

3. [7 valores]

Considere um grafo representativo de uma povoação de formigas em que cada vértice representa uma colónia de formigas, identificada por uma designação, e cada ramo a ligação entre duas colónias, ao qual está associado a concentração de feromona - proporcional ao número de formigas que se deslocaram nesse ramo.

Suponha que esta povoação de formigas é representada por um grafo não dirigido da classe `template ListAdjGrafo<TV,TR>`, com o qual se pretende representar o movimento das formigas entre as várias colónias. Assuma que num percurso efectuado pelas formigas estas deslocam-se sem repetir colónias já visitadas, sendo atraídas pelas ligações com maior concentração de feromona.

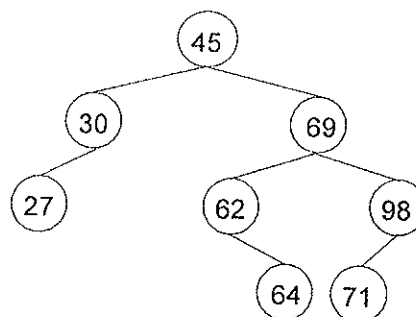
- a) Esboce a declaração da classe (não *template*) `Formigueiro` e respectivas classes `TV` e `TR` indicando apenas os atributos das classes (não é necessário indicar os métodos das classes).

Adicione à classe `Formigueiro` os métodos para responder às alíneas a seguir indicadas:

- b) Dada uma qualquer colónia inicial verificar se é possível percorrer todas as colónias do formigueiro, exactamente uma só vez, e retornar à colónia inicial, sem repetir colónias já visitadas.
- c) Dada uma colónia devolver o ramo que parte desta com maior concentração de feromona.
- d) Dada uma colónia inicial apresentar o caminho que inclui as n ligações com maior concentração de feromona. Note que o número de ligações tem de ser inferior ao número de colónias.

4. [2 valores]

Considere a seguinte árvore binária de pesquisa.



Supondo que é usado o mecanismo AVL de balanceamento deste tipo de árvores, mostre a evolução da árvore de acordo com a seguinte sequência de operações:

- a. remoção do 45 b. inserção do 85

Deve mostrar apenas as duas configurações finais, uma para cada alínea, indicando se necessário, as rotações aplicadas e respectiva justificação.

1. [7 valores]

Defina uma nova classe `LstFIFO<T>`, derivada da classe template `Lista<T>`, de tal modo que a lista funcione segundo a filosofia FIFO.

- a) Apresente a definição da classe `LstFIFO<T>`, e declare apenas os métodos necessários ao bom funcionamento da classe.
- b) Implemente o método para inserir elementos numa `LstFIFO`.
- c) Implemente o método para remover elementos de uma `LstFIFO`.
- d) Faça a sobrecarga do operador `%` para remover os **n últimos** elementos adicionados a uma `LstFIFO`.

2. [4 valores]

```
template <class T>
void metodo (T v[], int n)
{
    for ( ; n > 1; n--)
    {
        int pos_max = 0;
        for (int i = 1; i < n; i++)
            if (v[i] > v[pos_max])
                pos_max = i;

        T val_max = v[pos_max];
        v[pos_max] = v[n-1];
        v[n-1] = val_max;
    }
}
```



- a) Explique o que faz o código do método acima apresentado.
- b) Determine a complexidade temporal $T(n)$ do método em notação Big-Oh. Justifique.
- c) Proponha uma versão recursiva deste método.