, i - 1) /
factorial(factorial, i) + iter(iter, i + 1);
            };
            return C[C.count() - 1] + ITER(ITER, 1);
        };

        switch(C.count() - 1/*COLUMN: y;*/){
            case(3):
                strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub></sub>" + QString::number(N - 2/*COLUMN: y;*/) + "</sub>(x
= " + QString::number(B[0]) + ") "
                "= " + QString::number(P_n(), 'f',
6) + " ; ");
                C.clear();
                break;
            case(2):
                strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub></sub>" + QString::number(N - 2/*COLUMN: x, y;*/) +
"</sub>(x = " + QString::number(B[0]) + ") = " + QString::number(P_n(), 'f', 6)
+ " ; ");
                C.clear();
                break;
            case(1):
                strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub></sub>" + QString::number(N - 2/*COLUMN: x, y;*/) +
"</sub>(x = " + QString::number(B[0]) + ") = " + QString::number(P_n(), 'f', 6)
+ " ; ");
                C.clear();
                break;
            default:
                strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub></sub>" + QString::number(N - 2/*COLUMN: y;*/) + "</sub>(x
= " + QString::number(B[0]) + ") "
                "= " + QString::number(P_n(), 'f',
6) + " ; ");
                C.clear();
                break;
        };
        ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml()
+ strHtml + "<p></p>");
    }
    // #;
}

```

```

// И. Ф. С.:
{
    // #:
    for(size_t i = 0; i < CN; i++){
        if(i == 0){
            step_a = abs(A[i][0] - B[1]);
            Ga = i;
        }else if ( (step_a > (A[i][0] - B[1])) && (A[i][0] <=
B[1])){

            step_a = abs(A[i][0] - B[1]);
            Ga = i;
        }

        if(i == 0){
            step_b = abs(A[(CN-1)][0] - B[1]);
            Gb = (CN - 1);
        }else if ( (step_b > (A[(CN-1) - i][0] - B[1])) &&
(A[(CN-1) - i][0] > B[1])){
            //qDebug() << step_b;
            step_b = abs(A[(CN-1) - i][0] - B[1]);
            Gb = (CN-1) - i;
        }
    }

    //qDebug() << Ga;
    //qDebug() << Gb;

    print_general_S(newVec, Ga, C);
    if(!C.empty() && (C[C.count() - 1] == A[Ga][1])){
        QString strHtml;
        strHtml.clear();
        strHtml.append("<b style=\"color: red;\">Вычисления
</b><b style=\"color: black;\">:</b><br>");
        strHtml.append("<b style=\"color: green;\" >x = </b> " +
QString::number(B[1]) + "; ");
        strHtml.append("<b style=\"color: green;\" >x_0 = </b> " +
QString::number(A[Ga][0]) + "; ");
        strHtml.append("<b style=\"color: green;\" >h = </b> " +
QString::number(newVec[1][0] - newVec[0][0]) + "; ");
        double q = (B[1] - A[Ga][0]) / (newVec[1][0] -
newVec[0][0]);
        strHtml.append("<b style=\"color: blue;\" >q = (x -
x_0) / h = </b> (" + QString::number(B[1]) + " - "
+ QString::number(A[Ga][0]) + ") / " +
QString::number(newVec[1][0] - newVec[0][0])
+ " = " + QString::number(q) + "; ");
        ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() +
strHtml + "<p></p>");
        strHtml.clear();
        strHtml.append("<b style=\"color: red;\">Вычисления </b><b style=\"color:
black;\">:</b><br>");
    }
}

```

```

        auto P_n = [&]() -> double {
            auto ITER = [&](auto iter, size_t i) -> double {
                if(i >= C.count() - 2/*COLUMN: y;*/ ) return 0;
                auto factorial = [&](auto fact, size_t i) ->
double {
                    return (i == 1 || i == 0) ? i/*База
факториала*/ : i*fact(fact, i - 1);
                };

                static int STEP = 0;
                auto Q = [&](auto Q, size_t i) -> double {
                    int ST = STEP;
                    double B_0 = q - STEP;
                    double B_1 = q + STEP;

                    if(i == 1){
                        return B_0;
                    }else if(i % 2 == 0){
                        double V_1 = B_1;
                        double sa = V_1*Q(Q, i - 1);
                        return sa;
                    }else{
                        double V_0 = B_0;
                        STEP++;
                        return V_0*Q(Q, i - 1);
                    }
                };
                STEP = 0;
                if(i % 3 != 0){
                    qDebug() << "i: " << (2*i + 1)/3 << "STEP: "
<< STEP << "C_L: " << C[C.count() - 1 - i] << "C_R: " << C[C.count() - 2 - i]
<< ";";

                    int f = factorial(factorial, (2*i + 1)/3);
                    double qr = Q(Q, (2*i + 1)/3);
                    qDebug() << "factor(i): " << f << " Q(i): "

<< QString::number(qr);

                    STEP = 0;
                    return (C[C.count() - 1 - i] + C[C.count() -
2 - i]) * Q(Q, (2*i + 1)/3) / 2*factorial(factorial, (2*i + 1)/3) + iter(iter,
i + 2);

                }else{
                    qDebug() << "i: " << 2*i/3 << "; STEP:" <<
STEP << "C: " << C[C.count() - 1 - i] ;
                    int f = factorial(factorial, (2*i + 1)/3);
                    double qr = Q(Q, 2*i/3);
                    qDebug() << "factor(i): " << f << " Q(i): "

<< QString::number(qr);

                    STEP = 0;
                    //qDebug() << "; factor(i):" <<
factorial(factorial, (2*i + 1)/3) << "; Q(i):" << Q(Q, 2*i/3) << ";\n";
                    return (C[C.count() - 1 - i] + 0) * Q(Q,
2*i/3) / factorial(factorial, 2*i/3) + iter(iter, i + 1);
                }

