

# Geometria Computacional

## Trabalho 1

Luis Peñaranda

2014.2

O trabalho pode ser feito de jeito individual ou em dupla. Deve ser entregue por email a [luisp@dcc.ufrj.br](mailto:luisp@dcc.ufrj.br) em um arquivo compactado, que deve conter uma pasta com os programas correspondentes a cada exercício claramente indicados. Além dos programas, a pasta deve conter o arquivo `CMakeLists.txt` para compilar com CMake.

Os programas devem usar contenedores e iteradores da STL, referências a objetos e entrada/saída de C++. O uso de estruturas de dados de C, ponteiros e funções de entrada/saída de C (tais como `printf` e `scanf`) será penalizado. Além disso, a *const-correctness* dos programas será premiada com um bonus.

## 1 CGAL

### 1.1 Instalação

Não é trivial ter uma versão de CGAL instalada. Se não quiser usar o CGAL instalado no laboratório, o primeiro exercício consiste na instalação de CGAL. Pode-se encontrar ajuda na seção de instalação do manual online, <http://doc.cgal.org/latest/Manual/installation.html>.

### 1.2 Manual

Acessar o manual online, em <http://doc.cgal.org/latest/Manual/index.html>. Para referência, pode-se usar o capítulo para desenvolvedores; em especial, as leituras sugeridas ao final do capítulo.

## 2 Arranjos de segmentos

O exercício usará os arranjos 2D de CGAL. Deve-se consultar então o capítulo do manual correspondente. Lembre-se da existência de dois submanuais diferentes: usuário e referência. Deve-se primeiro ler o manual do usuário, e usar o manual de referência para consultar os protótipos das funções ao programar.

## 2.1 Exemplos do manual

Combinar os exemplos do manual para obter um programa que leia um arquivo com segmentos, cujos extremos estão dados por coordenadas `double`, e construa o arranjo correspondente. A saída do programa serão as tabelas que formam a DCEL, como visto na aula.

## 2.2 Geração de casos de teste

Escrever um programa para gerar segmentos aleatórios no plano. A saída deve ser possível de ler pelo programa da seção 2.1.

## 2.3 Casos degenerados

Escrever um programa para gerar todos os casos degenerados de segmentos no plano vistos na aula (eles são: segmentos com um ou dois extremos em comum, pontos com igual coordenada  $x^1$ , segmentos com extremos iguais, segmentos verticais<sup>2</sup>, vários segmentos que intersectam-se em um mesmo ponto). A saída deve ser possível de ler pelo programa da seção 2.1.

## 2.4 Bonus

Pesquisar no manual como usar as bibliotecas Qt em CGAL. Escrever um programa que imprima um arranjo de segmentos. A entrada será um conjunto de segmentos dentro de um arquivo, como os gerados nos itens anteriores.

---

<sup>1</sup>No algoritmo do livro, a linha de varredura era paralela ao eixo  $x$ ; em CGAL, a linha de varredura é paralela ao eixo  $y$  e segue a direção do eixo  $x$ .

<sup>2</sup>Como a linha de varredura é paralela ao eixo  $y$ , os segmentos verticais são degenerados, e não os horizontais como no algoritmo do livro.