

## Tarefa 1:

1. Construa uma classe *CConjuntoInt* que guarda um conjunto de números inteiros (estes não podem repetir). Implemente o construtor de cópia e as funções seguintes:

- *void Insert(int n)*; – para inserir um elemento novo no conjunto. Caso este elemento já exista, a função não faz nada; Inicialmente não se sabe quantos elementos vamos inserir.
- *bool Contains(int n)*; – para indicar se um dado elemento está no conjunto;
- *void Remove(int n)*; - para remover um elemento do conjunto. Caso este elemento não se encontre no conjunto, a função não faz nada;
- *void Empty()*; - para apagar todos os elementos do conjunto;
- *void Display()*; - para visualizar no ecrã os elementos do conjunto

2. Adiciona à classe *CConjuntoInt* as funções seguintes:

- *void Unir(const CConjuntoInt& add)*; - para efectuar a união com os elementos do conjunto representado pelo objecto *add*. O conjunto resultante não deve conter elementos repetidos.
- *void Substrair(const CConjuntoInt& dif)*; - para subtrair os elementos do conjunto representado pelo objecto *dif*.
- *void Interseccao(const CConjuntoInt& inter)*; - para efectuar a operação de intersecção com os elementos do conjunto representado pelo objecto *inter*. O conjunto resultante não deve conter elementos repetidos.

3. Teste a classe desenvolvida com a função *main* seguinte:

```
int main(int argc, char* argv[])
{
    CConjuntoInt c1; c1.Insert(4); c1.Insert(7); c1.Insert(6); c1.Insert(5);

    CConjuntoInt c2 = c1;
    c2.Remove(3); c2.Insert(2); c2.Insert(7); c2.Remove(5); c2.Remove(6);
    c1.Display();    c2.Display();

    cout << "Numero de elementos em c1: " << c1.Size() << endl;
    cout << "Numero de elementos em c2: " << c2.Size() << endl;
    cout << "c1 contem 6 ? : " << c1.Contains(6) << endl;
    cout << "c2 contem 6 ? : " << c2.Contains(6) << endl;
    cout << "Uniao:" << endl;
    c1.Display(); c2.Display(); c1.Unir(c2);    c1.Display();
    c2.Insert(3);
    cout << "Interseccao:" << endl;
    c1.Display(); c2.Display(); c1.Interseccao(c2); c1.Display();
    cout << "Diferenca:" << endl;
    c1.Display(); c2.Display(); c2.Substrair(c1); c2.Display();
    c1.Empty();    c1.Display();

    return 0;
}
```