                //qDebug() << C.count();
            };
            return C[C.count() - 1] + ITER(ITER, 1);
        };

```

```

switch(C.count() - 1/*COLUMN: y;*/) {
    case(3):
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(C.count() - 2/*COLUMN: y;*/) +
"</sub>(x = " + QString::number(B[1]) + ") "
                                "= " + QString::number(P_n(), 'f', 6) +
" ; ");
        C.clear();
        break;
    case(2):
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(N - 2/*COLUMN: x, y;*/) +
"</sub>(x = " + QString::number(B[1]) + ") = " + QString::number(P_n(), 'f', 6)
+ " ; ");
        C.clear();
        break;
    case(1):
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(N - 2/*COLUMN: x, y;*/) +
"</sub>(x = " + QString::number(B[1]) + ") = " + QString::number(P_n(), 'f', 6)
+ " ; ");
        C.clear();
        break;
    default:
        strHtml.append("<b style=\"color: orangeRed;\"
>P<sub>n</sub>(x) = P<sub>" + QString::number(C.count() - 2/*COLUMN: y;*/) +
"</sub>(x = " + QString::number(B[1]) + ") "
                                "= " + QString::number(P_n(), 'f', 6) +
" ; ");
        C.clear();
        break;
};
ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() +
strHtml + "<p></p>");
}
// #;
}

```

```

// И. Ф. Б.:
{
    // #:
    for(size_t i = 0; i < CN; i++){
        if(i == 0){
            step_a = abs(A[i][0] - B[2]);
            Ga = i;
        }else if ( (step_a > (A[i][0] - B[2])) && (A[i][0] <=
B[2])){

            step_a = abs(A[i][0] - B[2]);
            Ga = i;
        }

        if(i == 0){
            step_b = abs(A[(CN-1)][0] - B[2]);
            Gb = (CN - 1);
        }else if ( (step_b > (A[(CN-1) - i][0] - B[2])) &&
(A[(CN-1) - i][0] > B[2])){
            //qDebug() << step_b;
            step_b = abs(A[(CN-1) - i][0] - B[2]);
            Gb = (CN - 1) - i;
        }
    }

    qDebug() << Ga;
    qDebug() << Gb;

    print_general_B_1(newVec, Ga, C);
    if(!C.empty()) && (C[C.count() - 2] == A[Ga][1]){
        QString strHtml;
        strHtml.clear();
        strHtml.append("<b style=\"color: red;\">Вычисления
</b><b style=\"color: black;\">:</b><br>");
        strHtml.append("<b style=\"color: green;\" >x = </b> " +
QString::number(B[2]) + "; ");
        strHtml.append("<b style=\"color: green;\" >x_0 = </b> " +
QString::number(A[Ga][0]) + "; ");
        strHtml.append("<b style=\"color: green;\" >h = </b> " +
QString::number(newVec[1][0] - newVec[0][0]) + "; ");
        double q = (B[2] - A[Ga][0]) / (newVec[1][0] -
newVec[0][0]);
        strHtml.append("<b style=\"color: blue;\" >q = (x -
x_0) / h = </b> (" + QString::number(B[2]) + " - "
+ QString::number(A[Ga][0]) + ") / " +
QString::number(newVec[1][0] - newVec[0][0])
+ " = " + QString::number(q) + "; ");
        ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() +
strHtml + "<p></p>");
        strHtml.clear();
        strHtml.append("<b style=\"color: red;\">Вычисления
</b><b style=\"color: black;\">:</b><br>");

```

```

        auto P_n = [&]() -> double {
            auto ITER = [&](auto iter, size_t i) -> double {
                if(i >= C.count() - 2/*COLUMN: y;*/ ) return 0;
                auto factorial = [&](auto fact, size_t i) ->
double {
                    return (i == 1 || i == 0) ? i/*База
факториала*/ : i*fact(fact, i - 1);
                };

                static int STEP = 0;
                auto Q = [&](auto Q, size_t i) -> double {
                    int ST = STEP;
                    double B_0 = q - STEP;
                    double B_1 = q + STEP;

                    if(i == 1){
                        return B_0;
                    }else if(i % 2 == 0){
                        double V_1 = B_1;
                        double sa = V_1*Q(Q, i - 1);
                        return sa;
                    }else{
                        double V_0 = B_0;
                        STEP++;
                        return V_0*Q(Q, i - 1);
                    }
                };
                STEP = 0;

                if(i % 3 != 0){
                    qDebug() << "i: " << (2*i + 1)/3 << "STEP: "
<< STEP << "C_L: " << C[C.count() - 1 - i] << "C_R: " << C[C.count() - 2 - i]
<< ";";

                    int f = factorial(factorial, (2*i + 1)/3);
                    double qr = Q(Q, (2*i + 1)/3);
                    qDebug() << "factor(i): " << f << " Q(i): "
<< QString::number(qr);

                    STEP = 0;
                    return (C[C.count() - 1 - i] + C[C.count() -
2 - i]) * Q(Q, (2*i + 1)/3) / 2*factorial(factorial, (2*i + 1)/3) + iter(iter,
i + 2);

                }else{
                    qDebug() << "i: " << 2*i/3 << "; STEP:" <<
STEP << "C: " << C[C.count() - 1 - i] ;
                    int f = factorial(factorial, (2*i + 1)/3);
                    double qr = Q(Q, 2*i/3);
                    qDebug() << "factor(i): " << f << " Q(i): "
<< QString::number(qr);

