**Курсовая работа**

по дисциплине «Разработка клиент-серверных приложений»

Москва, 2022

**Цель:** разработать скрипт к заданию на языке программирования php, создать блок-схемы и привести примеры выполнения программы.

Задание:

Вариант-17

Разработать скрипт, создающий точечную диаграмму (в графическом виде) по следующим исходным параметрам: width, height - ширина и высота изображения (задаются через форму) data.dat - файл, содержащий данные (на каждой строке два числа, разделенные точкой с запятой: первое число -величина, откладываемая на горизонтальной оси, второе число откладывается на вертикальной оси).

**Теоретическая часть**

**Библиотека GD**

Язык PHP достаточно разнообразен и его использование не ограничивается только созданием скриптов, выполняющим различные манипуляции с данными. Он также позволяет работать с различными графическими элементами, в частности с изображениями, разных форматов. Для этих целей написана, специальная библиотека, под названием GD, которая расширяет его стандартный функционал. Данная библиотека, чем-то похожа на небольшой графический редактор, используя который, мы сможем легко обрабатывать изображения, используя только язык PHP.

Библиотека GD, позволяет создавать новые изображения, редактировать существующие, копировать одни изображения на другие, изменять размеры, а также наносить текст на изображения. PHP может удовлетворить практически все ваши основные потребности в управлении изображениями, используя библиотеку GD - сокращение для Graphic Draw. Если вы работаете в Windows, вы можете включить файл php\_gd2.dll в качестве расширения в php.ini. Но в нашем случае мы используем XAMPP. Для этого находим файл php\_gd2.dll в каталоге xampp\php\ext.

Также можно проверить, установлена ли GD в нашей системе, с помощью функции phpinfo().  GD входит в состав PHP начиная с PHP версии 4.3, поэтому вполне вероятно, что на нашем PHP уже установлен GD. Мы можем запустить phpinfo(), чтобы увидеть, установлен ли GD — на приведенном ниже экране показано, что он установлен.

Изображение выглядит как текст, снимок экрана, число, Шрифт

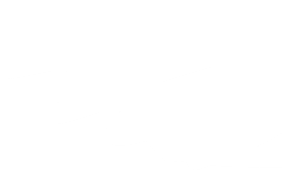
Автоматически созданное описание

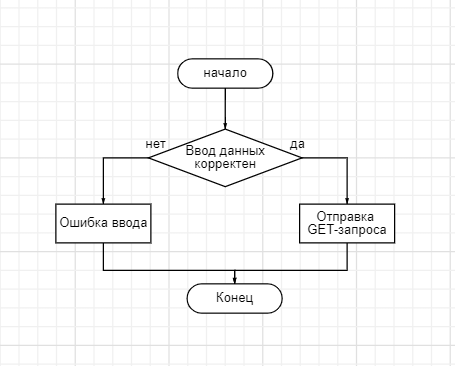
Рис.1-Информация по библиотеке GD

Если возникают какие-либо вопросы по установке, мы всегда можем посетить страницу требований, чтобы узнать больше информации.

**Ход работы:**

**Блок-схема формы**



****



**Листинг формы**

using System;

using System.Text;

using System.Windows.Forms;

using System.Net;

using System.IO;

namespace WindowsFormsApp4

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

int height, width;

if (!int.TryParse(textBox1.Text, out height) | !int.TryParse(textBox1.Text, out width))

{

throw new IOException("Неправильный ввод");

}

string uri = "http://127.0.0.1/katyzina/kurs/kursovaya.php?h=" + height + "&w=" + width;

HttpWebRequest request = (HttpWebRequest)WebRequest.Create(uri);

HttpWebResponse response = (HttpWebResponse)request.GetResponse();

StreamReader reader = new StreamReader(response.GetResponseStream(), Encoding.UTF8);

Console.WriteLine(reader.ReadToEnd());

Console.ReadLine();

pictureBox1.Load("http://127.0.0.1/katyzina/kurs/2.png");

