Лабораторная работа №2

Построение графиков функций

Цель работы: разработка модуля для построения графиков функций на декартовой плоскости в среде Microsoft Visual Studio.

Порядок выполнения работы

1. Разработать визуальный компонент Chart для построения графиков функций одной переменной на декартовой плоскости.
2. Протестировать и отладить программу с различными функциями.

Требования к программе

1. В качестве входных данных пользователь вводит интервал по оси абсцисс , на котором строится график. Программа при этом определяет автоматически, в каких пределах изменяются значения *y*.
2. Для разметки осей необходимо использовать параметр *h* – предпочтительный шаг сетки, измеряемый в пикселях. Данный параметр достаточно задать константой. Шаг сетки в мировой системе координат (в системе координат самого графика) должен выбираться автоматически из 1×10n, 2×10n, 5×10n таким образом, чтобы при пересчёте на пиксели быть наиболее близким к *h*. Значения *xi*, *yj*, на которых размечаются линии сетки должны нацело делиться на размер шага.
3. Программа должна при изменении размера окна пересчитывать координаты разметок и перестраивать график.
4. Числовые подписи к разметкам (линиям сетки) должны быть выровнены строго по центру разметки.
5. Ограничить область значений функций большими константами *y*max, *y*min (для построения, например, гиперболы вблизи *x*=0).

Приблизительное описание классов

// Класс - кривая

public class Curve

{

func F; // Указатель на функцию

Point[] screenPoints; // Список точек

Color color;

Curve (func, Color color); // Конструктор

void Draw (...); // Функция для рисования графика

void Reset (...); // Пересчитать точки screenPoints при изменении размера окна, инициализации и др.

...

}

// Класс - ось

public class Axis

{

Axis(Color color); // Конструктор

**/\* Функция для вычисления шагов для разметки сетки в мировой системе координат k и экранной h.**

**Входные параметры: k = width / (b - a). width – ширина поля вывода в пикселях. [a,b] – интервал в мировой системе координат, в котором меняется x или y.**

**h – предпочтительный шаг сетки в пикселях.**

**Выходные параметры: k = m×10n – полученный шаг в мировой системе координат, h – в экранной. \*/**

void Step(ref float k, ref float h, out int m, out int n);

}

// Класс – ось абсцисс

public class AbscissaAxis : Axis

{

AbscissaAxis(Color color, float min, float max) : base(color);

Draw(...);

}

// Класс – ось ординат

public class OrdinateAxis : Axis

{

public OrdinateAxis(Color color) : base(color);

Draw(...); **// Важно учесть, что ширина поля вывода подписей для данной оси определяется перед построением графика. Ширину текста можно определять функцией MeasureString (**<http://msdn.microsoft.com/ru-ru/library/9bt8ty58(v=vs.110).aspx>).

}

// Класс для вывода графиков, унаследован от UserControl

public class Chart : UserControl

{

Curve curve;

AbscissaAxis xAxis;

OrdinateAxis yAxis;

...

}

Для вертикального или горизонтального выравнивания текста разметок по центру нужно соответствующим образом задать последний аргумент format функции DrawString(string s, Font font, Brush brush, PointF point, StringFormat format):

format = new StringFormat();

format.Alignment = StringAlignment.Far;

format.LineAlignment = StringAlignment.Center;

xFormat = new StringFormat();

xFormat.Alignment = StringAlignment.Center;

Необязательные требования

1. Реализовать построение нескольких графиков на одной плоскости.
2. Если начало координат (x=0; y=0) попадает в область видимости, то выводить оси и разметку из данной точки (по умолчанию оси можно рисовать из нижнего левого угла).
3. Реализовать ввод функций в символьном виде во время выполнения программы. Для этого необходимо изучить материал по динамической компиляции и рефлексии на C# (функцию eval для Java) или другом языке программирования: <http://forum.codenet.ru/q26577/eval+%D0%B2+%D0%A1%23>;