                    STEP = 0;
                    return (C[C.count() - 1 - i] + 0) * Q(Q,
2*i/3) / factorial(factorial, 2*i/3) + iter(iter, i + 1);
                }
            };
            return (C[C.count() - 1] + C[C.count() - 2])/2 +
ITER(ITER, 2);
        };

```



```

        switch(C.count() - 1/*COLUMN: y;*/) {
            case(3):
                strHtml.append("<b style=\"color: orangeRed;\">P<sub>n</sub>(x) = P<sub>N - 2/*COLUMN: y;*/> + QString::number(N - 2/*COLUMN: y;*/> + QString::number(B[2]) + \">\" + QString::number(P_n(), 'f', 6) + \" ; \");
                C.clear();
                break;
            case(2):
                strHtml.append("<b style=\"color: orangeRed;\">P<sub>n</sub>(x) = P<sub>N - 2/*COLUMN: x, y;*/> + QString::number(N - 2/*COLUMN: x, y;*/> + QString::number(B[2]) + \">\" + QString::number(P_n(), 'f', 6) + \" ; \");
                C.clear();
                break;
            default:
                strHtml.append("<b style=\"color: orangeRed;\">P<sub>n</sub>(x) = P<sub>N - 2/*COLUMN: y;*/> + QString::number(N - 2/*COLUMN: y;*/> + QString::number(B[2]) + \">\" + QString::number(P_n(), 'f', 6) + \" ; \");
                C.clear();
                break;
        };
        ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml + "<p></p>");
    }

