Университет ИТМО
МФ КТиУ, Ф ПИиКТ

Лабораторная работа №3.2
Дисциплина «Вычислительная математика»

Приближение функций

Вариант:
Интерполирование многочленом Ньютона

Выполнил:
Студент группы P3212
Анищенко Анатолий Алексеевич

Преподаватель:
Перл Ольга Вячеславовна

г. Санкт-Петербург
2020 г.

Цель работы:

Реализовать интерполирование многочленом Ньютона для различного набора данных.

Описание использованного метода:

Интерполирование многочленом Ньютона это частный случай интерполирования многочленом Лагранжа, значения аргумента равномерно распределены на заданном отрезке: $x_i - x_{i-1} = h = const$, $i = 1, 2, 3, \dots, n$. Величина h называется шагом.

Будем искать интерполяционный многочлен Ньютона в следующем виде:

$$N(x) = a_0 + a_1(x - x_0) + a_1(x - x_0)(x - x_1) + \dots + a_n(x - x_0)(x - x_1)\dots(x - x_{n-1})$$

График многочлена должен проходить через заданные узлы. Для этого должно быть выполнено следующее условие: $N(x_i) = y_i \ (i = 0, 1, 2, \dots, n)$

$$\begin{cases} N(x_0) = a_0 = y_0 \\ N(x_1) = a_0 + a_1(x_1 - x_0) = a_0 + a_1 h = y_1 \\ N(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = a_0 + 2a_1 h + 2a_2 h^2 = y_2 \\ \dots \end{cases}$$

То есть:

$$\begin{cases} a_0 = y_0 \\ a_1 = \dfrac{y_1 - y_0}{h} = \dfrac{\Delta y_0}{h} \\ a_2 = \dfrac{y_2 - a_0 - 2a_1 h}{2h^2} = \dfrac{y_2 - y_0 - 2\Delta y_0}{2h^2} = \dfrac{\Delta^2 y_0,}{2h^2} \\ a_k = \dfrac{\Delta^k y_0}{k!\, h^k} \end{cases}$$

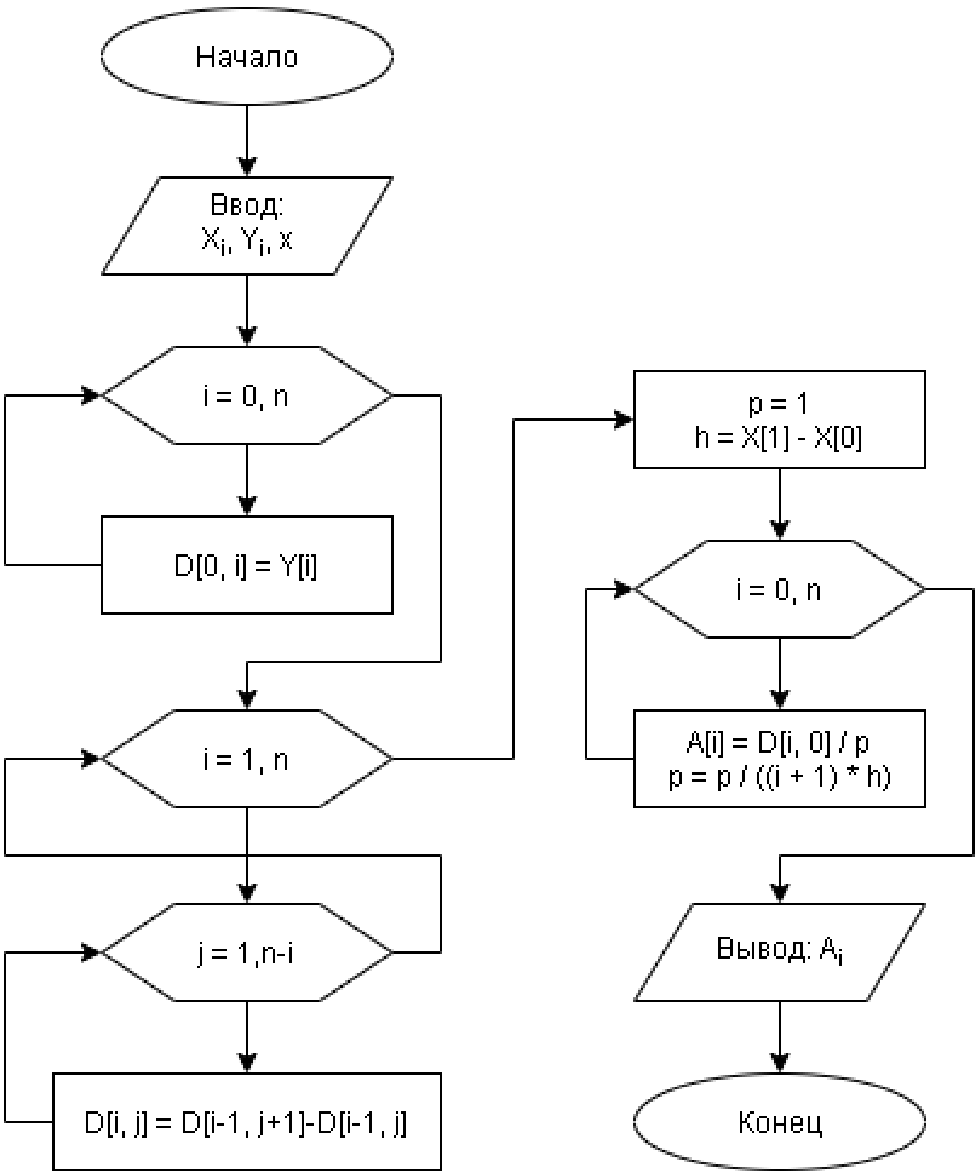
где $\Delta^k y_i$ – разность функции порядка k

