Университет ИТМО

ФПИиКТ

Вычислительная математика

Лабораторная работа 3

Интерполирование с помощью многочлена Лагранжа

Выполнил: Давтян Давид А.

Группа P3201

Преподаватель: Перл Ольга Вячеславовна

Санкт-Петербург, 2020 г.

Описание метода

Некоторая функция $f(x)$ задана на отрезке $[a, b]$, разбитом на части $[x_i, x_{i+1}]$ таким образом, что $a = x_0 < x_1 < \cdots < x_n = b$.

Многочленом Лагранжа будет сумма полиномов

$$L_n(x) = \sum_{i=0}^{n} l_i(x)$$

где $l_i(x)$ – полином степени n, удовлетворяющий условию:

$$l_i = \begin{cases} y_i, \text{если } i = j \\ 0, \text{иначе} \end{cases}$$

При этом полиномы $l_i(x)$ составляются так:

$$l_i(x) = c_j(x - x_0)(x - x_1) \ldots (x - x_{i-1})(x - x_{i+1}) \ldots (x - x_n)$$

Здесь $(x - x_i)$ заменён на коэффициент c_i. Эти коэффициенты называются коэффициентами Лагранжа.

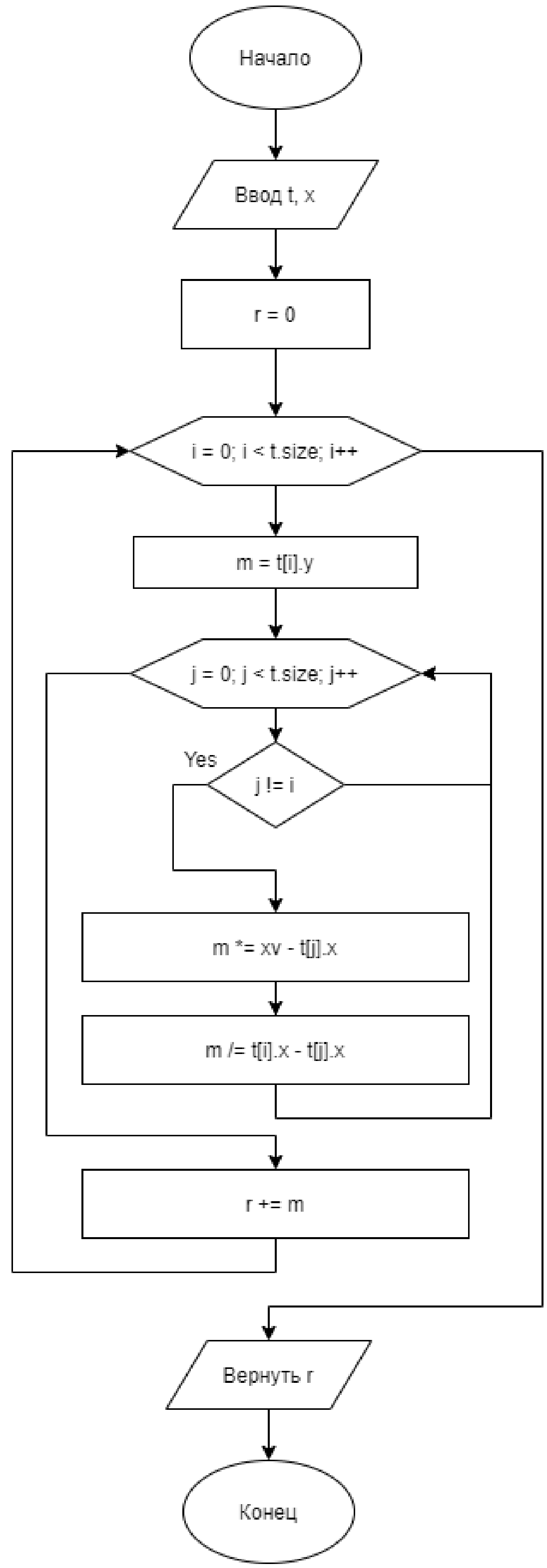
Пользуясь условием $L_n(x_i) = y_i$, найдём коэффициенты $c_0, c_1 \ldots, c_n$.

$$c_i = \frac{y_i}{(x_i - x_0)(x_i - x_1) \ldots (x_i - x_{i-1})(x_i - x_{i+1}) \ldots (x_i - x_n)}$$

Вместе с найденными коэффициентами многочлен Лагранжа будет записываться в виде:

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

Блок-схема

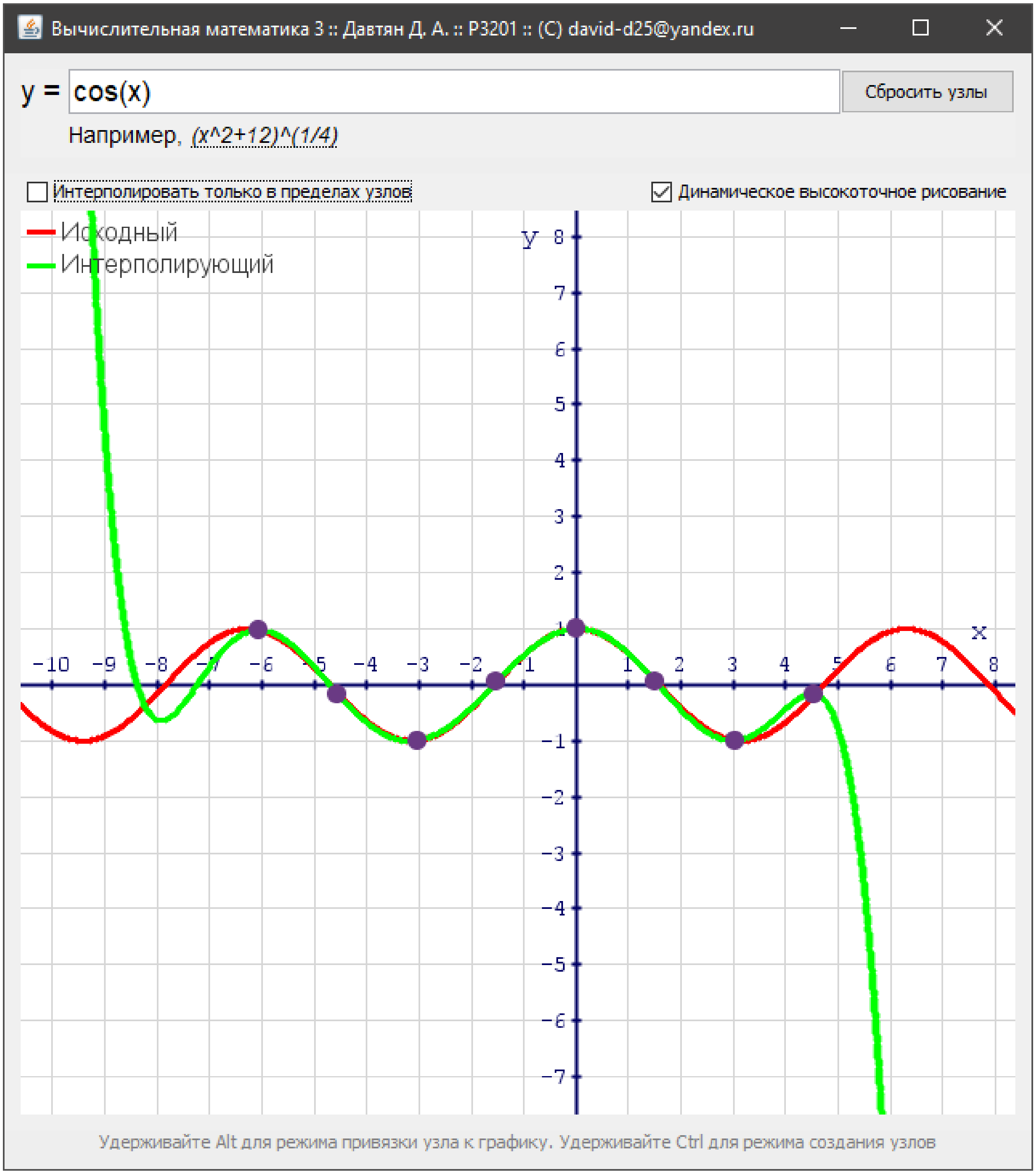

Начало
Ввод t, x
r = 0
i = 0; i < t.size; i++
m = t[i].y
j = 0; j < t.size; j++
Yes
j != i
m *= xv - t[j].x
m /= t[i].x - t[j].x
r += m
Вернуть r
Конец

