МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«Харківський Політехнічний Інститут»

Кафедра стратегічного управління

ЗВІТ

з лабораторної роботи №3

з дисципліни «Програмування веб-застосунків»

Перевірила

Виконав:

Студент групи КН-27

Харків – 2019

**Тема:** Робота з графічними зображеннями.

**Варіант:** №2 - Реалізувати алгоритм Ву для побудови лінії

**Хід роботи**

1. Код JavaScript алгоритму наведений ничжче

|  |
| --- |
| var canvas = undefined  var ctx = undefined  let scale = 1  window.onload = () => {  let x1 = document.getElementById("x1")  let x2 = document.getElementById("x2")  let y1 = document.getElementById("y1")  let y2 = document.getElementById("y2")  scale = document.getElementById("scale")  canvas = document.querySelector("canvas")  ctx = canvas.getContext("2d")  document.getElementById("clear").addEventListener("click", clear)  x1.addEventListener("input", clearAndDraw)  x2.addEventListener("input", clearAndDraw)  y1.addEventListener("input", clearAndDraw)  y2.addEventListener("input", clearAndDraw)  scale.addEventListener("input", clearAndDraw)  clearAndDraw()  drawRealLine(+x1.value,+y1.value,+x2.value,+y2.value)  }  function drawRealLine(x1,y1,x2,y2){  ctx.strokeStyle = "rgb(255,0,0,1)"  ctx.beginPath();  ctx.moveTo(x1,y1);  ctx.lineTo(x2, y2);  ctx.stroke();  }  function clearAndDraw(){  ctx.clearRect(0, 0, canvas.width, canvas.height);  draw\_line(+x1.value,+y1.value,+x2.value,+y2.value)  drawRealLine(+x1.value,+y1.value,+x2.value,+y2.value)  }  function clear(){  x1.value = ""  x2.value = ""  y1.value = ""  y2.value = ""  ctx.clearRect(0, 0, canvas.width, canvas.height);  }  function plot(x, y, c){  ctx.fillStyle = "rgb(0,0,0," + c  ctx.fillRect(x\*scale.value,y\*scale.value,scale.value,scale.value)  }  function ipart(x){  return Math.floor(x)  }    function round(x){  return Math.floor(x)  }    function fpart(x){  return x - Math.floor(x)  }  function draw\_line(x1,y1,x2,y2){  if (x2 < x1){  x2 = [x1, x1 = x2][0];  y2 = [y1, y1 = y2][0];  }    let dx = x2 - x1  let dy = y2 - y1  let gradient = dy / dx    // обработать начальную точку  let xend = round(x1)  let yend = y1 + gradient \* (xend - x1)  let xgap = 1 - fpart(x1 + 0.5)  let xpxl1 = xend // будет использоваться в основном цикле  let ypxl1 = ipart(yend)  plot(xpxl1, ypxl1, (1 - fpart(yend)) \* xgap)  plot(xpxl1, ypxl1 + 1, fpart(yend) \* xgap)  let intery = yend + gradient // первое y-пересечение для цикла    // обработать конечную точку  xend = round(x2)  yend = y2 + gradient \* (xend - x2)  xgap = fpart(x2 + 0.5)  let xpxl2 = xend // будет использоваться в основном цикле  let ypxl2 = ipart(yend)  plot(xpxl2, ypxl2, (1 - fpart(yend)) \* xgap)  plot(xpxl2, ypxl2 + 1, fpart(yend) \* xgap)    // основной цикл  for (let x = xpxl1 + 1; x < xpxl2 - 1; x++){  plot(x, ipart(intery), 1 - fpart(intery))  plot(x, ipart(intery) + 1, fpart(intery))  intery = intery + gradient  }  } |