$$\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i, \qquad \text{для } i = 1,2, \dots, n - k$$

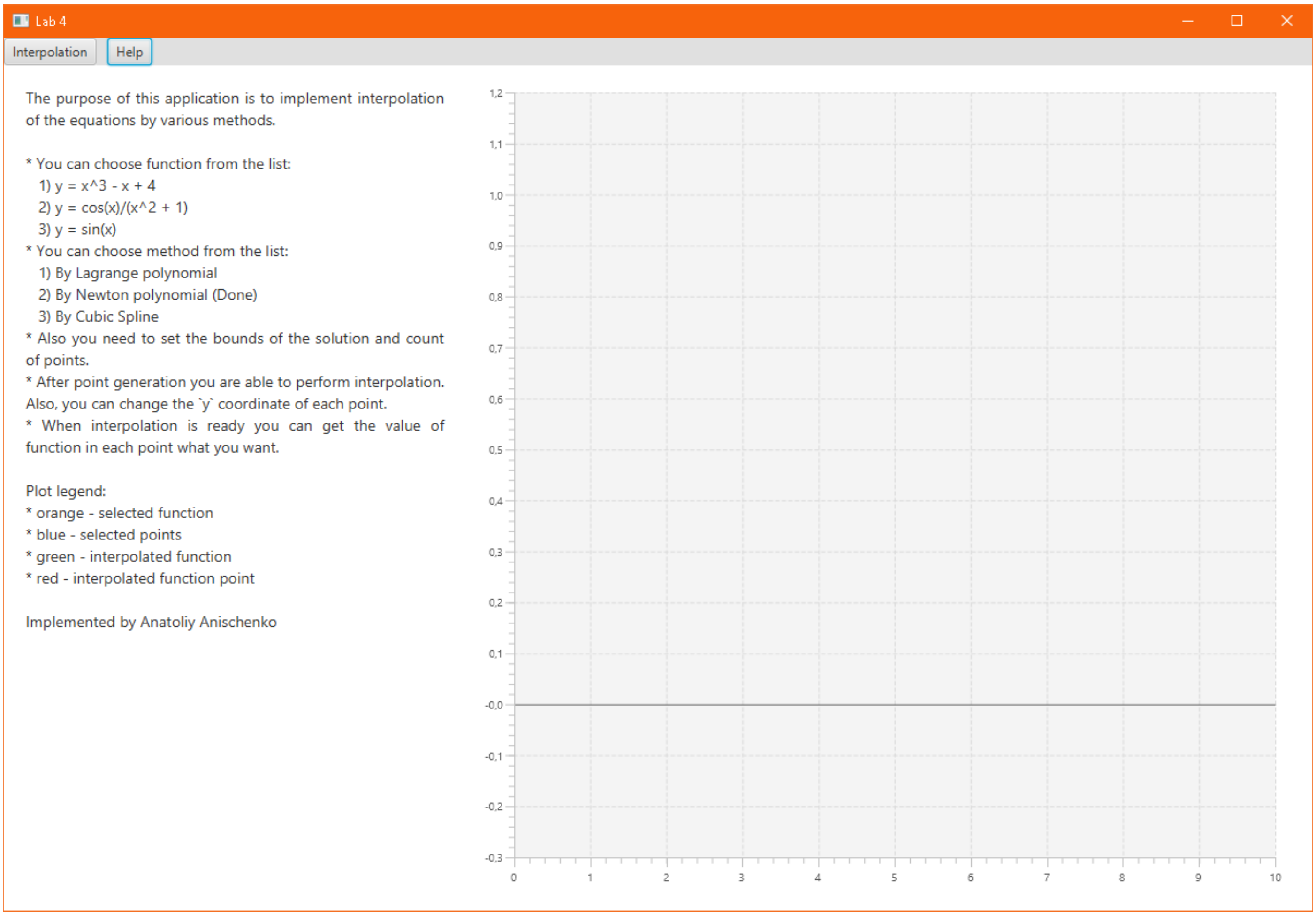
Листинг:

