

Учреждение образования  
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Кафедра интеллектуальных информационных технологий  
Отчёт  
по курсу «Графический интерфейс интеллектуальных систем»

Выполнил студент группы 121701: Дятлов Е.А.  
Проверила: Жмырко А.В.

МИНСК  
2023

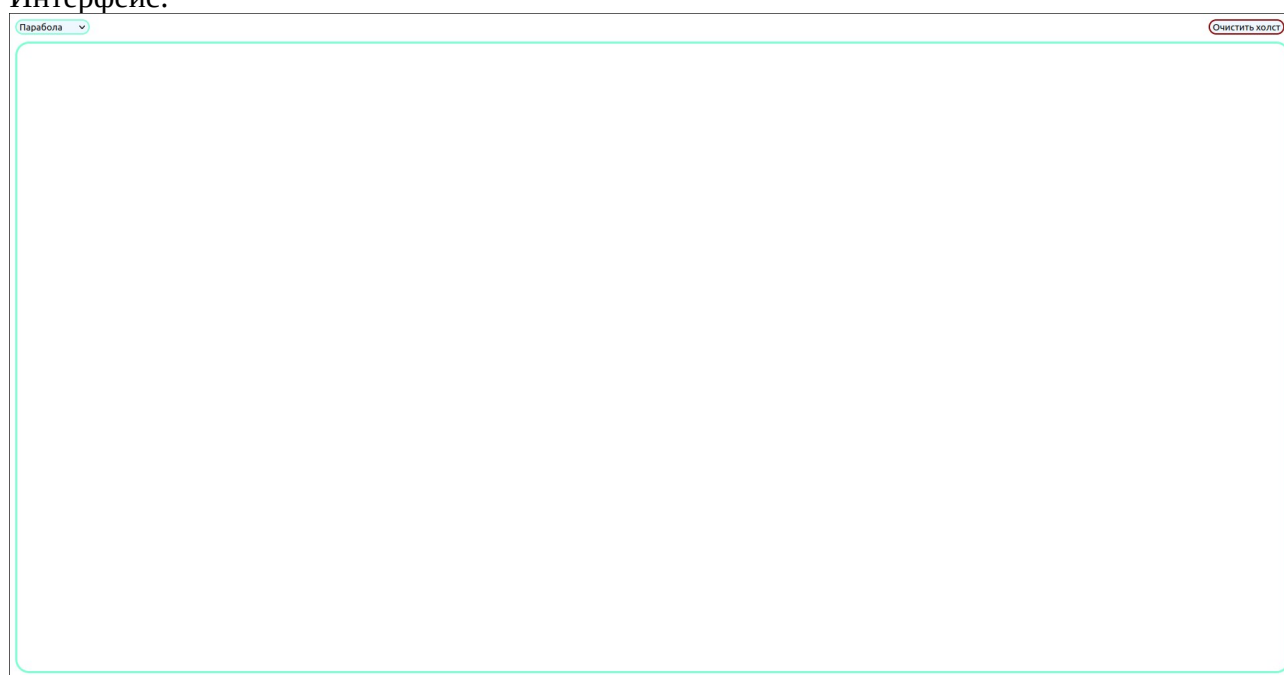
Тема: Генерация кривых второго порядка.

Цель: Разработать элементарный графический редактор, реализующий построение линий второго порядка: окружность, эллипс, гипербола, парабола. Выбор кривой задается из пункта меню и доступно через панель инструментов «Линии второго порядка». В редакторе кроме режима генерации линий второго порядка в пользовательском окне должен быть предусмотрен отладочный режим, где отображается пошаговое решение на дискретной сетке.

Теоретические сведения:

Кривые второго порядка являются важным классом геометрических фигур, определяемых квадратичными уравнениями. Они включают в себя окружности, эллипсы, гиперболы и параболы. Каждая из этих кривых имеет свои математические характеристики и алгоритмы для их генерации.