    // #;
}

};
general(general, n, A, C);
};
general_i();

```

Ссылка на git-Hub: [https://github.com/MineevS/CHM\\_3\\_6.git](https://github.com/MineevS/CHM_3_6.git)

### 3 Задание № 3

#### 3.1 Текст задания.

Вычислить значения функции при заданных значениях аргумента, используя *интерполяционную формулу Ньютона* для не равноотстоящих узлов и многочлен *Лагранжа*.

Таблица значений функции для вариантов 7 – {12} – 13	
X	Y
0.593	0.532050
0.598	0.535625
0.605	0.540598
0.613	0.546235
0.619	0.550431
0.627	0.555983
0.632	0.559428
0.640	0.568738
0.650	0.575298

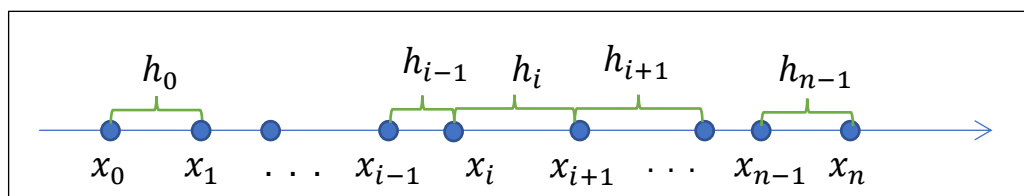
Значения аргумента для интерполяционных формул.

Ньютона с неравноотстоящими узлами		Многочлена Лагранжа	
№ вар	Значения аргумента	№ вар	Значения аргумента
N = 12	$0.609 + 0.002 * (\text{от } 9 \text{ до } 3*N)$	N = 12	$0.629 + 0.003 * (\text{от } 5 \text{ до } 3*N)$
Примечание. Запись «(от В до С)» означает, что множитель пробегает значения от В до С с шагом 1. В результате этого получается множество аргументов, для которых надо считать значение интерполируемой функции.			

#### 3.2 Теоретическая часть.

##### 3.2.1 Интерполяционные формулы для неравноотстоящих узлов.

Ранее, в задании № 1 были рассмотрены интерполяционные формулы для **равноотстоящих узлов**, но не изредка приходится встречаться с ситуацией, когда узлы некоторой функции находятся друг от друга не с одним и тем же шагом, а с различными шагами:



$$h_0 \neq h_1 \neq \dots \neq h_{i-1} \neq h_i \neq h_{i+1} \neq \dots \neq h_{n-1};$$

В этом случае говорят о **неравноотстоящих узлах** функции и для вычисления значения в промежуточных аргументах используют **интерполяционную формулу Ньютона для неравноотстоящих узлов** и **интерполяционную формулу Лагранжа**.

### 3.2.2 Интерполяционная формула Ньютона для неравноотстоящих узлов.

Используя разделённые разности, многочлен Лагранжа можно представить в виде первой интерполяционной формулы Ньютона. Интерполяционная формула Ньютона для функций с не равноотстоящими узлами интерполяции:

$$P(x) = y_0 + [x_0, x_1] \cdot (x - x_0) + [x_0, x_1, x_2] \cdot (x - x_0)(x - x_1) + \dots + [x_0, x_1, \dots, x_n] \cdot (x - x_0)(x - x_1) \cdot \dots \cdot (x - x_{n-1}),$$

где  $[x_0, \dots, x_n]$  — разделённая разность порядка  $n$ .

Для облегчения расчетов можно воспользоваться таблицей разделённых разностей:

Общий вид таблицы разделенных разностей:

$\Delta^n y_i$ $x_i$	$y_i$	$[x_i, x_{i+1}]$	$[x_i, x_{i+1}, x_{i+2}]$	$[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$	...	$[x_i, \dots, x_{i+m}]$
$x_0$	$y_0$	$[x_0, x_1] = \frac{y_1 - y_0}{x_1 - x_0}$	$[x_0, x_1, x_2] = \frac{[x_1, x_2] - [x_0, x_1]}{x_2 - x_0}$	$[x_0, x_1, x_2, x_3] = \frac{[x_1, x_2, x_3] - [x_0, x_1, x_2]}{x_3 - x_0}$	...	$[x_0, \dots, x_{0+m}]$
$x_1$	$y_1$	$[x_1, x_2] = \frac{y_2 - y_1}{x_2 - x_1}$	$[x_1, x_2, x_3] = \frac{[x_2, x_3] - [x_1, x_2]}{x_3 - x_1}$	$[x_1, x_2, x_3, x_4] = \frac{[x_2, x_3, x_4] - [x_1, x_2, x_3]}{x_4 - x_1}$	...	$[x_1, \dots, x_{1+m}]$
$x_2$	$y_2$	$[x_2, x_3] = \frac{y_3 - y_2}{x_3 - x_2}$	$[x_2, x_3, x_4] = \frac{[x_3, x_4] - [x_2, x_3]}{x_4 - x_2}$	$[x_2, x_3, x_4, x_5] = \frac{[x_3, x_4, x_5] - [x_2, x_3, x_4]}{x_5 - x_2}$	...	$[x_2, \dots, x_{2+m}]$

$x_3$	$y_3$	$[x_3, x_4] = \frac{y_4 - y_3}{x_4 - x_3}$	$[x_3, x_4, x_5] = \frac{[x_4, x_5] - [x_3, x_4]}{x_5 - x_3}$	$[x_3, x_4, x_5, x_6] = \frac{[x_4, x_5, x_6] - [x_3, x_4, x_5]}{x_6 - x_3}$	...	$[x_3, \dots, x_{3+m}]$
...	...	...	...	...	...	...
$x_m$	$y_m$	$\emptyset$	$\emptyset$	$\emptyset$	...	$\emptyset$

### 3.2.3 Интерполяционная формула Лагранжа для неравноотстоящих узлов.

- Пусть на  $[a, b]$  дано  $n + 1$  значение аргумента,

$$f(x_0) = y_0, \dots, f(x_n) = y_n.$$

- Требуется построить многочлен степени не выше  $n$ :

$$L_n(x_i) = y_i, \forall i = 0, 1, \dots, n$$

- Интерполяционный многочлен Лагранжа:

$$L_n(x) = \sum_{i=0}^n y_i \cdot \prod_{j=0, j \neq i}^n \frac{(x - x_j)}{(x_i - x_j)};$$

Интерполяционная формула Лагранжа.

Имеем: на  $[a, b]$  задано  $n + 1$  значение узлов:  $x_0, x_1, \dots, x_{n+1}$ .  $y = f(x)$ :  $y_0 =$

$$f(x_0), y_1 = f(x_1), \dots, y_n = f(x_n).$$

Стоит обратить внимание, что здесь не задано шага.

Требуется: построить полином  $L_n(x_i) = y_i, i = 0, 1, \dots, n$ ;

Рассмотрим полином:  $P_i(x_j) = 0$ , при  $i \neq j$ .

$$P_i(x_j) = 1, \text{ при } i = j.$$

$$P_i(x) = c_i(x - x_0)(x - x_1) \cdot \dots \cdot (x - x_{i-1})(x - x_{i+1}) \cdot \dots \cdot (x - x_{n-1})(x - x_n).$$

Положим  $x = x_i$ :

Получим:

$$P_i(x = x_i) = c_i(x - x_0)(x - x_1) \cdot \dots \cdot (x - x_{i-1})(x - x_{i+1}) \cdot \dots \cdot (x - x_{n-1})(x - x_n) = 1.$$

$$\Rightarrow c_i = \frac{1}{(x_i - x_0)(x_i - x_1) \cdot \dots \cdot (x_i - x_{i-1})(x_i - x_{i+1}) \cdot \dots \cdot (x_i - x_{n-1})(x_i - x_n)};$$

$$\Rightarrow P_i(x) = \frac{(x - x_0)(x - x_1) \cdot \dots \cdot (x - x_{i-1})(x - x_{i+1}) \cdot \dots \cdot (x - x_{n-1})(x - x_n)}{(x_i - x_0)(x_i - x_1) \cdot \dots \cdot (x_i - x_{i-1})(x_i - x_{i+1}) \cdot \dots \cdot (x_i - x_{n-1})(x_i - x_n)}.$$

$$L_n(x) = \sum_{i=0}^n P_i(x) \cdot y_i = \sum_{i=0}^n y_i \cdot \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_{n-1})(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_{n-1})(x_i-x_n)};$$

– Интерполяционная формула Лагранжа.

