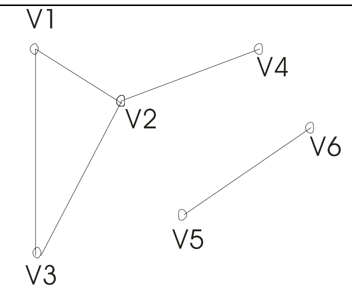
MO640 – Biologia Computacional
Prof. João Meidanis
Ata de Exercícios - 21/03/2005

Exercícios retirados do livro: SETUBAL, J. C.; MEIDANIS, J. *Introduction to Computational Molecular Biology*. PWS Publishing Company, 1997 páginas 43 e 44.

6. Dê a matriz e a lista de adjacência para os grafos da figura 2.1.
R:
Grafo a (não orientado)



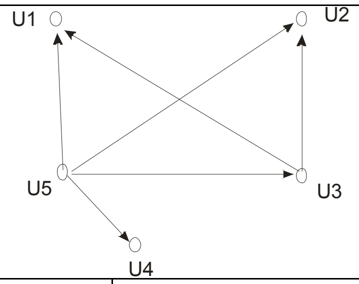
Lista de Adjacência:

```
v1 -> v2 , v3
v2 -> v1 , v3 , v4
v3 -> v1 , v2
v4 -> v2
v5 -> v6
v6 -> v5
```

Matriz de adjacência:

	v1	v2	v3	v4	v5	v6
v1	0	1	1	0	0	0
v2	1	0	1	1	0	0
v3	1	1	0	0	0	0
v4	0	1	0	0	0	0
v5	0	0	0	0	0	1
v6	0	0	0	0	1	0

Grafo b (orientado)



Lista de Adjacência:

```
u1 nil
u2 nil
u3 ->u1 , u2
u4 nil
u5 -> u1 , u2 , u3 , u4
```

Matriz de adjacência:

	u1	u2	u3	u4	u5
u1	0	0	0	0	0
u2	0	0	0	0	0
u3	1	1	0	0	0
u4	0	0	0	0	0
u5	1	1	1	1	0

8. Sejam dois algoritmos A_1 e A_2 para um mesmo problema cujo parâmetro n define o tamanho. Se A_1 roda em tempo O(n/log(n)) e A_2 roda em O(sqrt(n)) qual é o mais rápido assintoticamente?

R.:

A divisão da complexidade de A_1 pela de A_2, tende a infinito quando n tende a infinito. Assim, A_2 é mais rápido.

Prova:

$$\lim_{n\to\infty}\frac{n/\log n}{\sqrt{n}} = \lim_{n\to\infty}\frac{n}{\log n\sqrt{n}} = \lim_{n\to\infty}\frac{\sqrt{n}}{\log n} =$$

$$\lim_{n\to\infty}\frac{\frac{1}{2}n^{-\frac{1}{2}}}{\frac{1}{n}} = \lim_{n\to\infty}\frac{\frac{1}{2}\frac{1}{\sqrt{n}}}{\frac{1}{n}} = \lim_{n\to\infty}\frac{\frac{1}{2}}{\frac{1}{\sqrt{n}}} = \lim_{n\to\infty}\frac{\sqrt{n}}{2} = \infty$$

9. Leonhard Euler mostrou que um grafo é Euleriano se e somente se ele é conexo e todos os vértices têm grau par. Baseado nesta observação, proponha um algoritmo que encontre um ciclo Euleriano em um grafo caso ele exista.

R:

```
Função Ciclo_Euleriano;
      Entrada: G(V,E)

      Testa conectividade de G usando DFS (Depth-First Search) ou BFS
(Breadth-First Search)
      Testa se todos os nós de G possuem grau par.
      v<- qualquer vértice de V
      Ciclo <- Euler (V, E, v)
      Retorna Ciclo

      Função Euler (V′, E′, v′)
            P <- Encontra_Ciclo (V′, E′, v′) //Encontra um ciclo
                                                iniciando em v′

            Apaga de E′ as arestas em P.
            Seja H o sub-grafo restante que pode não ser mais conexo:
                  - todos os vértices de H terão grau par
                  - H será composto de H1 (V1,E1), .., Hn (Vn, En)sub-
                    componentes conexos.
            Seja vi um vértice em P e em Vi
            Para i = 1 até n
               C<- Euler (Vi, Ei, vi)
               P<- Junção (C,P, vi) // Junção de C e P em vi
            Fim Para
            C<- P
      Retorna C

Fim Função
```

14. Suponha um problema X de complexidade desconhecida. Você tem sucesso em reduzi-lo a um problema NP-Completo conhecido. O que se pode dizer sobre a complexidade de X?

R: Se a redução for polinomial, pode-se dizer que X pertence à NP.

Anotador(a): Renato Hirata