}

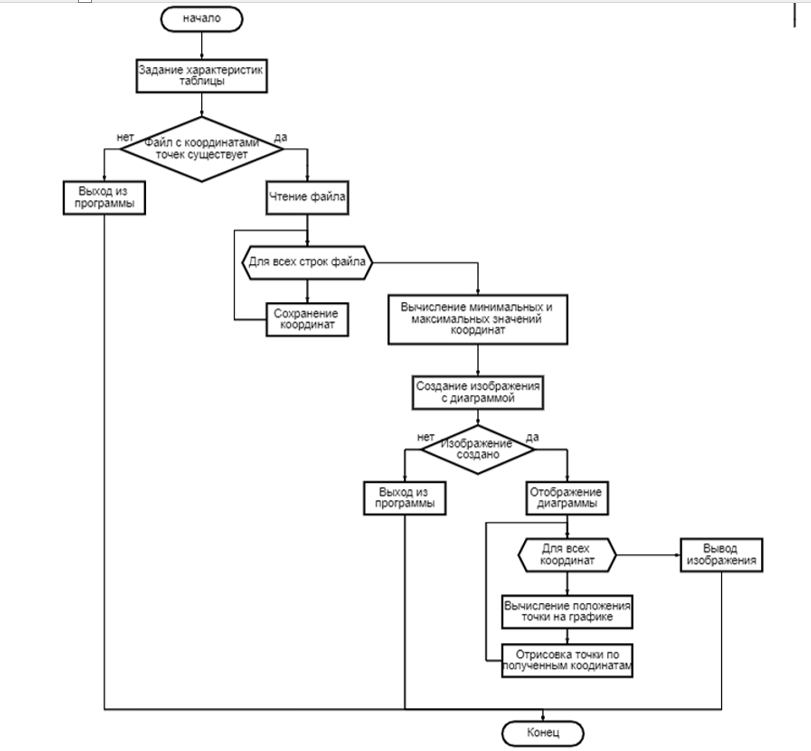
catch (Exception ex) { MessageBox.Show(ex.Message); }

}

}

}

**Блок-схема скрипта**

****

**Листинг скрипта**

<?php

//Файл данных

$filename = "data.txt";

//Высота изображения

$height = $\_GET['h'];

//Ширина изображения

$width = $\_GET['w'];

// Жирность точки

$point = 3;

//Считываем содержимое файла

if (!is\_file($filename)) exit("Отсутсвует файл данных");

$arr = file($filename);

//Записываем значения x и y в массив

$arr\_x = array();

$arr\_y = array();

// Определяем количество строк в файле

$number\_lines= count(file($filename));

foreach($arr as $line)

{

//Данные в текстовом файле записываем через ";"

$num = preg\_split("/[\;]+/",$line);

//Добавляем элементы в конец массива

array\_push($arr\_x, $num[0]);

array\_push($arr\_y, $num[1]);

}

//Находим коэффициенты для X

$min\_x = min($arr\_x);

$max\_x = max($arr\_x);

//Находим максимальную точку по модулю (ox)

$max\_x=((abs($min\_x) > abs($max\_x))?(abs($min\_x)):abs($max\_x));

//Находим коэффициенты для Y

$min\_y = min($arr\_y);

$max\_y = max($arr\_y);

//Максимальная точка по модулю (OY)

$max\_y=((abs($min\_y) > abs($max\_y))? (abs($min\_y)):abs($max\_y));

// Расчитаем максимальное значение по оси ординат

$height\_max = $height / 2;

// Расчитаем максимальное значение по оси абсцисс

$width\_max = $width / 2;

// Находим коэффициент

$koef\_x =$width\_max/$max\_x;

$koef\_y =$height\_max/$max\_y;

// Рисуем диаграмму

$img = imageCreate($width + 10,$height + 5);

if (!$img) exit("Не удалось создать изображение");

// Формируем и создаем цвета

$white = imagecolorallocate($img, 255, 255, 255);

$black = imagecolorallocate($img, 0, 0, 0);

// Фон делаем белым

imagefill($img, 1, 1, $white);

// Рисуем оси

imageLine($img, 0, $height\_max, $width, $height\_max, $black);

imageLine($img, $width\_max, 5, $width\_max, $height, $black);

// Вычисляем положение точки на графике

for ($i=0; $i < $number\_lines; $i++)

{

$image\_x = (float)($width\*($arr\_x[$i]\*$koef\_x - $width/2\*(-1))/($width\_max - $width/2\*(-1)))+5;

$image\_y = (float)($height\*($height\_max - $arr\_y[$i]\*$koef\_y)/($height\_max- $height/2\*(-1)))+5;

// Рисуем точку

imagefilledrectangle($img, $image\_x - $point, $image\_y - $point, $image\_x + $point, $image\_y + $point, $black);

}

//Создаем текст для оси

$txt\_X = "$max\_x";

$txt\_Y = "$max\_y";