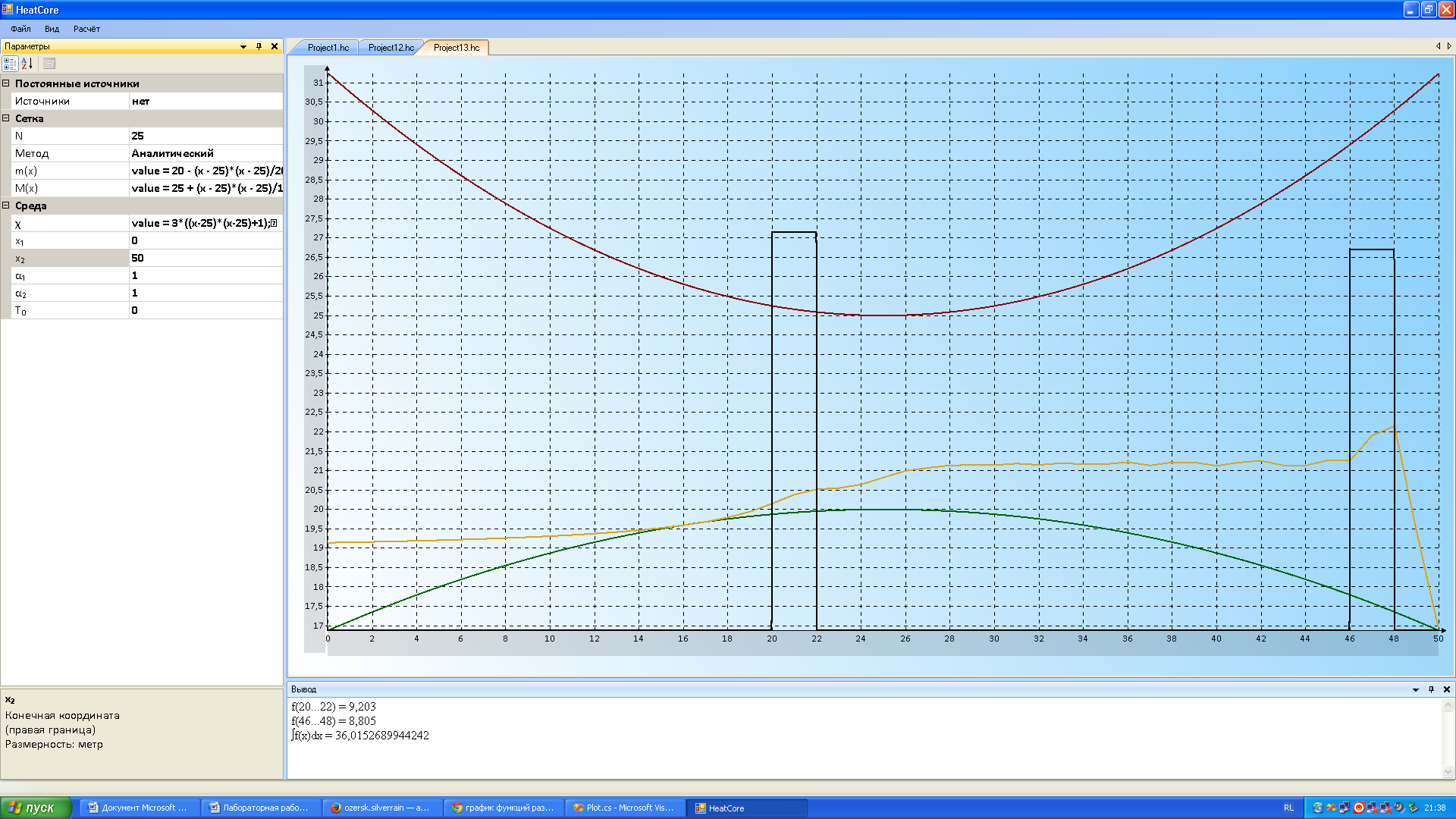
<http://www.codeproject.com/Articles/11939/Evaluate-C-Code-Eval-Function>;

<http://www.beyondlinux.com/2011/08/07/3-method-to-evaluate-expressions/>;

