Федеральное агентство по образованию

ФГБОУ ВО Уфимский государственный авиационный технический

университет

Кафедра ВМиК

ЛАБОРАТОРНАЯ РАБОТА №3

По дисциплине: «Инженерная и компьютерная графика»

«Программирование WEB-Графики»

Выполнил: студент группы ПРО-222

Сафронов И.А.

Проверил: Котельников В.А.

Уфа 2021

1. **Цель работы:** ознакомиться с Web-графикой.
2. **Задачи:**

* Открыть проект Canvas Paint.
* Реализовать функцию рисования линий, кругов, правильных многоугольников.

1. **Ход** **работы**

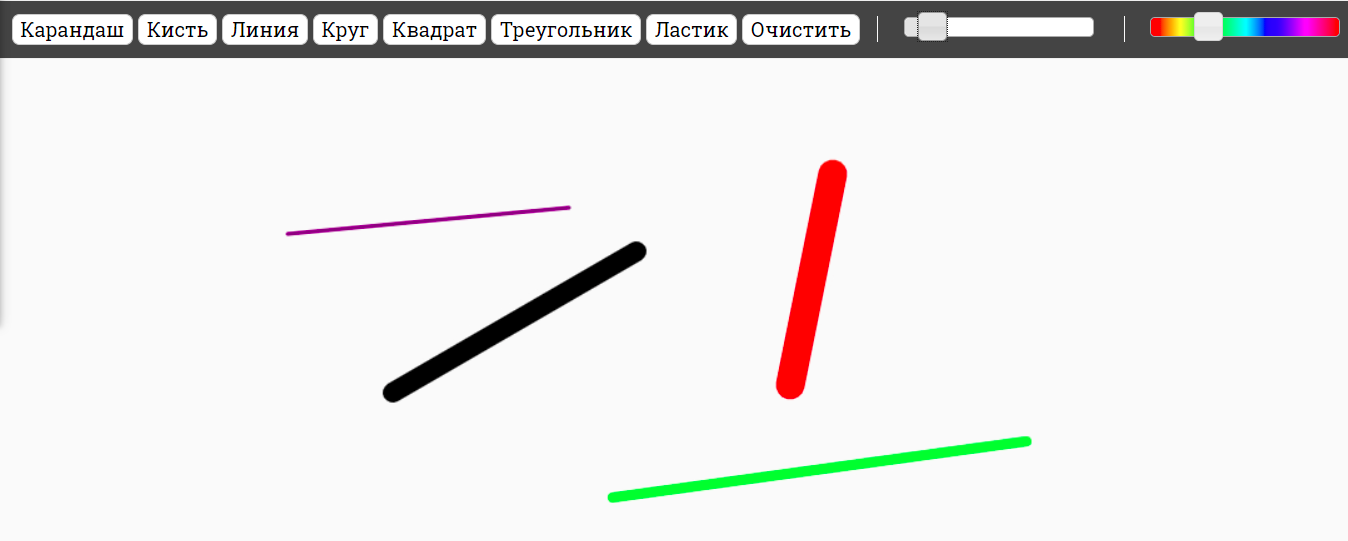


