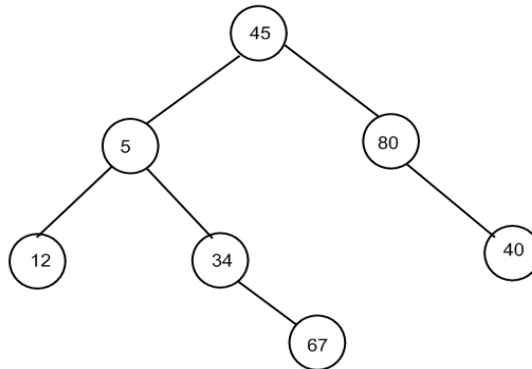
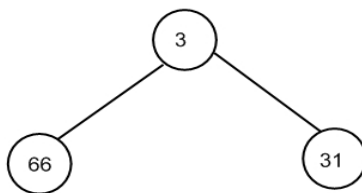


**DEVTITANS**  
**Básico de Estruturas de Dados e Algoritmos**

- 1) Baixe o arquivo **listas.c** e implemente a função "igual", que compara duas listas encadeadas e retorna 1 se forem iguais e 0 caso contrário.
- 2) Baixe o código **arvbin.c**. Implemente três funções para imprimir todos elementos da árvore usando os métodos pré-fixado, in-fixado e pós-fixado. Implemente uma função main que, usando as operações inicializa e cria, gere a seguinte árvore e depois imprima usando os três métodos.



- 3) Usando as funções cria e inicializa, enxertar a seguinte sub-árvore como filho esquerdo do nó 80. Imprimir a árvore resultante pelos três métodos (pré-fixado, in-fixado e pós-fixado). Utilize a função de busca para buscar o nó com valor "80" e, depois disso, fazer o enxerto.



- 4) Podar a sub-árvore que possui raiz com a informação "5" (inclusive). Utilize a função de busca para buscar o nó com valor "5" e, depois disso, fazer a poda. Imprimir a árvore resultante usando os três métodos (pré-fixado, in-fixado e pós-fixado).
- 5) Baixe o código **arvbinbusca.c**. Implementar uma função que imprima a árvore in-fixada. Implementar uma função main que insira os seguintes elementos 1, 2, 3, 5, 15 e 7. Após cada inserção imprima a árvore. Buscar os elementos 2 e 8, imprimindo na tela se cada elemento foi encontrado ou não.

**OBSERVAÇÕES:**

- Entregue um código C para cada problema acima;
- Compacte todos os arquivos C utilizando o compactador ZIP.