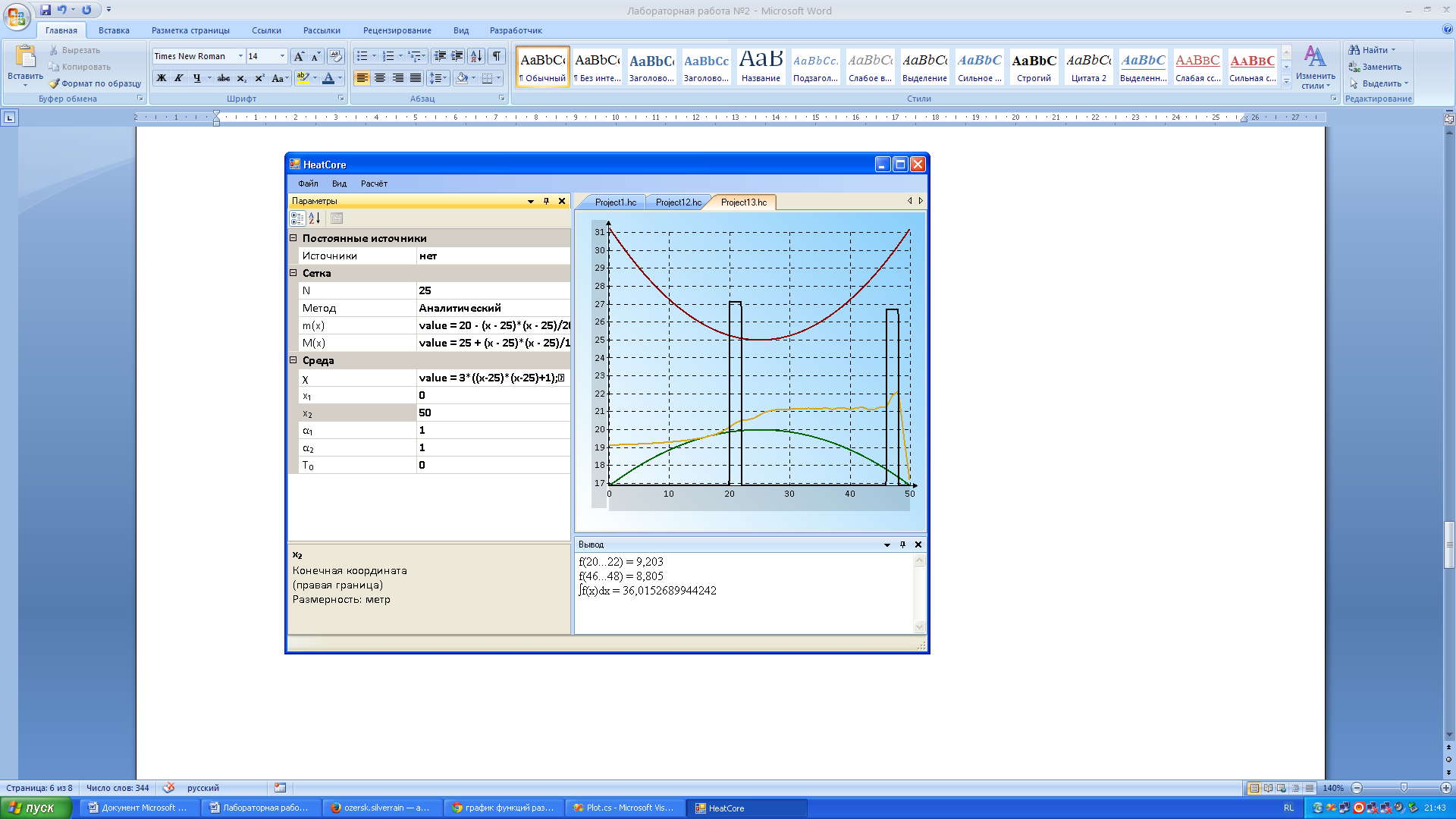
<http://javascript.ru/eval>;

<http://nkoksharov.blogspot.ru/2008/12/rhino-janino.html>.

