

Enquadramento
Aulas Teóricas
Aulas Práticas
Apontamentos
Útil
Teste Prático



[ED247] TAD Conjunto (ArrayListIntSet)

Neste problema deverá apenas submeter uma classe **ArrayListIntSet** (e não um programa completo).

O problema

Para este problema deve usar como base a classe [ArrayListIntSet](#).

A sua tarefa é criar uma classe **ArrayListIntSet**, que representa um conjunto de números inteiros, implementando o seguinte interface:

```
// Interface que define o TAD conjunto de numeros inteiros
public interface IntSet {
    public boolean contains(int x); // Retorna true se x esta no conjunto
    public boolean add(int x); // Adiciona x ao conjunto (devolve true se conseguir)
    public boolean remove(int x); // Remove x do conjunto (devolve true se conseguir)
    public int size(); // Retorna o numero de elementos do conjunto
    public void clear(); // Limpa o conjunto (torna-o vazio)

    // Metodos a adicionar (nao existentes na classe base)
    public boolean equals(IntSet s); // Retorna true se ambos os conjuntos sao iguais
    public IntSet intersection(IntSet s); // devolve um novo conjunto: a intersecao de ambos
}
```



Note que pode manter todo o código existente e apenas precisa de implementar dois novos métodos (equals e intersection) que deverá acrescentar à classe. Um exemplo de utilização seria:

```
public class TestED247 {
    public static void main(String[] args) {
        IntSet s1 = new ArrayListIntSet(100);
        IntSet s2 = new ArrayListIntSet(100);

        for (int i=1; i<=5; i++)
            s1.add(i);
        System.out.println("s1 = " + s1 + " | tamanho = " + s1.size());

        for (int i=3; i<=7; i++)
            s2.add(i);
        System.out.println("s2 = " + s2 + " | tamanho = " + s2.size());

        IntSet s3 = s1.intersection(s2);
        System.out.println("s3 = " + s3 + " | tamanho = " + s3.size());

        IntSet s4 = s2.intersection(s1);
        System.out.println("s4 = " + s4 + " | tamanho = " + s4.size());

        System.out.println("s1.equals(s2) = " + s1.equals(s2));
        System.out.println("s3.equals(s4) = " + s3.equals(s4));
    }
}
```

Quando executado este código deverá dar o seguinte output:

```
s1 = {1,2,3,4,5} | tamanho = 5
s2 = {3,4,5,6,7} | tamanho = 5
s3 = {3,4,5} | tamanho = 3
s4 = {3,4,5} | tamanho = 3
s1.equals(s2) = false
s3.equals(s4) = true
```

Input e Output

Deverá apenas submeter a classe **ArrayListIntSet**. O Mooshak irá criar várias instâncias da sua classe usando o construtor (como mostrado no exemplo de utilização) e irá fazer uma série de testes aos métodos por si implementados.

É garantido que o conjunto nunca terá mais do que 100 números diferentes, e que todos os números serão inteiros positivos entre 1 e 1000 (inclusive).

É garantido que os métodos são chamados de forma correcta (os argumentos fazem sentido e não geram excepções).