// Функции нанесения текста

imagestring($img, 1, $width - 10, $height\_max + 5, $txt\_X, $black);

imagestring($img, 1, $width\_max - 9, 0, $txt\_Y, $black);

imagestring($img, 1, $width\_max - 7, $height\_max+ 3, '0', $black);

//Сохраняем png изображение

imagepng($img, "2.png");

//Выводим изображение

echo ('<img src="2.png">');

?>

**Пример выполнения работы**

Изображение выглядит как текст

Автоматически созданное описание

Рис.1-Пример текстового файла, содержащего данные

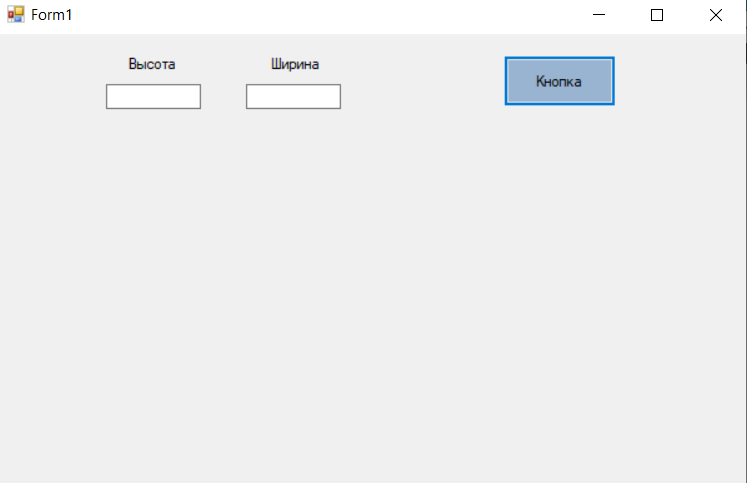


Рис.2-Пример интерфейса формы

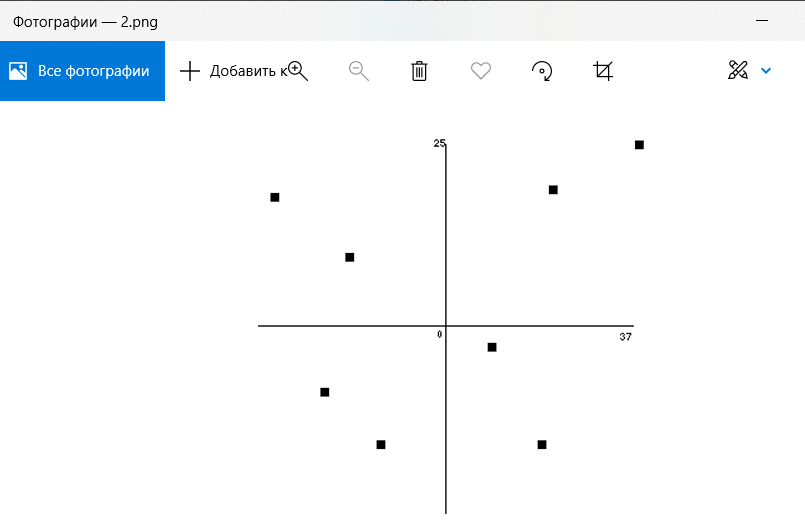