Листинг численного метода

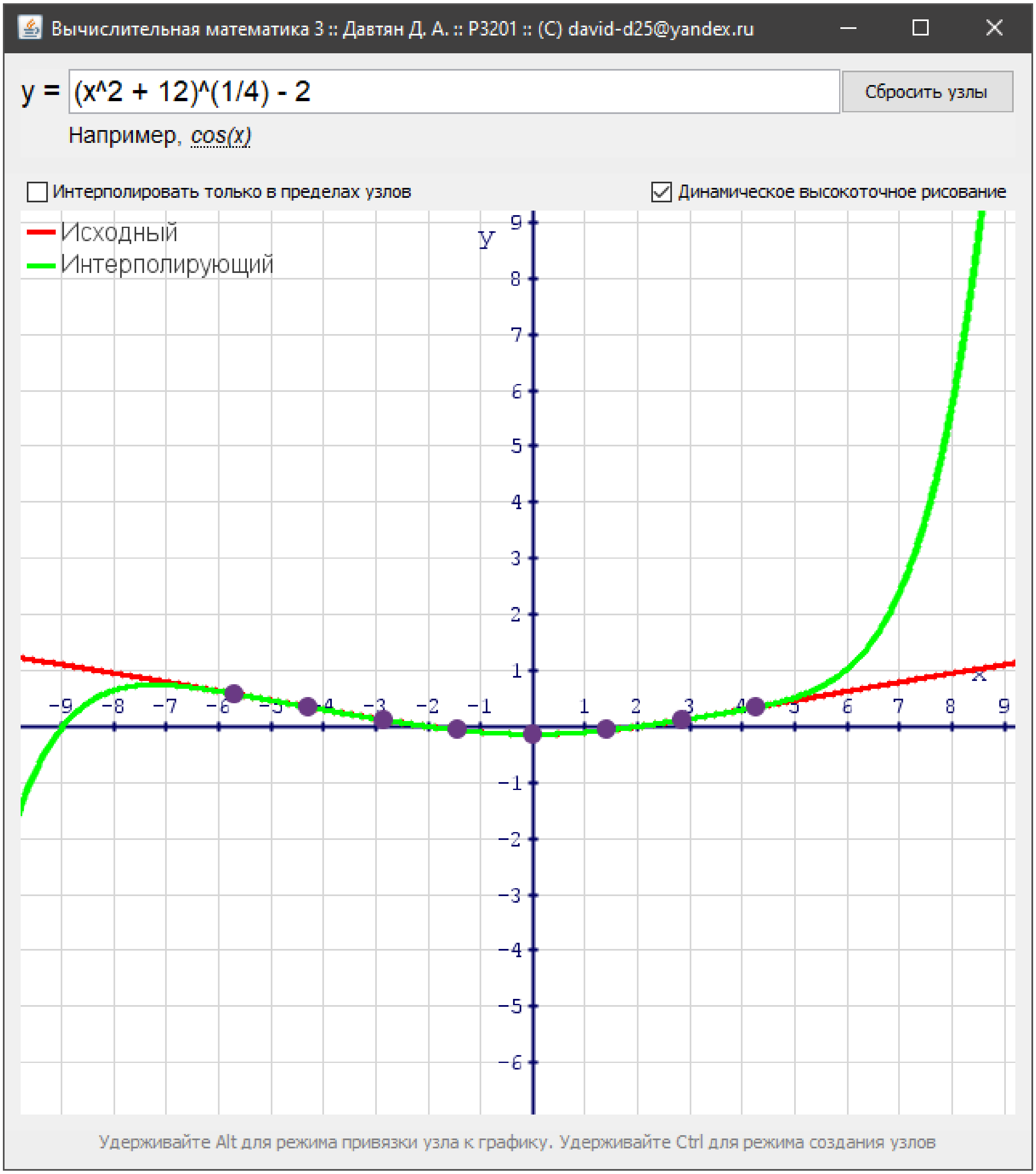
```java
package ru.david.compmath3.math.expression;

import ru.david.compmath3.math.model.XY;

import java.util.List;

public class LagrangeInterpolationExpression implements Expression {
    private List<XY> table;
    private Expression x;

    public LagrangeInterpolationExpression(List<XY> table, Expression x) {
        this.table = table;
        this.x = x;
    }

    @Override
    public Double value() {
        double result = 0;
        double xVal = x.value();

        for (int i = 0; i < table.size(); i++) {
            double monomial = table.get(i).y;
            for (int j = 0; j < table.size(); j++) {
                if (j != i) {
                    monomial *= xVal - table.get(j).x;
                    monomial /= table.get(i).x - table.get(j).x;
                }
            }
            result += monomial;
        }

        return result;
    }
}
```

