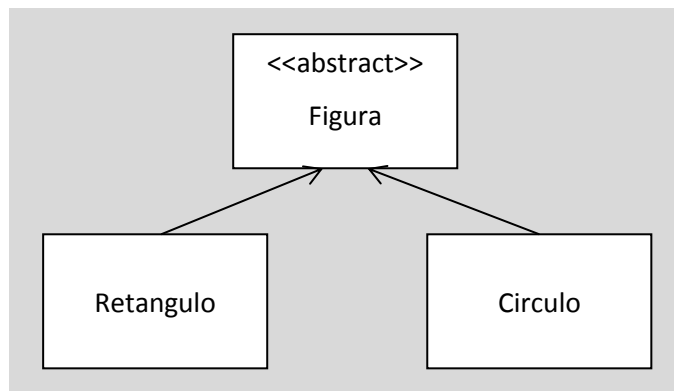


Objetivos Específicos: Utilização das interfaces Java nativas: *Comparable* e *Comparator*. Ordenação de instâncias armazenadas em contentores do tipo *ArrayList*. Método *equals*. Repetição *foreach*. Classes anónimas.

Exercício

Considere o **projeto Netbeans** do exercício **Figuras**, desenvolvido em aulas anteriores, que implementa a seguinte hierarquia de classes de **figuras** geométricas:



1. Crie um **contentor** do tipo ***ArrayList*** para guardar apenas instâncias da hierarquia ilustrada. Este contentor deve designar-se ***figuras2***.
2. **Armazene** no novo contentor as instâncias **Retangulo** e **Circulo** existentes no programa.
3. Programe uma **listagem** das instâncias armazenadas no contentor ***figuras2*** usando uma:
 - a) Repetição *for*
 - b) Repetição *foreach*
4. Usando a **interface** Java **nativa *Comparable*** programe **listagens** das instâncias armazenadas no contentor ***figuras2***:
 - a) Por **ordem crescente** das **áreas** das figuras.
 - b) Por **ordem decrescente** das **áreas** das figuras.
5. **Repita** o ponto anterior, mas usando agora a **interface** Java **nativa *Comparator***.
6. Remova do contentor ***figuras2*** uma instância, usando o método ***remove(Object o)*** da classe ***ArrayList*** e, passando por parâmetro, uma outra instância com as mesmas características da instância a remover. **Verifique** o resultado.
7. Nas classes da hierarquia reescreva o **método *equals***, herdado da classe ***Object***.
8. **Verifique** que o resultado do ponto 6 alterou-se.
9. **Teste** o novo código, comparando:
 - a) Duas instâncias com características iguais.
 - b) Duas instâncias do mesmo tipo com características diferentes.
 - c) Duas instâncias de tipos diferentes.
 - d) A mesma instância.