Рис.2-Пример сохраненного точечного графика в формате png

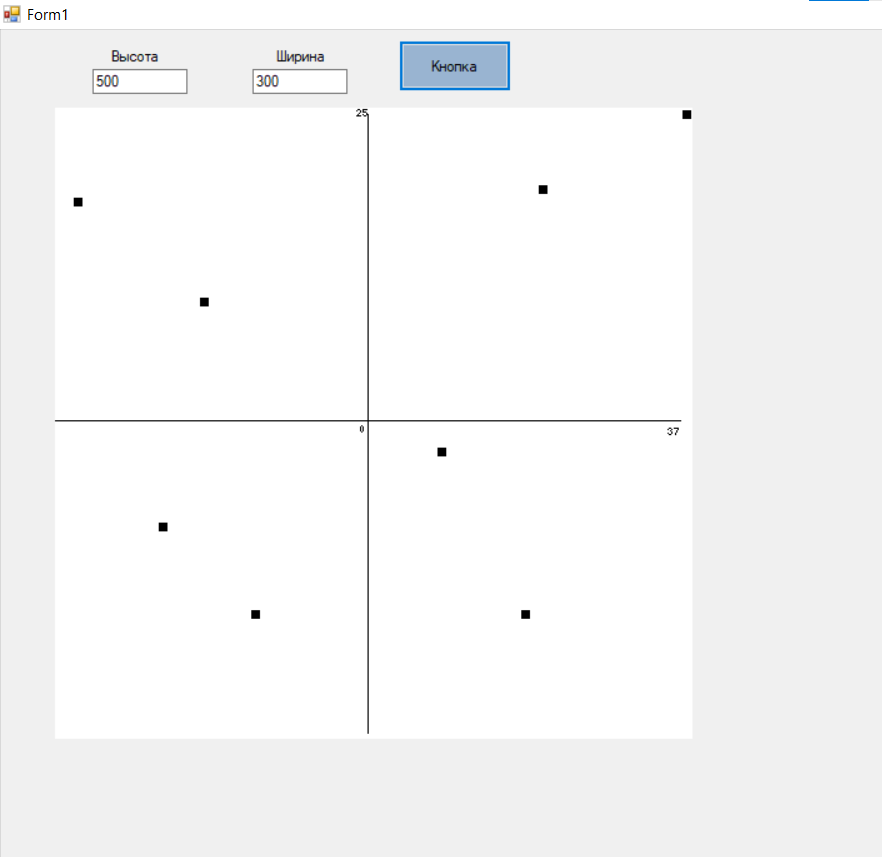


Рис.4-Пример выполнения программы

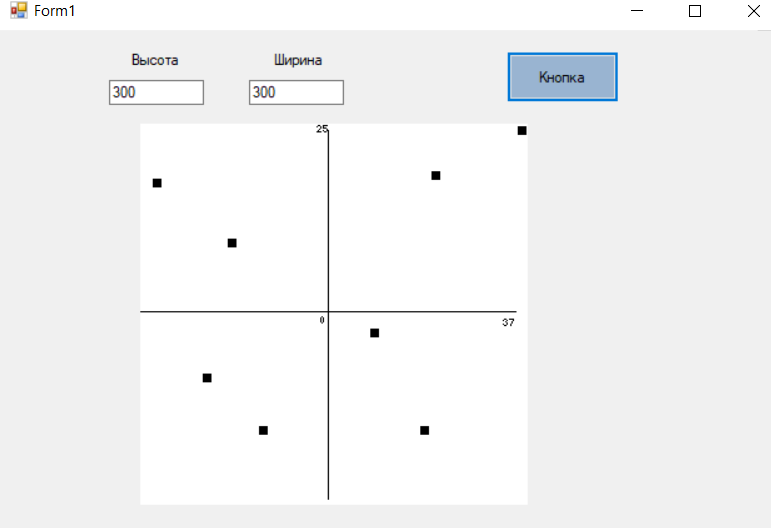


Рис.5-Пример выполнения программы

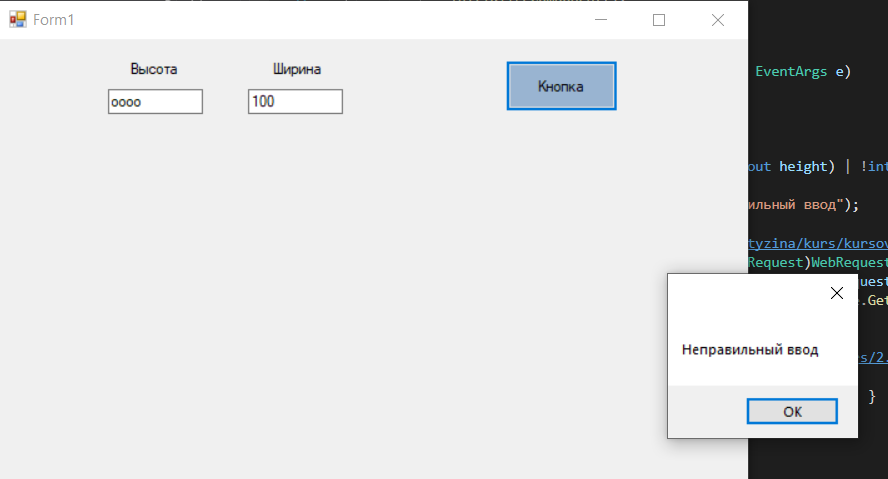


Рис.6- Пример вывода ошибки, при неправильном заполнении поля

**Вывод:** Я разработала скрипт к моему заданию на языке программирования php, создала блок-схемы и привела примеры выполнения программы. Научилась строить точечную диаграмму по ее исходным параметрам: ширина и высота изображения с помощью языка программирования php.