

Lab 8 - Acoplamento Lógico

DIG-DCC072 - ENGENHARIA DE SOFTWARE II

Aluno: Danilo Pimentel de Carvalho Costa

Matrícula: 2016058077

1) Implemente, na linguagem de sua escolha, uma solução para detectar acoplamento lógico de um determinado sistema. Detalhe a sua solução.

A solução implementada constrói um grafo, onde vértices são os arquivos e arestas são criadas entre dois arquivos que foram alterados juntos em um mesmo commit. As arestas não são direcionadas, e seu valor representa quantas vezes os dois arquivos foram alterados em um mesmo commit.

O programa em Python lê o arquivo com o resultado do comando acima, construindo uma matriz de adjacências que guarda o valor da aresta descrita acima. A matriz foi representada com um hash no programa, e as chaves do hash são tuplas com os nomes dos dois arquivos.

Depois de construída a matriz, o programa itera sobre as chaves e seus respectivos valores, procurando pelo valor máximo.

2) Selecione um sistema de sua escolha e, com base na sua implementação, responda as seguintes perguntas:

O projeto escolhido para análise foi <https://github.com/facebook/react>. O comando escolhido para gerar o log utilizado pela solução foi:
git log --name-only --pretty=format:"###NEW_COMMIT###".

a) Quais arquivos do sistema mais mudam em conjunto?

"package.json" e "yarn.lock" foram os arquivos mais alterados em conjunto, com 352 ocorrências.

b) Qual o arquivo do sistema que possui mais relacionamentos?

"package.json", com 13094 relacionamentos.

c) Qual o arquivo mais importante do sistema? Proponha uma heurística para mensurar importância.

A heurística implementada calcula uma pontuação para cada arquivo. Os pontos são calculados multiplicando a quantidade de vezes que o arquivo foi alterado pela quantidade de relacionamentos deste arquivo. Desta forma, a pontuação valoriza tanto arquivos que foram muito alterados, quanto arquivos que estão muito acoplados logicamente.

O arquivo package.json, dada esta heurística, é o mais importante, com 6625564 pontos.