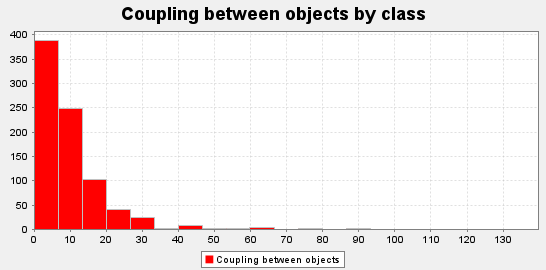
Metricas de Chidamber-Kemerer

FCT UNL

Daniel Gavinho - 59889

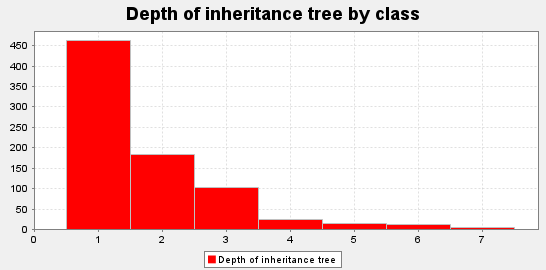
[Ano]

# Coupling Between Objects



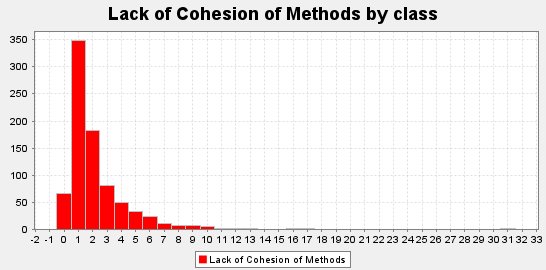
Neste gráfico conseguimos observar o coupling entre objetos. Daqui nós conseguimos tirar a dedução que existem algumas classes seletas que estão coupled (menção e utilização por outra classe, ou utilização de outra classe nela própria) com uma enorme quantidade de outras classes chegando ao máximo de ter entre 60-66 classes coupled em pelo menos uma só classe. Isto pode apresentar problemas de sobre dependência, que pode causar code smells do tipo de Couplers no futuro.

# Depth of inheritance tree



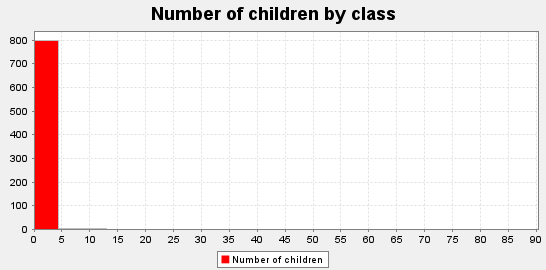
Neste gráfico conseguimos observar a depth of inheritance tree em função às classes. Daqui conseguimos tirar a dedução que maior parte das classes tem só um step antes do java.lang.object, por enquanto, há várias classes que tem entre 4 a 7 steps antes disto.

# Lack of Cohesion of Methods



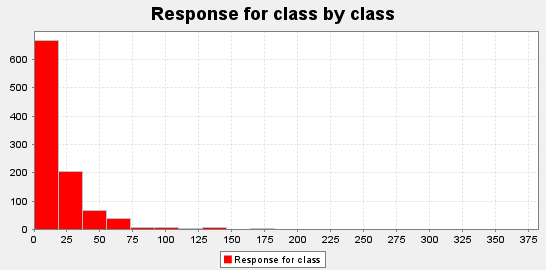
Neste gráfico conseguimos observar a Lack of Cohesion of Methods em função às classes. Daqui nós podemos tirar a dedução que maior parte das classes são de complexidade 1 que indica dificilmente se conseguiria separar esta classe em duas, ou seja, é coesa, por enquanto temos várias classes que tem valores superiores a 1 até um extremo de 10, isto significa que temos classes que estão a realizar demasiado, e eventualmente poderiam separar-se em várias.

# Number of Children



Neste gráfico conseguimos observar o número de crianças em função às classes. Não há muito de especial que se possa observar por esta classe pois no máximo temos +/- 4.5 crianças numa só classe.

# Response for Class



Neste gráfico conseguimos observar a response for class em função às classes. Daqui conseguimos deduzir que maior parte das classes chamam no máximo 0 a 24 métodos quando é chamado um especifico método ou construtor de si mesmo, por enquanto, existem algumas classes que no máximo chamam entre 75 a 145 métodos quando um método específico é chamado, isto pode levar a uma alta instabilidade dessas classes entre outros problemas de implementação.

# Weighted Method Complexity

# 

Neste gráfico conseguimos observar a weighted method complexity em função às classes. A partir disto conseguimos chegar á conclusão que maior parte das classes não tem ciclos muito complexos, por enquanto, seletas classes tem métodos que chegam a um alto valor de complexidade nos ciclos, isto pode levar o programa correr mais lentamente.