Рис. 1 Линии

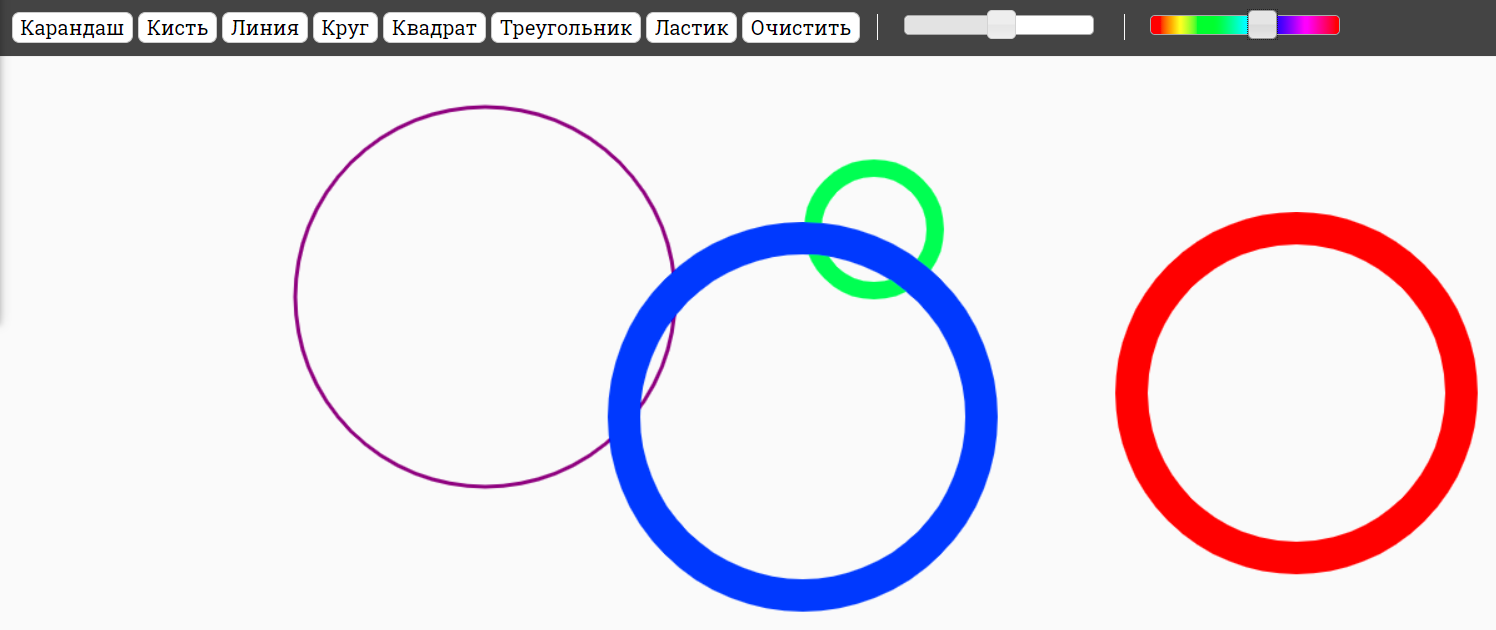


Рис. 2 Круги

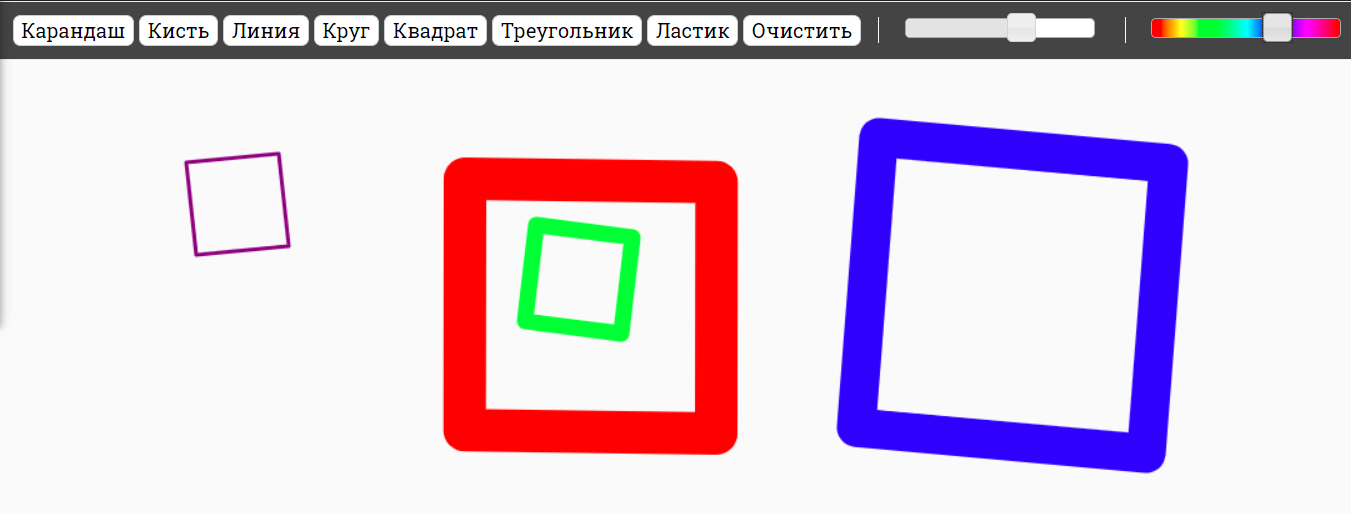


Рис. 3 Квадраты

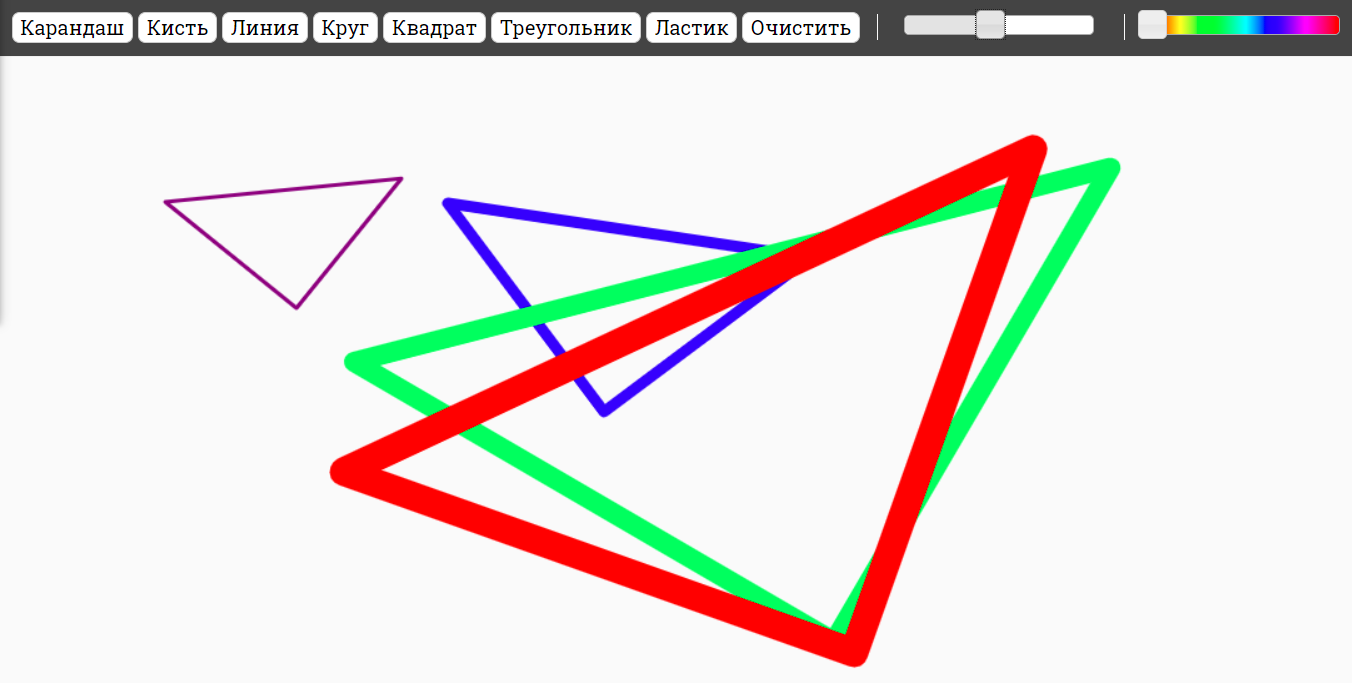


Рис. 4 Треугольники

1. **Вывод:**

В рамках лабораторной работы я ознакомился с Web-графикой.

