

**Instituto Superior Técnico**  
**LEIC-A**  
**Análise e síntese de algoritmos**  
**2º semestre 2013/2014**  
**Projecto 1**  
**Relatório**

Grupo nº183

João Miguel Neves, nº 70171

Rogério Marcial Pires, nº 49746

## **Introdução**

O problema descrito no enunciado do projecto apresenta pessoas e partilhas entre elas. Esta relação pode ser facilmente modelada num computador recorrendo a um grafo dirigido, em que cada pessoa é representada por um vértice do grafo, e cada partilha por um arco.

O problema de identificar grupos de pessoas nos quais circulam os mesmos conteúdos devido a uma sequência de partilhas é análogo ao problema de identificação de componentes fortemente ligados num grafo.

Define-se um componente fortemente ligado como um conjunto máximo de vértices que respeitam a condição de: se um vértice  $v$  é atingível (por algum caminho) partindo de um vértice  $u$ , esse vértice  $u$  é também atingível (por algum caminho) partindo do vértice  $v$ . Vértices isolados (sem arcos de entrada ou de saída) representam também componentes fortemente ligados.

## **Descrição da solução**

Na nossa solução decidimos optar uma lista de adjacências para representar o grafo de partilhas entre pessoas.

A identificação de componentes fortemente ligados é feita recorrendo a uma implementação do algoritmo de Tarjan.

Para a solução pedida é necessário apresentar o numero de componentes fortemente ligados, para isso basta incrementar um contador sempre que o algoritmo de Tarjan identifica um componente fortemente ligado e apresentar este valor no fim da execução do programa.

Para contar o numero de pessoas pertencentes ao maior grupo de partilha basta

inicializar um contador a 0 sempre que o algoritmo de Tarjan identifica um componente fortemente ligado e incrementar esse contador em uma unidade por cada vértice que se retira da pilha. No fim basta comparar o valor deste contador com uma variável que contem o numero de elementos do maior grupo ate ao momento, e caso o grupo actual for maior atribuir-lhe o valor do contador.

Para contar o numero de grupos de pessoas que apenas partilham informação dentro do mesmo grupo basta ser capaz de identificar os componentes fortemente ligados que estejam isolados. Para isso basta que sempre que o algoritmo de Tarjan identifica um componente fortemente ligado verificar se algum dos seus vértices tem pelo menos um arco para um vértice que não pertença ao mesmo componente fortemente ligado.

## Análise teórica

Para a representação do grafo, ponderámos duas hipóteses: a da utilização de lista de adjacências ou de uma matriz de adjacências. Optámos pela primeira, mais indicada para grafos esparsos e requerendo apenas  $O(V+E)$  de utilização de memória. Além disso a utilização de uma matriz de adjacências implicaria uma complexidade superior na execução do algoritmo de identificação de componentes fortemente ligados no caso do grafo a ser analisado ser esparso.

Relativamente a escolha do algoritmo para identificar componentes fortemente ligados num grafo, tivemos três algoritmos em consideração. O algoritmo de Kosaraju, o algoritmo de Tarjan e o algoritmo Path-based.

Apesar de todos os três algoritmos terem complexidade linear ( $O(V+E)$ ), o algoritmo de Kosaraju tem a desvantagem de ter de percorrer duas vezes o grafo, enquanto que os outros dois apenas precisam de percorrer o grafo uma vez.

Os algoritmos de Tarjan e Path-based são bastante semelhantes sendo que a principal diferença entre eles esta no facto de o algoritmo de Tarjan utilizar um vector de dimensão  $V$  para manter os valores de “low”, enquanto que o algoritmo Path-based necessita de uma pilha adicional.

A nossa escolha foi por isso a implementação do algoritmo de Tarjan.

## Avaliação experimental

Execução do programa tendo como input a informação contida no ficheiro de teste “t02.in” disponibilizado na página da cadeira.

INPUT:

6 9

1 5

2 1

2 4  
4 3  
3 2  
5 6  
6 5  
6 3  
2 3

OUTPUT:

1  
6  
1

O programa começa por ler a primeira linha do input extraindo a informação do numero de pessoas (6) e de partilhas (9). Com essa informação reserva espaço na memoria para conter um grafo com a informação que segue no input. De seguida o programa lê cada linha do input adicionando à lista de adjacências o arco correspondente. (De notar que a numeração dos vértices no input encontra-se no intervalo de 1 até V, enquanto que internamente a numeração utilizada encontra-se no intervalo de 0 até V-1). Uma vez adicionados todos os arcos o programa inicia a execução do algoritmo de Tarjan. Após o passo de inicialização do algoritmo de Tarjan e programa entra no ciclo principal, visitando cada vértice do grafo que ainda não tenha sido visitado. Quando o algoritmo de Tarjan identifica um componente fortemente ligado, (neste caso todos os vértices pertencem ao mesmo componente), vai retirando os vértices da pilha principal, incrementando um contador que irá conter o numero de elementos neste componente, e colocando os vértices numa outra pilha temporária criada para manter todos os vértices pertencentes ao mesmo componente fortemente ligado. Após ter terminado este passo incrementa-se o contador com o numero total de componentes fortemente ligados (passa a ter o valor 1) e compara-se o valor do contador de elementos no componente com o maior encontrado anteriormente ( $6 > 0$ ). De seguida o programa vai tentar verificar se este componente esta isolado de outros componentes, verificando para isso cada um dos arcos de todos os vertices contidos na pilha auxiliar até encontrar um arco para outro componente fortemente ligado. Neste caso, como só existe um componente o algoritmo chega ao fim e conclui que este componente se encontra isolado incrementando o contador de componentes isolados para 1. Como todos os vértices já foram visitados o algoritmo de Tarjan termina, e o programa escreve no output o valor do contador de componentes fortemente ligados (1), o valor da variável com o tamanho do maior componente (6) e o valor do contador de componentes isolados (1).