Она работает для любых наборов узлов, как *равноотстоящих*, так и *неравноотстоящих*. Он имеет единственный вид.

Пример № 31:

Дано: Построить Интерполяционный многочлен Лагранжа на трех узлах.

$$x_0 = 1, x_1 = e, x_2 = e^2.$$

Функции  $y = \ln(x)$ .

Решение:

1)  $n = 2$ :

$$L_n(x) = \sum_{i=0}^{n=2} y_i \cdot \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_{n-1})(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_{n-1})(x_i-x_n)} = y_0 \cdot \frac{(x-x_1)\dots(x-x_2)}{(x_0-x_1)\dots(x_0-x_2)} + y_1 \cdot \frac{(x-x_0)\dots(x-x_2)}{(x_1-x_0)\dots(x_1-x_2)} + y_2 \cdot \frac{(x-x_0)\dots(x-x_1)}{(x_2-x_0)\dots(x_2-x_1)}$$

2)  $y_i = f(x_i)$ .

$$y_0 = \ln(x_0) = \ln(1) = 0;$$

$$y_1 = \ln(x_1) = \ln(e) = 1;$$

$$y_2 = \ln(x_2) = \ln(e^2) = 2;$$

$$3) L_2(x) = 0 + \frac{(x-1)\dots(x-e^2)}{(e-1)\dots(e-e^2)} + 2 \cdot \frac{(x-1)\dots(x-e)}{(e^2-1)\dots(e^2-e)} = \frac{x^2}{e(e+1)(1-e)} + x \cdot \frac{e^3+e^2-e-1}{e(e+1)(1-e)^2} + \frac{2e-e^2-e^3}{e(e+1)(1-e)^2};$$

4) Для подсчёта значения в конкретной точке просто нужно вместо  $x$  поставить значение в полином.

### 3.2.4 Оценка погрешности интерполяционной формулы Лагранжа.

Имеем:

$$y = f(x), L_n(x), n + 1 \text{ точку: } x_0, x_1, \dots, x_n;$$

$$y = f(x_i): y_0 = f(x_0), y_1 = f(x_1), \dots, y_n = f(x_n);$$

Нужно построить многочлен Лагранжа как можно ближе к самой функции, чтобы погрешность была минимальной.

$$R_n = f(x) - L_n(x); \text{ — Погрешность.}$$

Будем считать, что  $f(x)$  на  $[a, b]$  непрерывна т. е имеет все производные до  $n + 1$  в узлах  $x_0, x_1, \dots, x_n$ :  $x_i \in [a, b]$ .

Введем вспомогательную функцию:

$$u(x) = f(x) - L_n(x) - k \cdot \Pi_{n+1}(x), (*)$$

$$k = \text{Const},$$

$$\Pi_0^n(x) = (x - x_0)(x - x_1) \cdot \dots \cdot (x - x_n)(x - x_{n-1})(x - x_n);$$

(Произведение всех  $x$ );

— Следующее произведение.

Функция  $u(x)$  имеет  $n + 1$  корень в точках  $x_0, x_1, \dots, x_n$ .

• Подберем такой коэффициент  $k$ , чтобы у  $u(x)$  появился  $n + 2$  й корень в какой-то фиксированной точке  $\bar{x}$  из заданного отрезка  $[a, b]$ .

$\bar{x}$  не должен совпадать с узлами интерполирования.  $x_0, x_1, \dots, x_n$ .

Для того, чтобы существовал такой корень  $\bar{x}$ , необходимо и достаточно, чтобы выполнялось

следующее условие:

$$f(\bar{x}) - L_n(\bar{x}) - k \cdot \Pi_{n+1}(\bar{x}) = 0; \rightarrow k = \frac{f(\bar{x}) - L_n(\bar{x})}{\Pi_{n+1}(\bar{x})}; \rightarrow \Pi_{n+1}(\bar{x}) = 0;$$

$\Rightarrow u(x)$ :  $n + 2$  корня, и обращается в 0:

$\Rightarrow [x_0; x_1], [x_1; x_2], \dots, [x_i; \bar{x}], [\bar{x}; x_{i+1}], \dots, [x_{n-1}; x_n]$ .

• Для оценки погрешности нужно получить выражение для  $k$  через производную  $u(x)$ .

• Вспомним, что функция  $u(x)$  должна иметь  $n+1$  производную и мы можем сказать, что:

$u^{(n+1)}(x)$  на  $[a, b]$  будет иметь по крайней мере один корень  $\varepsilon$ :  $u^{(n+1)}(\varepsilon) = 0$ .

• Теперь рассмотрим  $n + 1$  производную  $L_n^{(n+1)}(x)$ .

Поскольку многочлен имеет  $n$  степень, а производная  $n+1$ , то такая производная будет равно 0. т. е.  $L_n^{(n+1)}(x) = 0$ .

И надо ещё рассмотреть  $\Pi_n^{(n+1)}(x)$ :

$$\Pi_n^{(n+1)}(x) = (n+1)!$$

Подставляем все в формулу (\*) и получаем:

$$u^{(n+1)}(x) = f^{(n+1)}(x) - k \cdot (n+1)!;$$

Соответственно, если вместо  $x$  подставить какой-то корень  $\varepsilon$ , то значение функции  $u^{(n+1)}(x)$  будет равно 0.

$$\text{т. е. } u^{(n+1)}(\varepsilon) = 0;$$

$$\Rightarrow u^{(n+1)}(\varepsilon) = f^{(n+1)}(x) - k \cdot (n+1)! = 0; \rightarrow k = \frac{f^{(n+1)}(\varepsilon)}{(n+1)!};$$

$$\frac{f(\bar{x}) - L_n(\bar{x})}{\Pi_{n+1}(\bar{x})} = \frac{f^{(n+1)}(\varepsilon)}{(n+1)!};$$

$$f(\bar{x}) - L_n(\bar{x}) = \frac{f^{(n+1)}(\varepsilon)}{(n+1)!} \cdot \Pi_{n+1}(\bar{x});$$

$\bar{x}$  – произвольная точка из  $[a, b]$

$$\Rightarrow R_n = f(x) - L_n(x) = \frac{f^{(n+1)}(\varepsilon)}{(n+1)!} \cdot \Pi_{n+1}(x);$$

- Формула оценки погрешности сверху,

$\Rightarrow$  которая справедлива для всех точек включая и

$\Rightarrow$  узлы интерполирования.

$$M_{n+1} = \max_{a < x < b} |f^{(n+1)}(x)|;$$

Получим:

$$|R_n| = |f(x) - L_n(x)| \leq \frac{M_{n+1}}{(n+1)!} \cdot \Pi_{n+1}(x);$$

– Оценка абсолютной погрешности И. Ф. Л.

Пример № 32:

$$y = \ln(x), x_0 = 1, x_1 = e, \dots, x_2 = e^2.$$

Задание:

- Определить, с какой точностью можно посчитать значение функции  $y(x)$ :

$$x = 2.$$

- 1) Считаем производные:

$$y' = \frac{1}{x}; y'' = -\frac{1}{x^2}; y''' = \frac{2}{x^3};$$

- 2) Найдем максимальное значение по 3 производной:

$$M_3 = \max_{1 < x < e^2} |y'''(x)| = \max_{1 < x < e^2} |y'''(x)|;$$

- посчитаем значение 3 производной в узлах:

$$y'''(x_0) = \left| \frac{2}{1^3} \right| = 2; y'''(x_1) = \left| \frac{2}{e^3} \right| = 0, \dots;$$

$$y'''(x_2) = \left| \frac{2}{e^6} \right| = 0.0\dots;$$

$$\Rightarrow M_3 = 2;$$

$$\Rightarrow |R_n| \leq \frac{2}{(2+1)!} \cdot |\Pi_3(x)|,$$

$x$  - значение в той точке, в которой считаем.

$$|R_n| \leq \frac{1}{3} \cdot |(2-1)(2-e)(2-e^2)|,$$

$$\Pi_3(x) = (x-x_0)(x-x_1)(x-x_2);$$

$$|R_n| \leq \frac{1}{3} \cdot |(2-1)(2-e)(2-e^2)|,$$

$$\Pi_3(x) = (x-x_0)(x-x_1)(x-x_2);$$

$$R_2 \leq \frac{1}{3} \cdot |(2-1)(2-e)(2-e^2)| \sim 1.23;$$

- Абсолютная погрешность.