Приложение 1. Листинг программы:

**index.html:**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Paint</title>

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="http://code.jquery.com/ui/1.10.3/themes/smoothness/jquery-ui.css" type="text/css">

<link rel="stylesheet" href="css/main.css" type="text/css">

</head>

<body>

<div class="tools">

<button class="tools-pen" type="button">Карандаш</button>

<button class="tools-brush" type="button">Кисть</button>

<button class="tools-line" type="button">Линия</button>

<button class="tools-circle" type="button">Круг</button>

<button class="tools-square" type="button">Квадрат</button>

<button class="tools-triangle" type="button">Треугольник</button>

<button class="tools-eraser" type="button">Ластик</button>

<button class="tools-clear" type="button">Очистить</button>

<div class="tools-subset">

<div class="tools-size"></div>

</div>

<div class="tools-subset">

<canvas id="tools-colour-ref" width="100" height="1"></canvas>

<div class="tools-colour"></div>

</div>

</div>

<div id="paintcontainer" class="paint-container"></div>

<script src="http://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"></script>

<script src="http://code.jquery.com/ui/1.10.3/jquery-ui.js"></script>

<script src="js/sketch.min.js"></script>

<script src="js/main.js"></script>

</body>

</html>

**main.js:**

$(function() {

(function() {

var ref = document.getElementById('tools-colour-ref'),

ctx = ref.getContext('2d'),

grd = ctx.createLinearGradient(0, 0, 100, 0);

grd.addColorStop(0, '#FF0000');

grd.addColorStop(0.16, '#FFFF00');

grd.addColorStop(0.33, '#00FF00');

grd.addColorStop(0.50, '#00FFFF');

grd.addColorStop(0.66, '#0000FF');

grd.addColorStop(0.83, '#FF00FF');

grd.addColorStop(1, '#FF0000');

ctx.fillStyle = grd;

ctx.fillRect(0, 0, 100, 30);

})();

Sketch.create({

container: document.getElementById('paintcontainer'),

autoclear: false,

setup: function() {

var self = this;

this.\_tool = 'pen';

this.\_colour = 'purple';

this.\_size = 3;

this.\_active = {

status: false,

time: 0,

touches: []

};

this.lineCap = 'round';

this.lineJoin = 'round';

$('.tools-size').slider({

orientation: 'horizontal',

range: 'min',

min: 1,

max: 50,

value: 3,

slide: function(e, ui) {

self.\_size = ui.value;

}

});

$('.tools-colour').slider({

orientation: 'horizontal',

min: 0,

max: 100,

value: 80,

slide: function(e, ui) {

var x = $('#tools-colour-ref').width() \* (ui.value / 100),

colour = document.getElementById('tools-colour-ref').getContext('2d').getImageData(x, 0, 1, 1),

red = colour.data[0].toString(16),

green = colour.data[1].toString(16),

blue = colour.data[2].toString(16);

if(red.length === 1) red = '0' + red;

if(green.length === 1) green = '0' + green;

if(blue.length === 1) blue = '0' + blue;

self.\_colour = '#' + red + green + blue;

}

});

$('.tools-eraser').click(function() {

self.\_erase = true;

self.\_tool = 'pen';

});

$('.tools-pen, .tools-brush, .tools-line, .tools-circle, .tools-square, .tools-triangle').click(function() {

self.\_erase = false;

});

$('.tools-pen').click(function() {

self.\_tool = 'pen';

});

$('.tools-brush').click(function() {

self.\_tool = 'brush';

});

$('.tools-line').click(function() {

self.\_tool = 'line';

});

$('.tools-circle').click(function() {

self.\_tool = 'circle';

});

$('.tools-square').click(function() {

self.\_tool = 'square';

});

$('.tools-triangle').click(function() {

self.\_tool = 'triangle';

});

$('.tools-clear').click(function() {

self.clear();

});

flag = 1;

coordX = 0;

coordY = 0;

X2 = 0;

Y2 = 0;

},

update: function() {

},

mousedown: function() {

this.\_active.status = true;

this.\_active.time = this.now;

this.\_active.touches = this.touches;

if(this.\_tool == 'line'){

if(flag == 1)

{

coordX = event.pageX;

coordY = event.pageY - 50;

flag = 2;

X2 = 0;

Y2 = 0;

}

else{

flag = 1;

}

}

if(this.\_tool == 'circle'){

if(flag == 1)

{

coordX = event.pageX;

coordY = event.pageY - 50;

flag = 2;

}

else{

oX = event.pageX - coordX;

oY = event.pageY - coordY - 50;

if(Math.abs(oX) > Math.abs(oY)){

rad = Math.abs(oX);