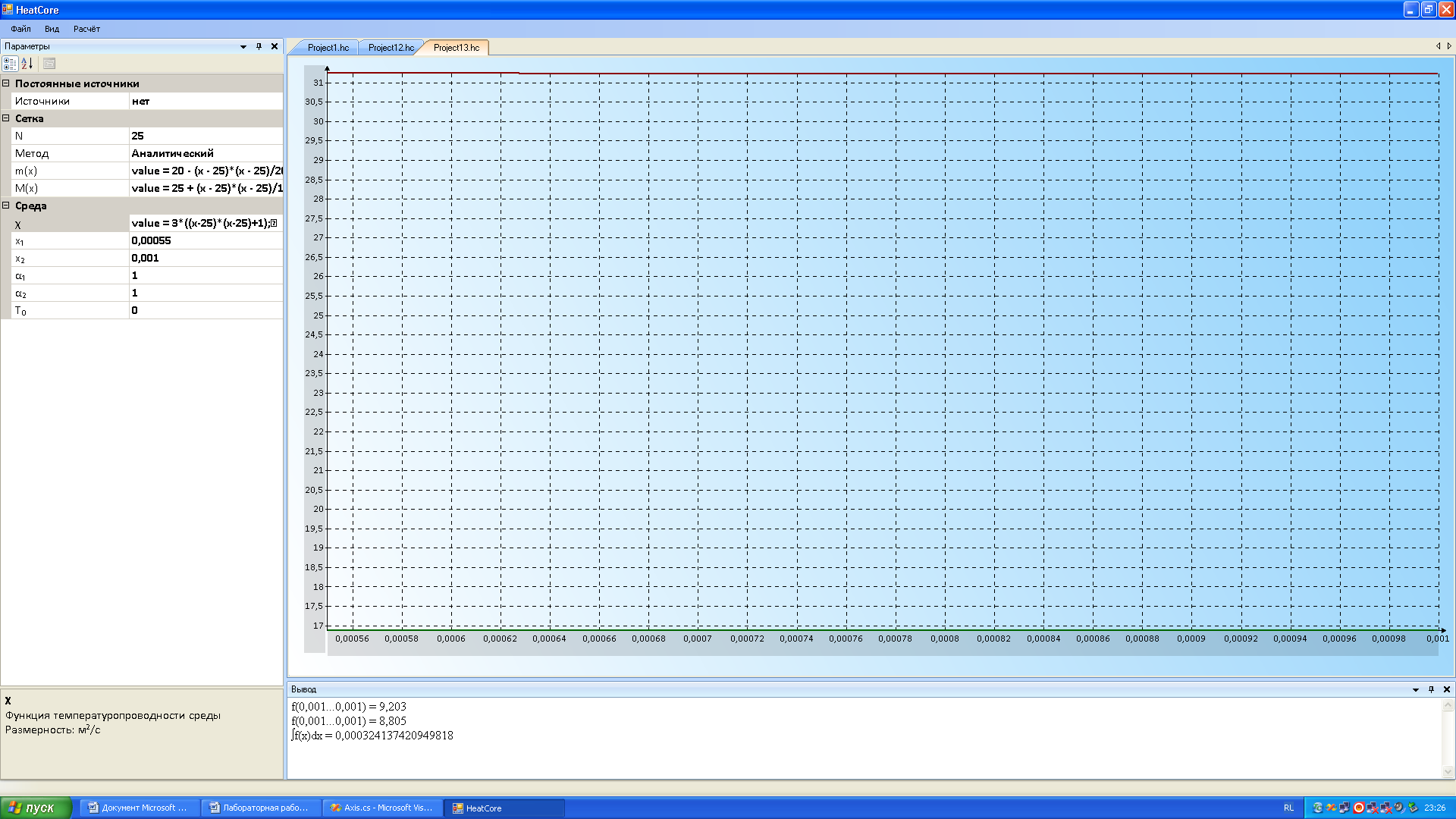
Пример. Мировой шаг по оси *x* - 2×100, *y* - 5×10-1. Предпочтительный шаг сетки *h* = 30 пикселей.



Поведение того же графика при изменении размера окна (шаг в мировой системе координат увеличивается, экранный остаётся приблизительно тем же):



Пример разметки с маленьким шагом по оси *x*:



Содержание отчёта

1. Название темы.
2. Цель работы.
3. **Схема структуры классов.**
4. Текст программы.
5. Результат работы программы (снимки экрана).