2. Розмітка HTML наведена нижче

|  |
| --- |
| <html lang="en">  <head>  <meta charset="utf-8">  <title>The HTML5 Herald</title>  <meta name="description" content="The HTML5 Herald">  <meta name="author" content="SitePoint">  </head>  <body>  <script src="script.js"></script>  X1: <input type = "text" value = "0" id = "x1" title = "X1">  X2: <input type = "text" value = "200" id = "x2" title = "X2">  Y1: <input type = "text" value = "0" id = "y1" title = "Y1">  Y2: <input type = "text" value = "50" id = "y2" title = "Y2">  Scale: <input type = "text" value = "1" id = "scale" title = "Scale">  <button id = "clear">Очистить</button>  <br>  <canvas style = "border : 1px solid red; width: 100%"></canvas>  </body>  </html> |

1. Діаграму діяльності зображено на рисунку 1

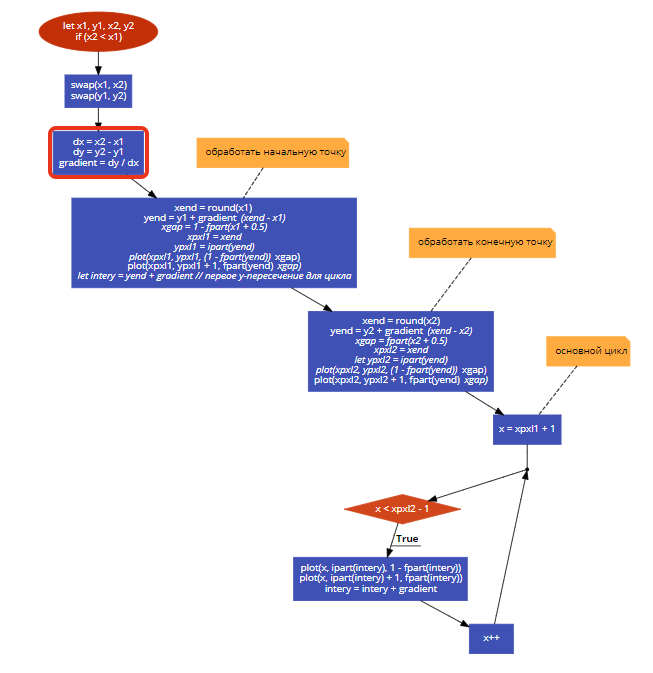
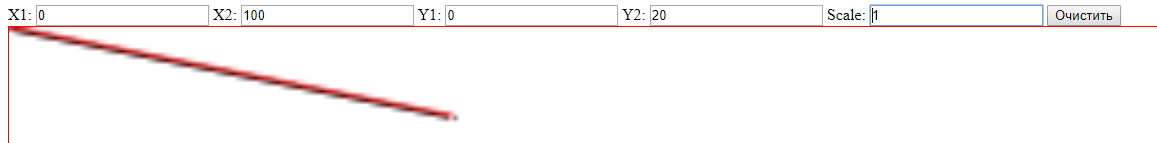


Рисунок 1 – Діаграма діяльності алгоритму Ву

1. Приклад виконання программи зображено на рисунку 2



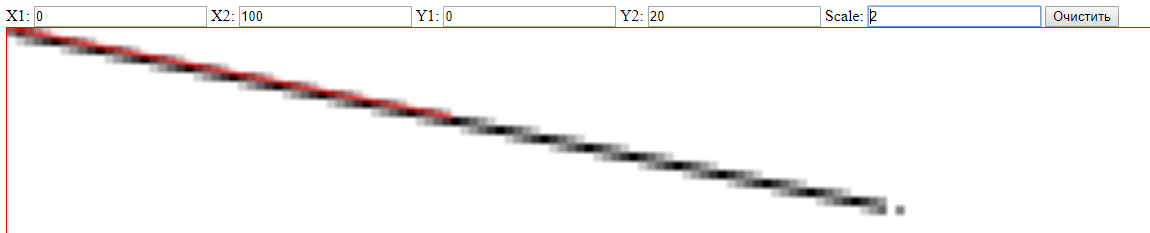


Рисунок 2 – Робота програми

Висновки

При виконанні цієї лабораторної роботи було реалізовано алгоритм побудови лінії Ву у мові JavaScript. Було здійсненно знайомство з елементом canvas розмітки HTML.