}

else{

rad = Math.abs(oY);

}

this.beginPath();

this.lineWidth = this.\_size;

this.strokeStyle = this.\_colour;

this.arc(coordX + oX, coordY + oY, rad, 0, 2 \* Math.PI);

this.stroke();

this.closePath();

flag = 1;

}

}

if(this.\_tool == 'square'){

if(flag == 1)

{

firstX = event.pageX;

firstY = event.pageY - 50;

flag = 2;

}

else{

thirdX = event.pageX;

thirdY = event.pageY - 47;

secondX = ((thirdX - firstX) / 2) \* Math.cos(0) - ((thirdY - firstY) / 2) \* Math.sin(-30) + firstX;

secondY = ((thirdX - firstX) / 2) \* Math.sin(-30) + ((thirdY - firstY) / 2) \* Math.cos(0) + firstY;

fourthX = ((thirdX - firstX) / 2) \* Math.cos(0) - ((thirdY - firstY) / 2) \* Math.sin(30) + firstX;

fourthY = ((thirdX - firstX) / 2) \* Math.sin(30) + ((thirdY - firstY) / 2) \* Math.cos(0) + firstY;

this.beginPath();

this.lineWidth = this.\_size;

this.strokeStyle = this.\_colour;

this.moveTo(firstX, firstY);

this.lineTo(secondX, secondY);

this.lineTo(thirdX, thirdY);

this.lineTo(fourthX, fourthY);

this.closePath();

this.stroke();

flag = 1;

}

}

if(this.\_tool == 'triangle'){

if(flag == 1)

{

firstX = event.pageX;

firstY = event.pageY - 50;

flag = 2;

}

else{

secondX = event.pageX;

secondY = event.pageY - 50;

thirdX = (secondX-firstX)\*Math.cos(0)-(secondY-firstY)\*Math.sin(30)+firstX;

thirdY = (secondX-firstX)\*Math.sin(30)+(secondY-firstY)\*Math.cos(0)+firstY;

this.beginPath();

this.lineWidth = this.\_size;

this.strokeStyle = this.\_colour;

this.moveTo(firstX, firstY);

this.lineTo(secondX, secondY);

this.lineTo(thirdX, thirdY);

this.closePath();

this.stroke();

flag = 1;

}

}

},

mouseup: function() {

this.\_active.status = false;

if(this.\_tool == 'line' && flag == 2) {

this.\_active.status = true;

}

},

mousemove: function() {

if(!this.\_active.status) return;

this.fillStyle = this.strokeStyle = (this.\_erase ? '#fafafa' : this.\_colour);

for(var i = 0; i < this.touches.length; i++) {

var touch = this.touches[i];

if(this.\_tool == 'pen' || this.\_tool == 'brush') {

if(this.\_tool == 'brush') {

var ratio = Math.round((this.now - this.\_active.time) / 100)/100;

ratio = ratio\*4;

if(ratio > 0.9) ratio = 0.9

this.lineWidth = this.\_size \* (1 - ratio);

} else {

this.lineWidth = this.\_size;

}

this.beginPath();

this.moveTo(touch.ox, touch.oy);

this.lineTo(touch.x, touch.y);

this.stroke();

this.closePath();

} else if(this.\_tool == 'line' && flag == 2) {

if (X2 != 0){

this.beginPath();

this.lineWidth = this.\_size+2;

this.strokeStyle = '#FAFAFA';

this.moveTo(coordX, coordY);

this.lineTo(X2, Y2 - 50);

this.stroke();

this.closePath();

}

X2 = event.pageX;

Y2 = event.pageY;

this.beginPath();

this.lineWidth = this.\_size;

this.strokeStyle = this.\_colour;

this.moveTo(coordX, coordY);

this.lineTo(X2, Y2 - 50);

this.stroke();

this.closePath();

}

}

}

});

});

Приложение 2. Скриншот коммитов

Изображение выглядит как текст

Автоматически созданное описание

Рис. 5 Скриншот коммитов

Приложение 3. Ссылка на GitHub

<https://github.com/MenikUG/1_laba_Graphics>