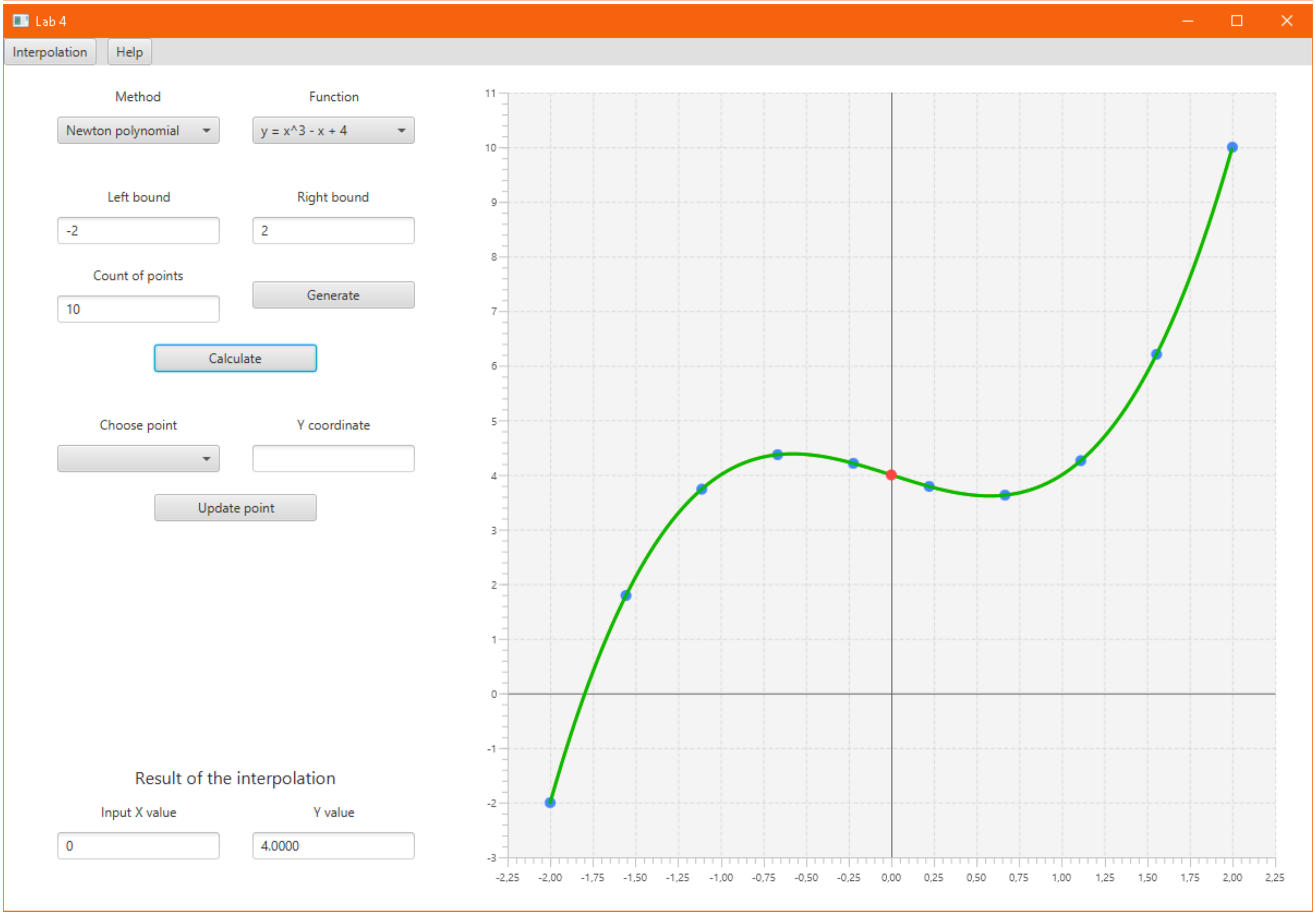
```
private DerivativeFunc solveByNewtonPolynomial(List<Point> points) {
    int n = points.size();
    double[][] dividedDiff = new double[n][n];

    for (int i = 0; i < n; i++) {
        dividedDiff[0][i] = points.get(i).second;
    }

    for (int i = 1; i < n; i++) {
        for (int j = 0; j < n - i; j++) {
            dividedDiff[i][j] = dividedDiff[i - 1][j + 1] - dividedDiff[i - 1][j];
        }
    }

    Point p0 = points.get(0);

    double h = points.get(1).first - p0.first;

    return new DerivativeFunc() {
```

