SEGUNDA PROVA DE ALGORITMOS EM GRAFOS
BCC, 1o. SEMESTRE DE 2004

1. [2 pontos] Descreva o que ocorre quando executamos o algoritmo de Kosaraju em um DAG.
2. [3 pontos] Seja G um grafo conexo e $c\colon E(G) \to \mathbb{R}$ uma função custo definida nas arestas de G. Por simplicidade, suponha que c é injetora, isto é, não há arestas com custos iguais em G. Seja T uma árvore geradora de G. Para $e \in E(T)$, sejam T_1^e e T_2^e os dois componentes de $T - e$. Prove que são equivalentes:
   (i) T é uma árvore geradora de custo mínimo em G.
   (ii) Toda aresta $e \in E(T)$ tem peso mínimo dentre as arestas do corte $E(V(T_1^e), V(T_2^e)) = \big\{\{u,v\} \in E(G)\colon u \in V(T_1^e),\ v \in V(T_2^e)\big\}$.
   [*Sugestão.* Para provar que (ii) implica (i), faça o seguinte: suponha que T não seja uma árvore mínima e seja T^* uma árvore mínima. Seja e uma aresta em $E(T) \setminus E(T^*)$, e considere $E(V(T_1^e), V(T_2^e))$. Seja f uma aresta em $E(T^*) \cap E(V(T_1^e), V(T_2^e))$ adequada. Considere $T^* + e - f$.]
3. [3 pontos] Seja G um grafo dirigido e $c\colon E(G) \to \mathbb{R}$ um função custo definida nos arcos de G. Por simplicidade, suponha que o custo de todo arco é não-negativo. Fixe $s \in V(G)$, e suponha que todo vértice de G é acessível a partir de s. Para todo $x \in V(G)$, escrevemos $d_c(s,x)$ para o custo mínimo dos caminhos dirigidos de s a x. Um *potencial* p *para* (G,c) é uma função $p\colon V(G) \to \mathbb{R}$ tal que
   $$p(v) \le p(u) + c(u,v) \text{ para todo } (u,v) \in E(G). \tag{1}$$
   Ademais, dizemos que um arco (u,v) é *justo* se vale a *igualdade* em (1). Suponha que um potencial p satisfaz as seguintes duas propriedades:
   (a) $p(s) = 0$,
   (b) todo vértice $x \in V(G)$ é acessível a partir de s por um caminho que só contém arcos justos.
   Prove os seguintes fatos:
   (i) Para todo $x \in V(G)$, vale que $p(x) \le d_c(s,x)$. [*Observação.* Para este item, a única hipótese necessária sobre o potencial p é que $p(s) \le 0$.]
   (ii) Para todo $x \in V(G)$, vale que $d_c(s,x) \le p(x)$. [*Sugestão.* Seja (P_k) a seguinte asserção: *se x é acessível a partir de s através de um caminho com k arcos, todos eles justos, então $d_c(s,x) \le p(x)$.* Prove que (P_k) vale para todo k por indução em k.]
   Conclua que $d_c(s,x) = p(x)$ para todo vértice x de G.

4. [4 pontos] Considere a implementação do algoritmo de Dijkstra visto em sala (Programa 21.1):

```
#define GRAPHpfs GRAPHspt
#define P (wt[v] + t->wt)
void GRAPHpfs(Graph G, int s, int st[], double wt[])
  { int v, w;  link t;
    PQinit(); priority = wt;
    for (v = 0; v < G->V; v++)
      { st[v] = -1; wt[v] = maxWT; PQinsert(v); }
    wt[s] = 0.0; PQdec(s);
    while (!PQempty())
      if (wt[v = PQdelmin()] != maxWT)
        for (t = G->adj[v]; t != NULL; t = t->next)
          if (P < wt[w = t->v])
            { wt[w] = P; PQdec(w); st[w] = v; }
  }
```

Seja G o grafo de 5 vértices e 10 arcos com custos nos arcos dado por

```
0: 1(5) 3(10)
1: 2(2) 3(3) 4(9)
2: 0(7) 4(6)
3: 1(2) 4(1)
4: 2(4)
```

Na notação acima, por exemplo, `0: 1(5)` significa que há um arco de `0` para `1` de custo `5`.

(i) Execute o Programa 21.1 no grafo G dado acima, com $s = 0$, ilustrando a evolução do algoritmo através de diagramas convenientes (uns 6 diagramas bastam).

(ii) Determine um potencial $p\colon V(G) \to \mathbb{R}$ para certificar que você encontrou as distâncias de 0 a todos os outros vértices em G em (i) acima corretamente (veja a Questão 3). Você deve dizer explicitamente por que sua função p é um potencial e por que ela de fato certifica as distâncias encontradas.

(iii) Descreva um algoritmo de complexidade $O(n + m)$ que poderia ser executado após o Programa 21.1 para verificar que de fato as distâncias a partir de s foram determinadas corretamente. Por simplicidade, suponha que todos os vértices de G são acessíveis a partir de s. (Como você pode se livrar dessa hipótese?)