# Code Metrics (com recurso ao *plugin* MetricsReloaded para o IntelliJ IDEA)

## Chidamber-Kemerer

### Explicação das métricas recolhidas

O conjunto de métricas oferecido pelo *plugin* que foi escolhido denomina-se de “*Chidamber-Kemerer metrics*”. Este conjunto foi desenvolvido seguindo a métrica que se denomina pelo mesmo nome – *Source Lines of Code* (SLOC), ou *Lines of Code* (LOC).

Ora, este conjunto de métricas é usado, por norma, para medir algumas características de sistemas orientados a objetos, como classes, passagem de mensagens, herança e encapsulamento. Por outro lado, a falta de manutenção de software exige que sejam feitas no sistema existente.

Assim, para classes, este conjunto mede as seguintes métricas:

* Acoplamento entre objetos (CBO)
* Profundidade da árvore de herança (DIT)
* Falta de coesão de métodos (LCOM)
* Número de filhos (NOC)
* Resposta para a classe (RFC)
* Complexidade do método ponderado (WMC)

### Potenciais locais “problemáticos”

Em média, temos um valor de 21.00 para a medida de resposta para a classe. Um valor elevado de *RFC* indica um maior número de falhas, cujas classes são mais complexas e mais difíceis de entender. A testagem e *debugging* será mais complicada:

### A screenshot of a computer Description automatically generated with medium confidence

### Relação mantida com os *Code Smells* identificados

Os problemas apresentados no ponto anterior, em conjunto com análise dos restantes resuldos propostos pelo decorrer da métrica, conferem alguns dos pontos que foram apresentados pelo nosso grupo na fase anterior como *code smells*, tais como:

* “Classe que devia ser um enumerado”
* “Classe com múltiplos métodos, definida dentro de uma interface”
* “Classe exaustivamente extensa”
* “Método demasiado grande e complexo”
* “Utilização de múltiplos ifs em vez que um único switch”
* “Duplicação de código”
* “Valores que deviam ser constantes”
* “Constantes não utilizadas”