Интерфейс:



Листинг кода:

```
app.js
const canvas = document.getElementById('canvas');
const ctx = canvas.getContext('2d');
let centerX, centerY;
let isDrawing = false;

canvas.addEventListener('mousedown', startDrawing);
function startDrawing(event) {
  isDrawing = true;
  centerX = event.clientX - canvas.offsetLeft;
  centerY = event.clientY - canvas.offsetTop;
}

canvas.addEventListener('mousemove', draw);
function draw(event) {
  if (!isDrawing) return;
```

```
const curveType = document.getElementById('curveType').value;
const a = Math.abs(event.clientX - canvas.offsetLeft - centerX);
const b = Math.abs(event.clientY - canvas.offsetTop - centerY);
const step = 1 / Math.max(a, b);
```

```
ctx.clearRect(0, 0, canvas.width, canvas.height);
ctx.beginPath();
```

```
for (let angle = 0; angle < 2 * Math.PI; angle += step) {
  let x, y;
```

```
  switch (curveType) {
    case 'circle':
      x = centerX + a * Math.cos(angle);
      y = centerY + a * Math.sin(angle);
      break;
    case 'ellipse':
      x = centerX + a * Math.cos(angle);
      y = centerY + b * Math.sin(angle);
      break;
    case 'hyperbola':
      x = centerX + a / Math.cos(angle);
      y = centerY + b * Math.tan(angle);
      break;
    case 'parabola':
      x = centerX + a * angle;
      y = centerY + b * Math.pow(angle, 2);
      x2 = centerX - a * angle;
      y2 = centerY + b * Math.pow(angle, 2);
      ctx.fillRect(x2, y2, 1, 1);
      break;
  }
```

```
  ctx.fillRect(x, y, 1, 1);
}
```

```
ctx.closePath();
}
```

```
canvas.addEventListener('mouseup', stopDrawing);
canvas.addEventListener('mouseout', stopDrawing);
function stopDrawing() {
  isDrawing = false;
}
```

```
function clearCanvas() {
  ctx.clearRect(0, 0, canvas.width, canvas.height);
}
```

```

index.html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>graph</title>
<style>
#canvas {
border: 1px solid black;
cursor: crosshair;
}

#toolbar {
margin-bottom: 10px;
}

#toolbar button {
margin-right: 5px;
}
</style>
</head>
<body>
<div id = "toolbar">
<select id="curveType" style="border: 2px solid aquamarine; background-color: aliceblue;
border-radius: 20px;">
<option value="circle">Окружность</option>
<option value="ellipse">Эллипс</option>
<option value="hyperbola">Гипербола</option>
<option value="parabola">Парабола</option>
</select>
<button onclick="clearCanvas()" style="border: 2px solid darkred; background-color:
aliceblue; border-radius: 20px; position: absolute; right: 10px;">Очистить
холст</button>
</div>
<canvas id="canvas" width="1825" height="900" style="border: 3px solid aquamarine;
border-radius: 20px;"></canvas>
<script src="app.js"></script>
</body>
</html>

```

Алгоритм построения окружности:

$$x = \text{centerX} + a * \text{Math.cos}(\text{angle});$$

$$y = \text{centerY} + a * \text{Math.sin}(\text{angle});$$

Алгоритмическая оценка:  $O(R)$ , где  $R$  - радиус окружности.

Алгоритм построения окружности:

$$x = \text{centerX} + a * \text{Math.cos}(\text{angle});$$

$$y = \text{centerY} + b * \text{Math.sin}(\text{angle});$$

Алгоритмическая оценка:  $O(b)$ , где  $b$  - полуось эллипса.

Алгоритм построения окружности:

$x = \text{centerX} + a / \text{Math.cos}(\text{angle});$

$y = \text{centerY} + b * \text{Math.tan}(\text{angle});$

Алгоритмическая оценка:  $O(a)$ , где  $a$  - параметр гиперболы.

Алгоритм построения окружности:

$x = \text{centerX} + a * \text{angle};$

$y = \text{centerY} + b * \text{Math.pow}(\text{angle}, 2);$

$x2 = \text{centerX} - a * \text{angle};$

$y2 = \text{centerY} + b * \text{Math.pow}(\text{angle}, 2);$

Алгоритмическая оценка:  $O(a)$ , где  $a$  - параметр параболы.

Вывод:

В ходе выполнения данной лабораторной работы был разработан элементарный графический редактор, позволяющий генерировать и отображать кривые второго порядка: окружности, эллипсы, гиперболы и параболы.

Графический редактор позволяет выбрать нужную кривую через панель инструментов и отображать ее на дискретной сетке. Это позволяет пользователю наглядно наблюдать ее конечный результат.

В результате работы были получены иллюстрации, демонстрирующие результаты работы графического редактора. Эти иллюстрации могут быть использованы для визуализации результатов и анализа работы алгоритмов.

Выводы из данной лабораторной работы заключаются в том, что генерация кривых второго порядка требует использования специфических алгоритмов, которые учитывают их математические свойства. Элементарный графический редактор является полезным инструментом для визуализации и изучения этих кривых. Он может быть расширен и улучшен для поддержки дополнительных функций и возможностей.