Примеры

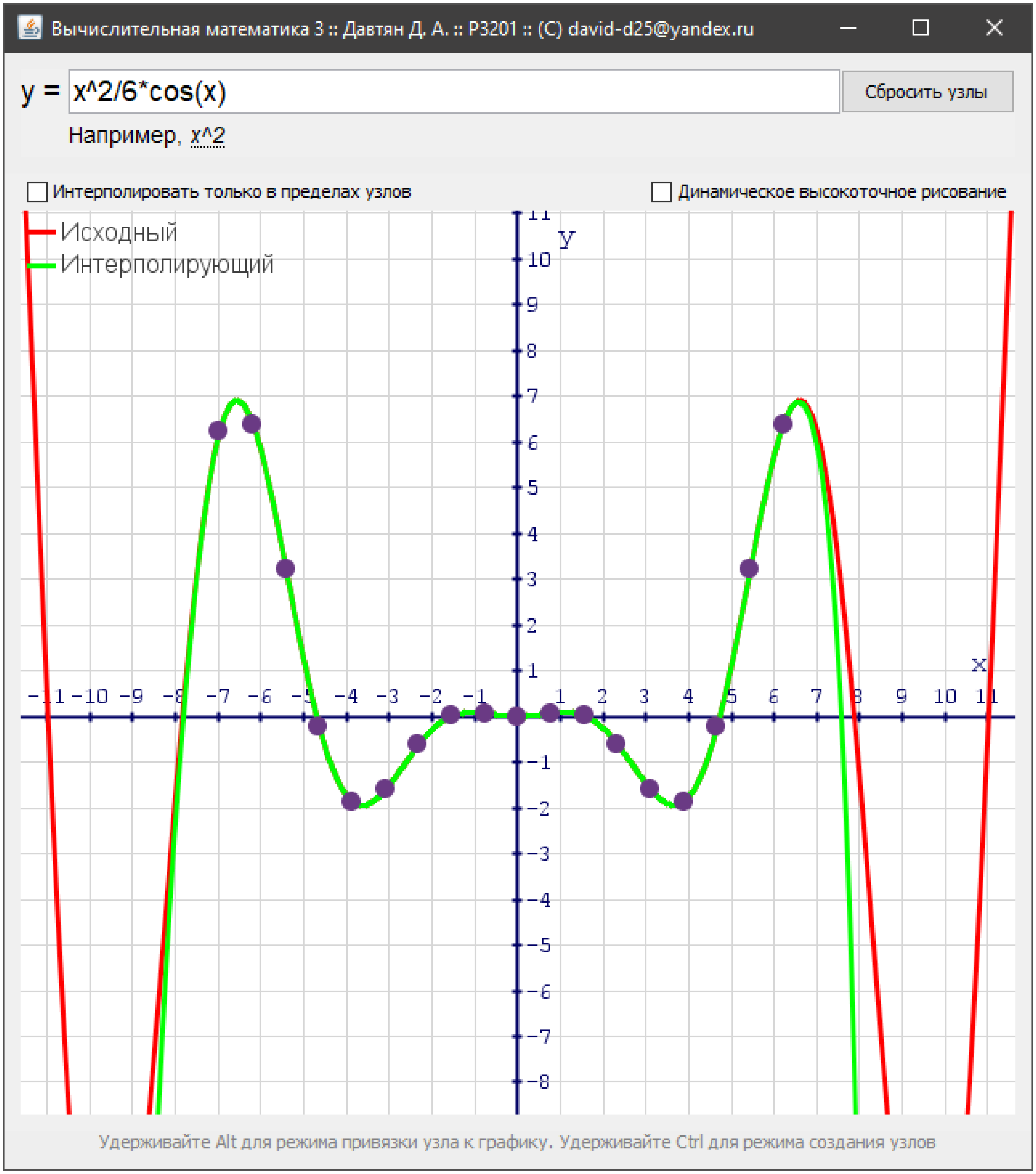
Пример 1 - cos(x)