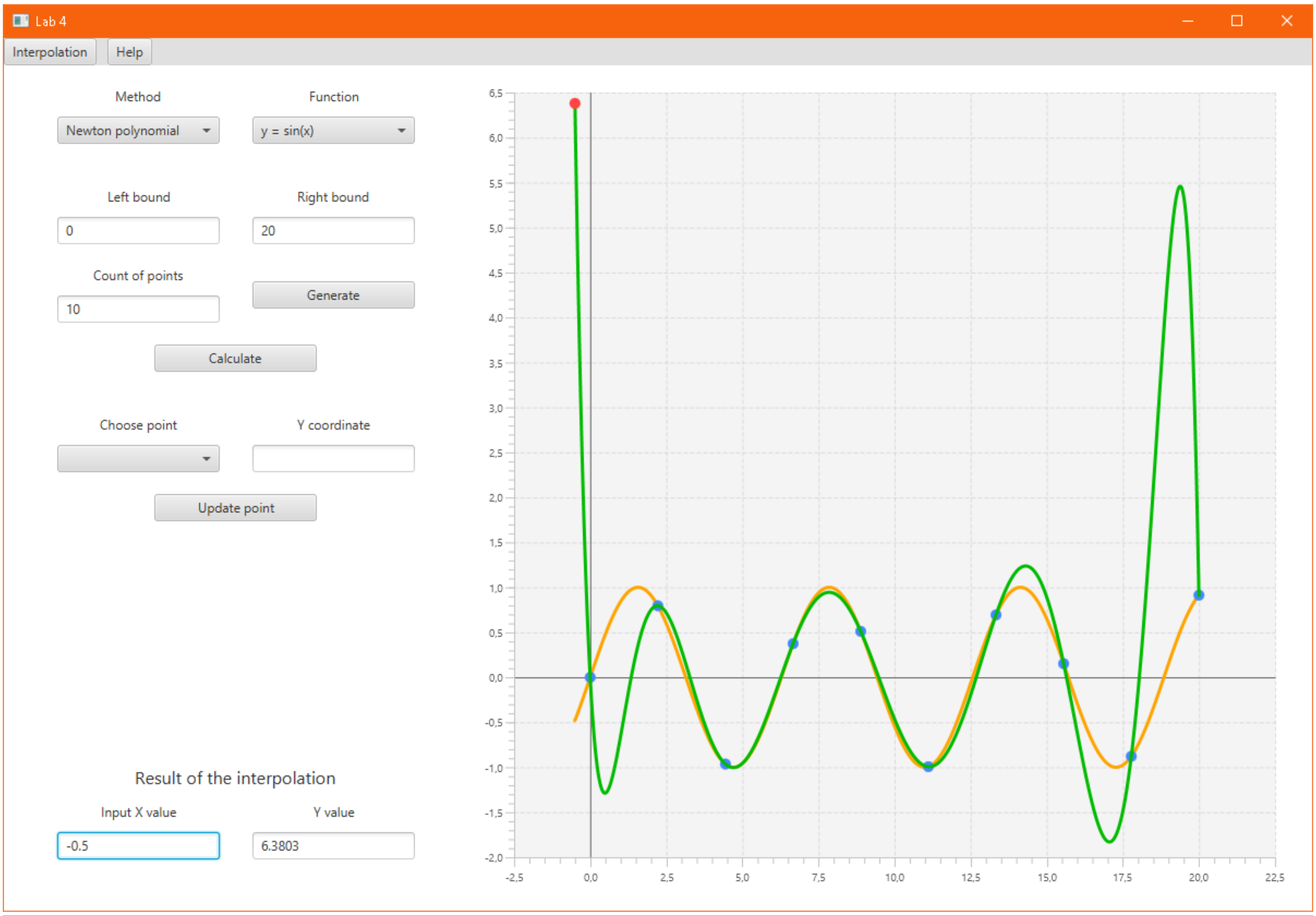
```
        @Override
        public double get(double argument) {
            double res = p0.second;

            double q = (argument - p0.first) / h;
            double product = 1;

            for (int i = 1; i < n; i++) {
                product *= q + 1 - i;
                product /= i;

                res += product * dividedDiff[i][0];
            }

            return res;
        }

        @Override
        public Interval[] getNotAllowedScope() {
            return new Interval[0];
        }
    };
}
```

