UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ELETRÔNICA E COMPUTAÇÃO
ELC1106 – REDES DE COMUNICAÇÃO DE DADOS
PROFESSOR: CARLOS HENRIQUE BARRIQUELLO

4ª Aula prática: roteiro

Título: Roteamento com o algoritmo de caminho mais curto de Edsger Dijkstra

Objetivo: Responder a seguinte questão: **a) Qual é caminho mais curto entre dois vértices quaisquer com o maior custo no grafo e quanto é o custo?**

Roteiro:

1) O arquivo de simulação "grafo_de_rede.m" está disponível no Moodle. Abra-o no MATLAB e execute-o para gerar um grafo com pontos aleatórios.

2) Implemente o algoritmo de Dikjstra para responder a questão "a".

3) Escreva um relatório respondendo a pergunta "a", com sua implementação do algoritmo, e faça *upload* do relatório da aula no Moodle.

4) Modifique o código da simulação para implementar também o algoritmo de Bellmann-Ford.

Algoritmo de Dijkstra

1º passo: iniciam-se os valores:

para todo v ∈ V[G]
    d[v]← ∞
    π[v] ← -1
d[s] ← 0

V[G] é o conjunto de vértices(v) que formam o Grafo G. d[v] é o vetor de distâncias de s até cada v. Admitindo-se a pior estimativa possível, o caminho infinito. π[v] identifica o vértice de onde se origina uma conexão até v de maneira a formar um caminho mínimo.

2º passo: temos que usar o conjunto Q, cujos vértices ainda não contém o custo do menor caminho d[v] determinado.

Q ← V[G]

3º passo: realizamos uma série de relaxamentos das arestas, de acordo com o código:

```
enquanto Q ≠ ø
    u ← extrair-mín(Q)              //Q ← Q - {u}
    para cada v adjacente a u
        se d[v] > d[u] + w(u, v)    //relaxe (u, v)
          então d[v] ← d[u] + w(u, v)
              π[v] ← u
              Q ← Q ∪ {v}
```

w(u, v) é o peso(weight) da aresta que vai de u a v.

u e v são vértices quaisquer e s é o vértice inicial.
Para *extrair-mín(Q),* pode-se usar uma lista ou vetor de vértices onde se extrai o elemento u com menor valor d[u].
No final do algoritmo teremos o menor caminho entre s e qualquer outro vértice de G.