MAC 747 GEOMETRIA COMPUTACIONAL

PROVA 2

1. Instruções

Esta prova é individual e deverá ser entregue até às 17 horas da segunda-feira dia 24 de novembro na secretaria do Departamento de Ciência da Computação (sala 256 do Bloco A).

2. Questões

1. (1.0 ponto—Vértice do fecho convexo) Dado um conjunto S com n pontos de E^2 descreva um algoritmo linear que determina se um ponto $p \in S$ é um vértice do fecho convexo de S. Mostre a corretude e analise complexidade do seu algoritmo.
2. (1.0 ponto—Camadas convexas) Dado um conjunto S de n pontos no plano nós definimos a *camadas convexas* de S indutivamente. A primeira camada convexa de S consiste da fronteira de $\operatorname{conv}(S)$ (ou seja, a linha poligonal (ou cadeia) fechada que tem como vértices os vertices de $\operatorname{conv}(S)$). Para $i > 1$, defina S_i como sendo os pontos de S que restaram após removermos as camadas $i = 1, 2, \ldots, i-1$. Então a i-ésima camada convexa de S é a fronteira de $\operatorname{conv}(S_i)$ se $S_i \neq \emptyset$. (Veja Figura 1.)

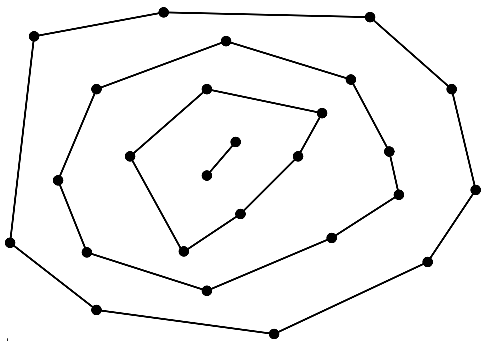

Figura 1. Camadas convexas de um conjunto de pontos.

Descreva um algoritmo de complexidade de tempo $O(n^2)$ que constrói as camadas convexas de um dado conjunto S de n pontos. Mostre a corretude e analise a complexidade do seu algoritmo.

Date: 19 de novembro de 1997.

3. (1.0 ponto—Conjunto de pontos $\Rightarrow$ polígono) Dado um conjunto S de n pontos no plano descreva um algoritmo de complexidade de tempo $O(n \log n)$ que constrói um polígono (simples) que tem como vértices os pontos de S. Mostre a corretude e analise a complexidade do seu algoritmo.
4. (1.5 pontos—Fecho convexo de intersecção de retas) Dado um conjunto H de n retas no plano tal que duas-a-duas as retas são não-paralelas descreva um algoritmo de complexidade de tempo $O(n \log n)$ que constrói o fecho convexo dos pontos de intersecção entre as retas. Mostre a corretude e analise a complexidade do seu algoritmo.
5. (1.0 ponto) Prove que se S é um conjunto de n pontos de E^2 com no máximo k pontos cocirculares então cada vértice do diagrama de Voronoi $\mathrm{Vor}(S)$ de S tem grau entre 3 e k.
6. (1.5 ponto—Grafo de vizinhanca relativa) Uma *grafo de vizinhanca relativa* (*relative neighborhood graph*) de um conjunto S de n pontos de E^2, denotado por $\mathrm{RNG}(S)$, é um grafo cujo conjunto de vértices é S e existe uma aresta ligando dois pontos p e q, $\{p, q\} \subseteq S$, se

$$\text{distância}(p,q) \leq \min_{r \in S \setminus \{p,q\}} \max\{\text{distância}(p,r), \text{distância}(q,r)\}.$$

   (Esta desigualdade determina uma região 'proibida' para o ponto r se p e q são adjacentes no grafo de $\mathrm{RNG}(S)$. Esta região é formada pela intersecção dos círculos de centro p e q e raio distância(p,q).)
   (a) Prove que toda aresta de $\mathrm{RNG}(S)$ é uma aresta de $\mathrm{Del}(S)$.
   (b) (MST $\subseteq$ RNG) Prove que toda aresta de uma árvore geradora mínima de S ($\mathrm{MST}(S)$) é uma aresta de $\mathrm{RNG}(S)$.
   (c) Descreva um algoritmo de complexidade de tempo $O(n^2)$ que constrói o grafo de vizinhanca relativa de um dado conjunto S de n pontos. Mostre a corretude e analise a complexidade do seu algoritmo.
7. (1.0 pontos—Atualização dinâmica do diagrama de Delaunay) Dado o digrama de Delaunay $\mathrm{Del}(S)$ de um conjunto S de n pontos de E^2 e um ponto $p \in S$ descreva um algoritmo que constrói o diagrama de Delaunay de $S \setminus \{p\}$. A complexidade de tempo do seu algoritmo deve ser $O(k \log k)$ onde k é o número de arestas adjacentes ao sítio p. Mostre a corretude e analise a complexidade do seu algoritmo.
8. (2.0 pontos—Construcão do diagrama de Delaunay *on-line*) Seja S um conjunto de n sítios em E^2 e seja p um sítio tal que $p \notin S$ e $p \in \mathrm{conv}(S)$.
   (a) Mostre que se p é um sítio no interior do triângulo $\triangle(a,b,c)$ do diagrama $\mathrm{Del}(S)$ então pa, pb e pc são arestas de $\mathrm{Del}(S \cup \{p\})$.
   (b) Mostre que se p é um sítio da aresta ab do diagrama $\mathrm{Del}(S)$ que é compartilhada pelos triângulos $\triangle(a,b,c)$ e $\triangle(d,b,a)$ de $\mathrm{Del}(S)$ então pa, pb, pc e pd são arestas do digrama $\mathrm{Del}(S \cup \{p\})$.
   (c) Chamaremos de *suspeitas* as arestas de $\mathrm{Del}(S)$ que suspeitamos que não fazem parte de $\mathrm{Del}(S \cup \{p\})$. Após as operações descritas nos itens (a) e (b) quais arestas de $\mathrm{Del}(S)$ são *supeitas*? Mostre

como testar em tempo constante se uma aresta suspeita pertence ou não a Del$(S \cup \{p\})$.

(d) Se constatarmos que uma aresta suspeita ab não pertence a Del$(S \cup \{p\})$ então qual aresta deverá ser inserida no diagrama? Depois de inserirmos esta aresta as quais arestas quer eram `não-suspeitas` passam a ser `suspeitas`?

(e) Quantas vezes uma aresta `suspeita` será testada?

(f) Dados o diagrama de Delaunay de um conjunto S de n sítios de E^2, um sitio $p \notin S$ e o triângulo $\triangle(a, b, c)$ do diagrama de Delaunay de S que contém p, descreva um algoritmo que constrói o diagrama de Delaunay de $S \cup \{p\}$. A complexidade do seu algoritmo deve ser $O(k)$ onde k é o número de arestas inseridas mais o número de arestas removidas. (Se você não conseguir um algoritmo $O(k)$ então tente descrever um algoritmo $O(n)$.) Mostre a corretude e analise a complexiade do seu algoritmo. (Se $p \notin \operatorname{conv}(S)$ então Del$(S \cup \{p\})$ pode ser construído através do Algoritmo `MergeDelaunay`.)

[Note que o algoritmo pedido por este item é o passo central de um algoritmo *on-line* que constrói o diagrama de Delaunay de um conjunto de n sítios em tempo $O(n^2)$.]