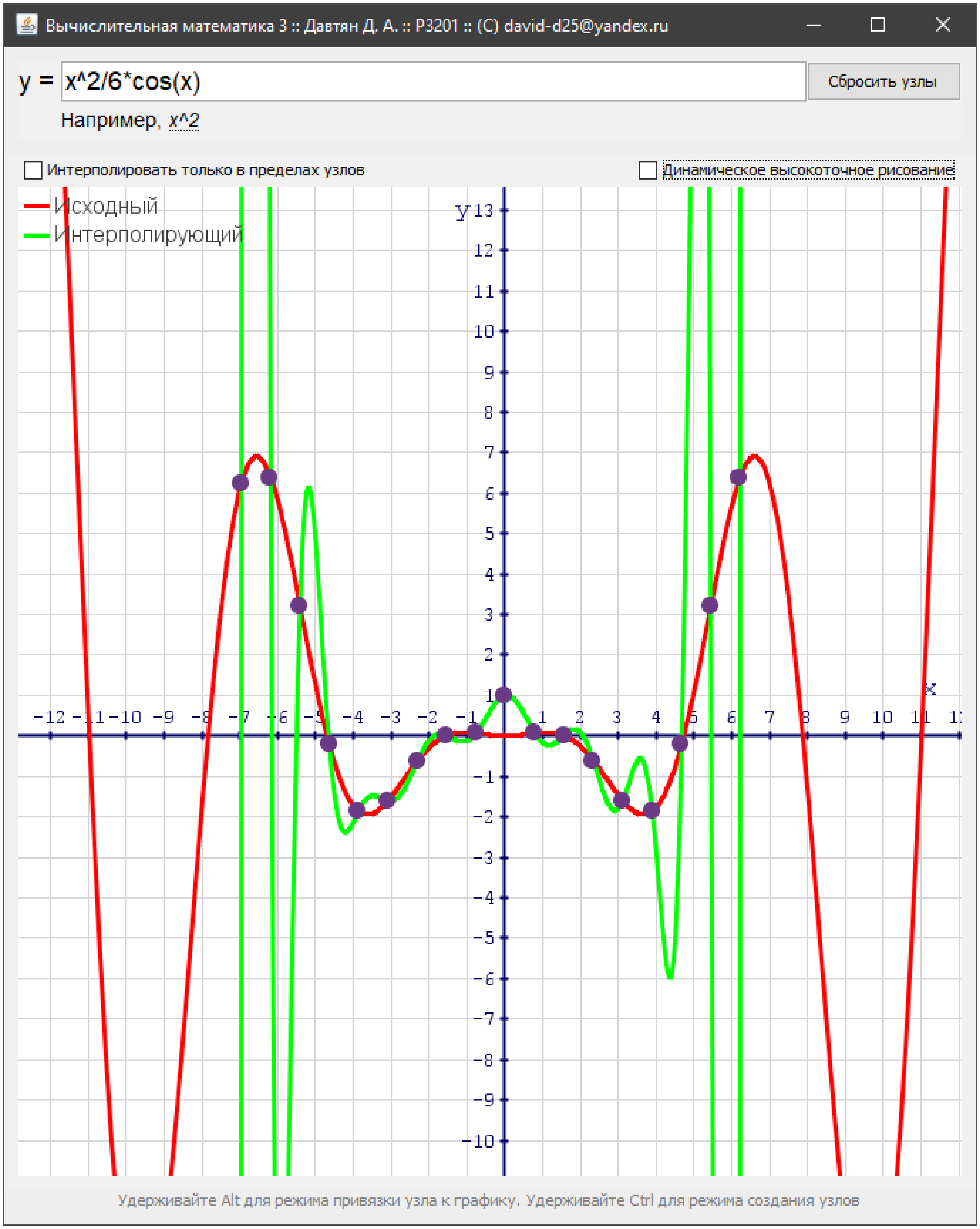
Пример 2 - (x^2 + 12)^(1/4) - 2



Пример 3 - x^2/6*cos(x)



Пример 3.1, в котором узел (0; 0) был установлен в (0; 1)



Вывод

Интерполяционный многочлен Лагранжа – одна из форм записи полинома степени n. Так как сам интерполяционный многочлен Лагранжа является многочленом, интерполирование с его помощью очень эффективно, когда исходная функция является многочленом (если количество узлов больше, чем степень исходного многочлена). Многочлен Лагранжа можно применить тогда, когда между узлами разное расстояние. В методе Ньютона, наоборот, так сделать не получится: узлы должны быть равноотстоящие. Однако метод Ньютона имеет следующее преимущество: метод Лагранжа требует перерасчитывать все коэффициенты при добавлении нового узла, тогда как метод Ньютона позволяет добавить к уже существующему многочлену одно слагаемое. Что касается интерполирования кубическими сплайнами, они представляют собой один из способов кусочно-полиноминальной (в отличие от методов Лагранжа и Ньютона, являющихся глобальными) интерполяции, когда отрезок разбирвают на частичные отрезки и на каждом из них приближенно заменяют исходную функцию многочленом невысокой степени. Их удобно применять, когда исходная функция является тригонометрической или периодической. Что касается сравнения с методом аппроксимации, то следует обратить внимание на разницу в постановке задач аппроксимации и интерполяции: интерполирующая функция должна принадлежать к определенному классу и в точках $x_0 \dots x_n$ принимать те же значения, что и исходная функция. Для аппроксиманта это требование не обязательно, но должен выполняться критерий наилучшего приближения. В методе наименьших квадратов после выбора класса аппроксимирующей функции $f(x_i, A, B, C, \dots)$ строится сумма вида $Q = \sum_{i=1}^{n}[f(x_i, A, B, C, \dots) - y_i]^2$. Исходными значениями параметров $A, B, C, \dots$ полагаются числа, обеспечивающие минимум суммы Q.