**Análise e Síntese de Algoritmos**

**1º Projeto**

**Relatório**

**Grupo 85**

**João Paquete nº89477**

**Introdução**

Neste relatório abordo a minha solução para o problema proposto no 1º projeto de ASA. O objetivo era o de encontrarmos, numa rede de routers, as diferentes sub-redes, os seus identificadores, e os routers que se removidos criariam mais sub-redes.

O input dizia-nos o número de routers da rede, o número de ligações existentes e quais eram essas ligações. Daí tínhamos que extrair o número de sub-redes encontradas, qual o router com o maior identificador de cada sub-rede (o id da sub-rede), qual o número de pontos de articulação encontrados (routers que se retirados, aumentariam o número de sub-redes), e por fim, se todos estes pontos fossem removidos, qual seria o tamanho da maior sub-rede resultante.

**Descrição da solução**

A rede dada, tratada como um grafo não dirigido, foi guardada utilizando uma lista de adjacências, por ocupar menos memória que uma matriz de adjacências quando o número de vértices do grafo é muito elevado.

Para encontrar quer as sub-redes, quer os pontos de articulação, foi usado uma versão alterada do algoritmo de Tarjan. Este algoritmo percorre todos os vértices e guarda informação sobre o seu tempo de descoberta, o seu *low* (o menor valor de descoberta dos seus descendentes) e o número de filhos de cada vértice.

O algoritmo começa pelo vértice com o maior id, verificando assim que o vértice *root* é sempre o id da sub-rede. Marca o tempo de descoberta e o *low* do *root*, que têm o mesmo valor. De seguida, faz uma nova procura na primeira adjacência do vértice atual, ficando guardado que ele é o vértice pai deste novo vértice a ser analisado. Esta recursão é feita até um vértice não ter descendentes, ou ter descendentes que já foram visitados.

Depois verifica se o vértice atual é ou não um ponto de articulação, verificando se:

* O seu tempo de descoberta <= low do vértice descendente;
* O vértice atual é o vértice *root*, e tem 2 ou mais filhos.

Depois vai retrocedendo os passos, verificando se todos os vértices por onde passou são pontos de articulação.

Depois de ter feito este algoritmo para todas as sub-redes, e guardado todos os pontos de articulação, estes pontos são removidos do grafo, e o algoritmo é corrido outra vez, desta vez guardando também o tamanho de cada sub-rede, de maneira a encontrar a maior sub-rede resultante, pedida como output.