### 3.3 Практическая часть. (Результаты работы программы.)

**Принятые данные :**

**A :**

x	y
0.59300	0.532050
0.59800	0.535625
0.60500	0.540598
0.61300	0.546235
0.61900	0.550431
0.62700	0.555983
0.63200	0.559428



0.64000	0.568738
0.65000	0.575298

**X\_N:**

9	36
---	----

**X\_L :**

5	36
---	----

**Точность :**  $\varepsilon = 0.00001$ ;

---

**Вычисления :**

**newVec :**

0.593000	0.532050
0.598000	0.535625
0.605000	0.540598
0.613000	0.546235
0.619000	0.550431
0.627000	0.555983
0.632000	0.559428
0.640000	0.568738
0.650000	0.575298

**Предварительное выделение элементов для И. Ф. Н. с неравноотстоящими узлами :**

**newVec :**

x	y	[x <sub>0</sub> ; x <sub>1</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ; x <sub>3</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ; x <sub>3</sub> ; x <sub>4</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ; x <sub>3</sub> ; x <sub>4</sub> ; x <sub>5</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ; x <sub>3</sub> ; x <sub>4</sub> ; x <sub>5</sub> ; x <sub>6</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ; x <sub>3</sub> ; x <sub>4</sub> ; x <sub>5</sub> ; x <sub>6</sub> ; x <sub>7</sub> ]	[x <sub>0</sub> ; x <sub>1</sub> ; x <sub>2</sub> ; x <sub>3</sub> ; x <sub>4</sub> ; x <sub>5</sub> ; x <sub>6</sub> ; x <sub>7</sub> ; x <sub>8</sub> ]
0.593	<b>0.532</b>	<b>0.715</b>	<b>-0.381</b>	<b>-0.298</b>	<b>27.800</b>	<b>-1386.044</b>	<b>48507.895</b>	<b>940894635.514</b>	<b>-88459210557.408</b>
0.598	0.536	0.710	-0.387	0.425	-19.326	505.764	44270555.765	-4101280366.258	0
0.605	0.541	0.705	-0.378	-0.135	-2.130	1859869.106	-168996023.281	0	0
0.613	0.546	0.699	-0.381	-0.193	65093.289	-5744951.942	0	0	0
0.619	0.550	0.694	-0.385	1757.326	-147469.933	0	0	0	0
0.627	0.556	0.689	36.519	-2814.242	0	0	0	0	0
0.632	0.559	1.164	-28.208	0	0	0	0	0	0
0.640	0.569	0.656	0	0	0	0	0	0	0
0.650	0.575	0	0	0	0	0	0	0	0

**Ответ :**

**Y :**

x	y
0.627000	0.555983
0.629000	0.557314
0.631000	0.558692
0.633000	0.560218
0.635000	0.562031
0.637000	0.564286
0.639000	0.567105
0.641000	0.570503
0.643000	0.574270
0.645000	0.577805
0.647000	0.579890
0.649000	0.578395
0.651000	0.569897
0.653000	0.549203
0.655000	0.508767
0.657000	0.437973
0.659000	0.322286
0.661000	0.142239
0.663000	-0.127753
0.665000	-0.520775
0.667000	-1.079008
0.669000	-1.855557
0.671000	-2.916538
0.673000	-4.343439
0.675000	-6.235788
0.677000	-8.714140
0.679000	-11.923420

0.681000	-16.036635
----------	------------

**Вычисления :**

**newVec :**

0.593000	0.532050
0.598000	0.535625
0.605000	0.540598
0.613000	0.546235
0.619000	0.550431
0.627000	0.555983
0.632000	0.559428
0.640000	0.568738
0.650000	0.575298

**Ответ :**

**Y :**

x	y
0.644000	0.576125
0.647000	0.579890
0.650000	0.575298
0.653000	0.549203
0.656000	0.477970
0.659000	0.322286
0.662000	0.020361
0.665000	-0.520775
0.668000	-1.436192
0.671000	-2.916538
0.674000	-5.224428
0.677000	-8.714140
0.680000	-13.855016
0.683000	-21.259000
0.686000	-31.712781
0.689000	-46.214996
0.692000	-66.019020
0.695000	-92.681843
0.698000	-128.119607
0.701000	-174.670351
0.704000	-235.164581
0.707000	-313.004264
0.710000	-412.250903
0.713000	-537.723350
0.716000	-695.106048
0.719000	-891.068415
0.722000	-1133.396106
0.725000	-1431.134907
0.728000	-1794.748047
0.731000	-2236.287743
0.734000	-2769.581784
0.737000	-3410.436030

**Время работы: 0.242 ;**

### 3.4 Приложение № 1 – Примеры работы программы.

MainWindow

агрузка данных

Введите количество известных узлов таблицы: 9

Значения аргумента для интерполяционных формул Ньютона

X\_N = {0.609 + 0.002\*C} : ( 9 36 ) : N = 1; ;

Точность ε = 0.00001

MainWindow

агрузка данных

Введите количество известных узлов таблицы:

9

Значения аргумента для интерполяционных формул Ньютона

$X_N = \{0.609 + 0.002 \cdot C\} :$ 

9

36

 $] : H = 1 ; ) ;$

Точность  $\epsilon =$ 

0.00001

Значения аргумента для интерполяционных формул Лагранжа

$X_L = \{0.629 + 0.003 \cdot C\} :$ 

5

36

 $] : H = 1 ; ) ;$

Вычисление

Таблица значений функции

для варианта № 12

x	y
0.59300	0.532050
0.59800	0.535625
0.60500	0.540598
0.61300	0.546235
0.61900	0.550431
0.62700	0.555983
0.63200	0.559428
0.64000	0.568738
0.65000	0.575298

Вычисления :

newVec :

0.593000	0.532050
0.598000	0.535625
0.605000	0.540598
0.613000	0.546235
0.619000	0.550431
0.627000	0.555983
0.632000	0.559428
0.640000	0.568738
0.650000	0.575298

Вычисления :

newVec :

0	0.593000	0.532050
0	0.598000	0.535625
0	0.605000	0.540598
0	0.613000	0.546235
0	0.619000	0.550431
0	0.627000	0.555983
0	0.632000	0.559428
0	0.640000	0.568738
0	0.650000	0.575298

Ответ :

Y :

x	y
0.644000	0.576125
0.647000	0.579890
0.650000	0.575298
0.653000	0.549203
0.656000	0.477970
0.659000	0.322286
0.662000	0.020361
0.665000	-0.520775
0.668000	-1.436192
0.671000	-2.916538
0.674000	-5.224428
0.677000	-8.714140
0.680000	-13.855016
0.683000	-21.259000
0.686000	-31.712781
0.689000	-46.214996
0.692000	-66.019020
0.695000	-92.681843
0.698000	-128.119607
0.701000	-174.670351
0.704000	-235.164581
0.707000	-313.004264
0.710000	-412.250903
0.713000	-537.723350
0.716000	-695.106048
0.719000	-891.068415
0.722000	-1133.396106
0.725000	-1431.134907
0.728000	-1794.748047
0.731000	-2236.287743
0.734000	-2769.581784
0.737000	-3410.436030

### 3.5 Приложение № 2 – Листинг программы.

