

Trabalho de Computação Gráfica

Data de entrega: 02/04/2016

Autora: Giovanna Cazelato Pires RA: 141151171

Sumário

Introdução.....	2
Procedimentos	3
Resultados.....	4
Cálculo do Tempo	7
Bibliografia	8

Introdução

Como solicitado pelo professor de Computação Gráfica do curso de Ciências da Computação da Universidade Estadual Paulista “Junior de Mesquita Filho”, foram realizados dados algoritmos de geração de linha e circunferência em aula para implementar para a linguagem C. E utilizando a biblioteca do OpenGL deveríamos mostrar os pontos na tela.

Procedimentos

Para realizar esse trabalho, utilizei o Code::Blocks(versão 13.12) para implementar em C os algoritmos dados em aula.

Resultados

Segue abaixo imagens de como os códigos funcionaram em sua execução.

Todas as funções que geraram linhas rodaram com os pontos (4,7) e (14,14).

Todas as funções que geraram circunferências rodaram com os pontos de centro (0,0) e raio 7.

- DDA (Digital Differential Analyzer):

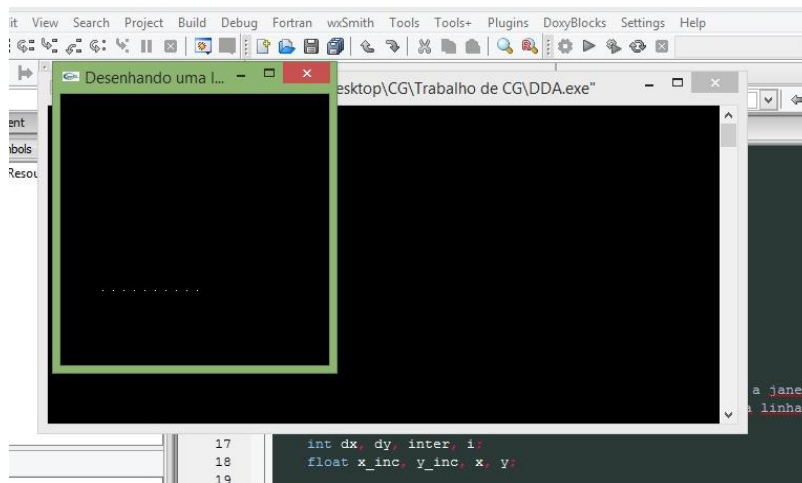


Figura 1 : DDA

-Bresenham para Linhas :

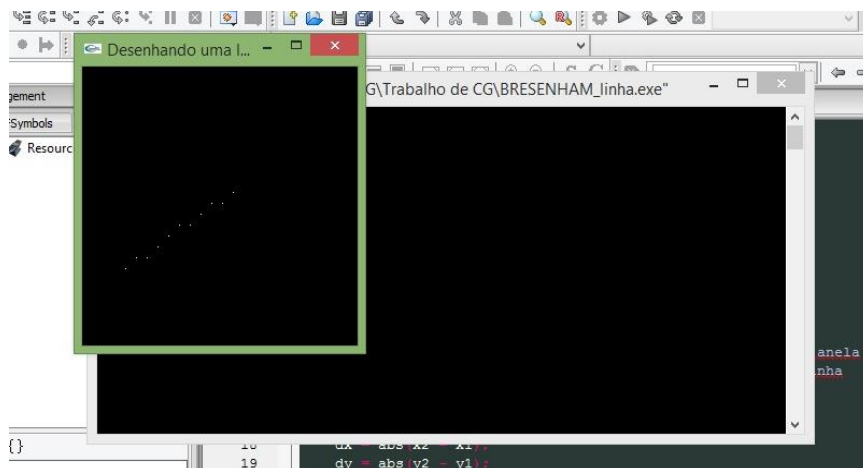


Figura 2 : Bresenham para linhas

- Equação da Circunferência:

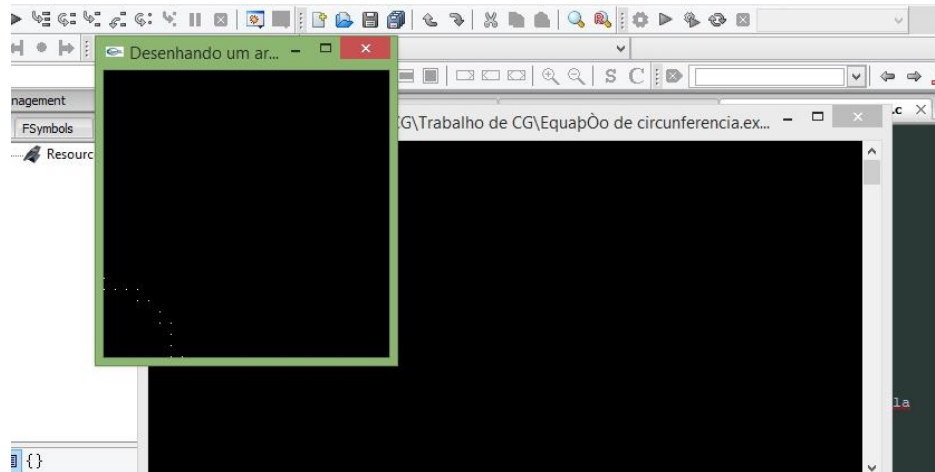


Figura 3: Equação de circunferência

-Bresenham para Circunferências :

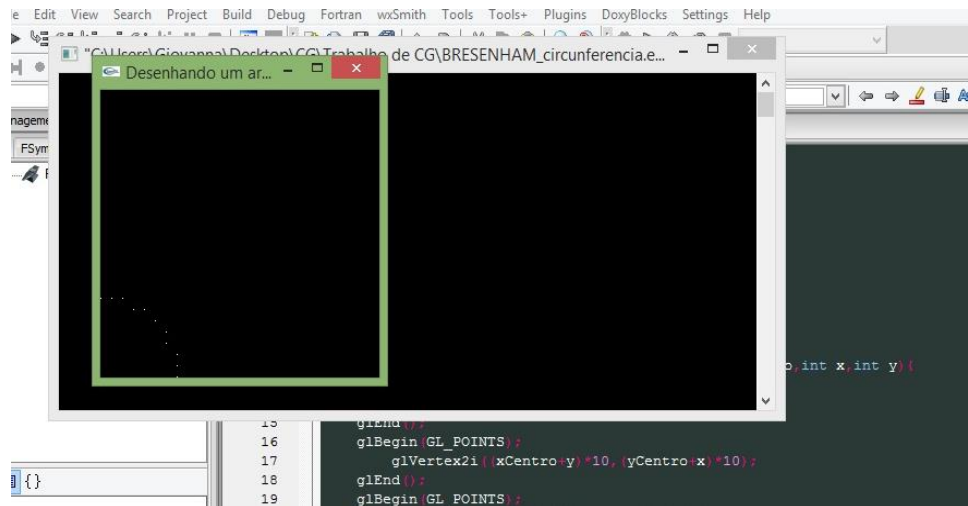


Figura 4 : Bresenham para circunferências

Para ver as Print Screens em seu tamanho original clique nos links abaixo:

- Figura 1: <http://imgur.com/q2qIP5o>
- Figura 2: <http://imgur.com/kQY99Jg>
- Figura 3: <http://imgur.com/KGkO42l>
- Figura 4: <http://imgur.com/p6Fw09v>

Cálculo do Tempo

Para calcular o tempo dos programas utilizei o tempo de execução que o próprio Code::Blocks fornece. Todas as vezes (no mínimo umas 20 a 30 execuções) que executei o programa o seu tempo foi de 2 segundos, em todos os programas, ou seja, se tiver alguma diferença de tempo entre uma função e outra, é na casa dos milissegundos para menos.

Bibliografia

<http://www.inf.pucrs.br/~manssour/OpenGL/Tutorial.html>

<http://www.di.ubi.pt/~agomes/cig/praticas/01-lab-opengl.pdf>

<http://www.dca.ufrn.br/~ambj/opengl/>