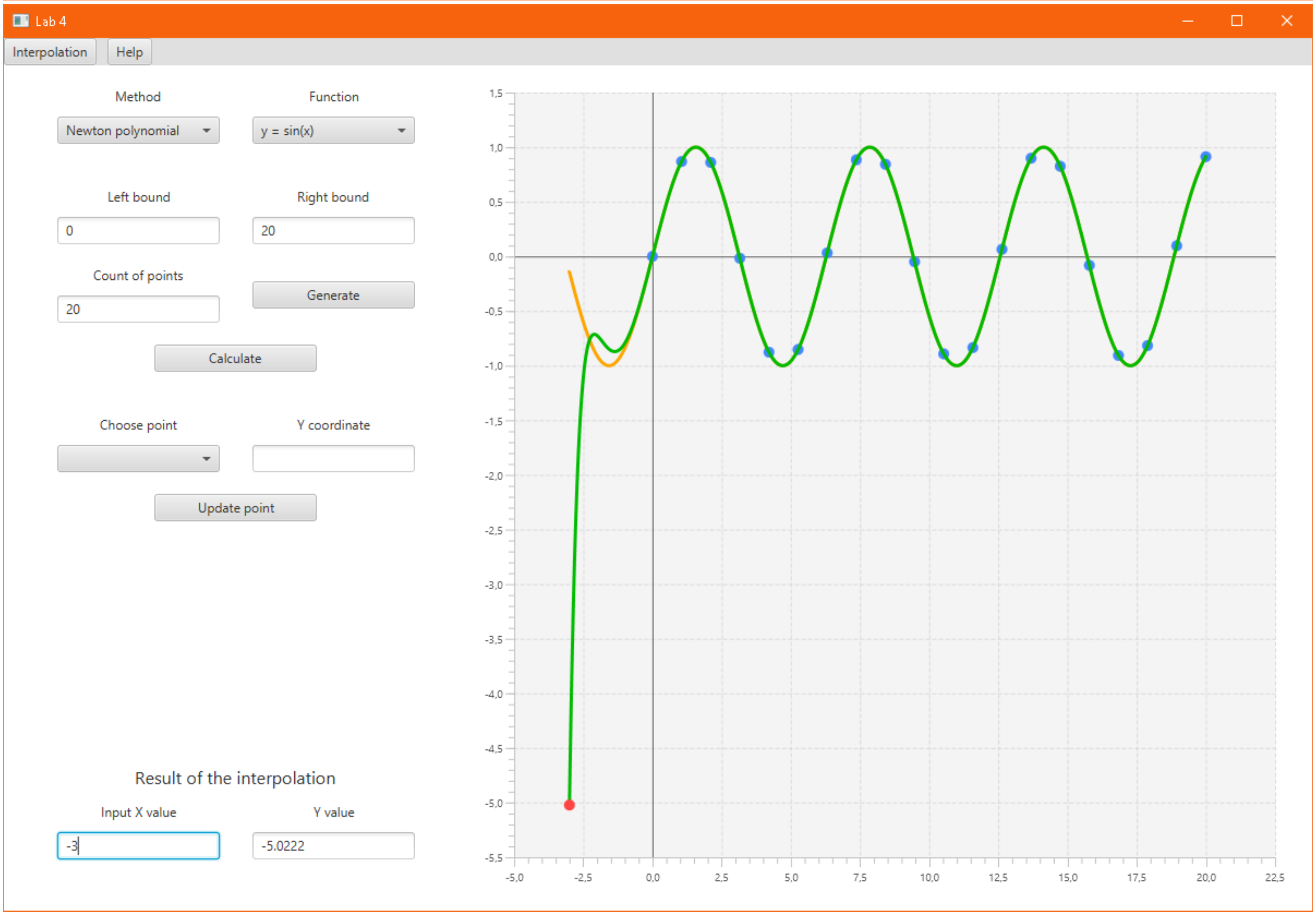


Результат работы:






Lab 4
Interpolation
Help
Method
Newton polynomial
Function
y = sin(x)
Left bound
0
Right bound
20
Count of points
10
Generate
Calculate
Choose point
Y coordinate
Update point
Result of the interpolation
Input X value
-0.5
Y value
6.3803


Lab 4
Interpolation
Help
Method
Newton polynomial
Function
y = sin(x)
Left bound
0
Right bound
20
Count of points
20
Generate
Calculate
Choose point
Y coordinate
Update point
Result of the interpolation
Input X value
-3
Y value
-5.0222

Выводы:

В данной лабораторной работе мы должны были реализовать или интерполяцию или аппроксимацию. Основное различие аппроксимации и интерполяции заключается в том, что интерполянт принадлежит к определённому классу функции (вид функции строго задан) и обязательно проходит через заданные узлы, а для аппроксиманта это требование не является обязательным.

Аппроксимация подбирает коэффициенты аппроксиманта для наилучшего приближения к аппроксимируемой функции. Интерполяция подбирает коэффициенты для интерполянта так, чтобы он проходил через все заданные узлы.

Сравнение методов:

1) Интерполирование кубическими сплайнами
   Один из способов кусочно-полиноминальной интерполяции, когда весь отрезок разбивают на частичные отрезки и на каждом из частичных отрезков приближенно заменяют исходную функцию многочленом невысокой третьей степени, в отличие от формул Ньютона и Лагранжа, где отрезок не разбивается. Интерполяцию кубическими сплайнами рационально применять, если $f(x)$ - периодическая или тригонометрическая функция. Кроме всего вышесказанного следует отметить, что большую точность интерполяции можно получить применением методов сплайн–интерполяции.
2) Формула Лагранжа
   Основным отличием этого метода является то, что его можно применять для таблиц с различными расстояниями между узлами. В отличие от формулы Ньютона можно добавить произвольные точки.
3) Формула Ньютона
   Можно применять только для таблиц с равноудаленными узлами. Для добавления новой точки не надо пересчитывать все коэффициенты, в отличие от формулы Лагранжа. Нельзя добавить произвольную точку, точки строго фиксированы.