```
int Ga, Gb;
double step_a, step_b;

auto print_general = [&](QVector<QVector<double>>& vec) -> void {
    strHtml_A.clear();
    strHtml_A.append("<b style=\"color: red;\">Вычисления </b><b style=\"color: black;\">:</b><br>");
    strHtml_A.append("<b>newVec :</b><table align=center border=1 style=\"border-collapse: collapse;\">");

    if (vec.count() == 2) strHtml_A.append("<tr><td align=center> x </td><td align=center> y </td></tr>");
    foreach(auto str, vec){
        strHtml_A.append("<tr>");
        foreach(auto elem, str){
            elem == 0 ? strHtml_A.append("<td align=center> " +
                QString::number(elem) + " </td>")
                : strHtml_A.append("<td align=center> " +
                QString::number(elem, 'f', 6) + " </td>");
        }
        strHtml_A.append("</tr>");
    }
    strHtml_A.append("</table>");
    ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
    "<p></p>");
};

auto print_general_o = [&](QVector<QVector<double>>& vec) -> void {
    strHtml_A.clear();
    strHtml_A.append("<b style=\"color: red;\">Ответ </b><b style=\"color: black;\">:</b><br>");
    strHtml_A.append("<b>Y :</b><table align=center border=1 style=\"border-collapse: collapse;\">");

    strHtml_A.append("<tr><td align=center> x </td><td align=center> y </td></tr>");
    foreach(auto str, vec){
        strHtml_A.append("<tr>");
        foreach(auto elem, str){
            elem == 0 ? strHtml_A.append("<td align=center> " +
                QString::number(elem) + " </td>")
                : strHtml_A.append("<td align=center> " +
                QString::number(elem, 'f', 6) + " </td>");
        }
        strHtml_A.append("</tr>");
    }
    strHtml_A.append("</table>");
    ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
    "<p></p>");
};
```

```

    auto print_general_N = [&](QVector<QVector<double>>& vec, QVector<double>&
c) -> void {
        strHtml_A.clear();
        strHtml_A.append("<b style=\"color: green;\">Предварительное выделение
элементов для И. Ф. N. с неравноотстающими узлами </b><b style=\"color:
black;\">:</b><br>");
        strHtml_A.append("<b>newVec :</b><table align=center border=1
style=\"border-collapse: collapse;\">");

        strHtml_A.append("<tr><td align=center> x </td>"
            "<td align=center> y </td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub>]
</td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub>] </td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub> ; x<sub>3</sub>] </td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub> ; x<sub>3</sub> ; x<sub>4</sub>] </td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub> ; x<sub>3</sub> ; x<sub>4</sub> ; x<sub>5</sub>] </td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub> ; x<sub>3</sub> ; x<sub>4</sub> ; x<sub>5</sub> ; x<sub>6</sub>]
</td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub> ; x<sub>3</sub> ; x<sub>4</sub> ; x<sub>5</sub> ; x<sub>6</sub> ;
x<sub>7</sub>] </td>"
            "<td align=center> [x<sub>0</sub> ; x<sub>1</sub> ;
x<sub>2</sub> ; x<sub>3</sub> ; x<sub>4</sub> ; x<sub>5</sub> ; x<sub>6</sub> ;
x<sub>7</sub> ; x<sub>8</sub>] </td></tr>");

        for(size_t i = 0; i < CN; i++){
            strHtml_A.append("<tr>");
            for(size_t j = 0; j < vec[i].count(); j++){
                if(i == 0 && j > 0){
                    vec[i][j] == 0 ? strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 0) + "</b></td>")
                    : strHtml_A.append("<td align=center><b
style=\"color: DarkCyan;\"> " + QString::number(vec[i][j], 'f', 3) +
"</b></td>");
                    c.push_front(vec[i][j]);
                }else{
                    vec[i][j] == 0 ? strHtml_A.append("<td align=center> " +
QString::number(vec[i][j]) + " </td>")
                    : strHtml_A.append("<td align=center> " +
QString::number(vec[i][j], 'f', 3) + " </td>");
                }
            }
            strHtml_A.append("</tr>");
        }
        strHtml_A.append("</table>");
        ui->textBrowser_1->setHtml(ui->textBrowser_1->toHtml() + strHtml_A +
"<p></p>");
    };
    print_general(A);
    auto Duble_A = A;

```

```

    auto general_i = [&]() -> void {
        size_t n = 3;
        auto FL = [CN = CN] (QVector<QVector<double>> NewVec, size_t N, double
EPS) -> bool {
            bool FLAG = false;
            for(size_t i = 0; i < CN; i++){
                FLAG = (abs(NewVec[i][N-1]) > EPS) ? (true) : ((FLAG ==
true) ? (true) : (false));
            }
            return FLAG;
        };

        auto general = [&](auto GEN, int64_t N, QVector<QVector<double>>& A,
QVector<double>& C) -> void {
            QVector<QVector<double>> newVec(CN, QVector<double>(N, 0));
            for(size_t i = 0; i < CN; i++){// Копирование данных из A в newVec.
                for(size_t j = 0; j < A[i].count(); j++){
                    newVec[i][j] = A[i][j];
                }
            }

            auto razd_raznost = [] (auto r, QVector<QVector<double>>& A, int64_t
K, int64_t i) -> double {
                return (A[i + 1][K - 1] - A[i][K - 1]) / ((A[i + K - 1][0] -
A[i][0]));
            };

            for(size_t i = 0; i < CN - (N - 2); i++){
                newVec[i][N - 1] = razd_raznost(razd_raznost, A, N - 1, i);
            }

            if((N <= CN)){// FL(newVec, N, eps) &&
                GEN(GEN, N + 1, newVec, D);
            }else{
                // 1. Интерполяционный многочлен Ньютона.
                {
                    print_general_N(newVec, D); // print one, end iteration;
                    Y = QVector<QVector<double>>(B[1] - B[0] + 1,
QVector<double>(2, 0));

                    auto P_n = [D = D, A = A] (auto P_n, double x, uint32_t i =
0) -> double {
                        if(i > D.count()) return 0;
                        auto R = [A = A] (auto r, double x, uint32_t i) ->
double {
                            if (i == 1) return 1;
                            return (x - A[i - 2][0]) * r(r, x, i - 1);
                        };
                    };
                }
            }
        };
    };

```

```

        if (i == 0){
            return D[D.count() - 1] + P_n(P_n, x, i + 2);
        }else{
            return D[D.count() - i]*R(R, x, i) + P_n(P_n, x, i
+ 1);
        }
    };
    for(int i = B[0]; i <= B[1]; i++){
        double x = 0.609 + 0.002*i;
        double y = P_n(P_n, x);
        Y[i - B[0]][0] = x;
        Y[i - B[0]][1] = y;
    }
    print_general_o(Y);
}

// 2. интерполяционный многочлен лагранжа.
{
    print_general(Duble_A);
    W =
QVector<QVector<double>>(Duble_A.count(),QVector<double>(3, 0));
    for(size_t i = 0; i < Duble_A.count(); i++){
        for(size_t j = 1; j < 3; j++){
            W[i][j] = Duble_A[i][j - 1];
        }
    }
    print_general(W);
    auto P_n_L = [W = Duble_A](auto P_n_L, double x, int32_t i
= 0) -> double {
        if(i >= W.count()) return 0;
        auto Z = [] (auto z, double x, QVector<QVector<double>>
W, int32_t i, int32_t j = 0) -> double {
            if(i == j){
                return z(z, x, W, i, j + 1);
            }else if(i >= W.count() || j >= W.count()){
                return 1;
            }else {
                return(x - W[j][0])/(W[i][0] - W[j][0])*z(z, x,
W, i , j + 1);
            }
        };
    };
}

```



```

        return W[i][1]*Z(Z, x, W, i) + P_n_L(P_n_L, x, i + 1);
    };
    P = QVector<QVector<double>>(E[1] - E[0] + 1,
QVector<double>(2, 0));
    for(int i = E[0]; i <= E[1]; i++){
        double x = 0.629 + 0.003*i;
        double y = P_n_L(P_n_L, x);
        P[i - E[0]][0] = x;
        P[i - E[0]][1] = y;
    }
    print_general_o(P);
    }
    };
    general(general, n, A, C);
};
general_i();

```

Ссылка на git-Hub: [https://github.com/MineevS/CHM\\_3\\_6.git](https://github.com/MineevS/CHM_3_6.git)