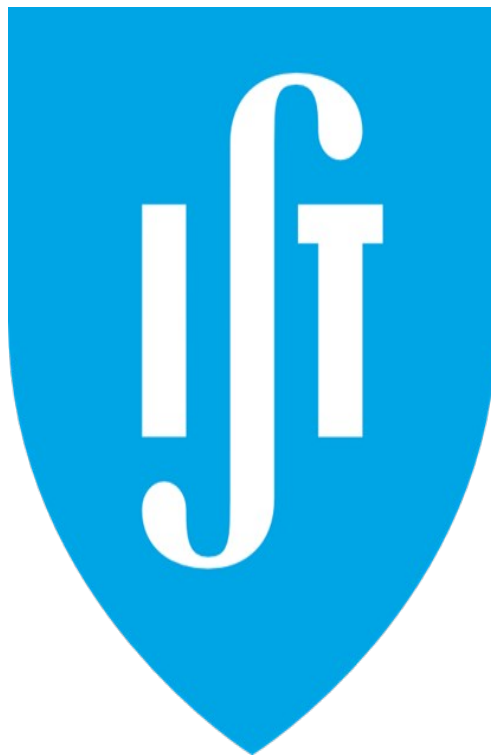


Relatório de Análise e Síntese de Algoritmos

Projecto Parte 1



GRUPO 134

André Gonçalves

65956

Miguel Silva

68230

Introdução

O presente projecto tem como objectivo o desenvolvimento da solução para um problema de partilhas envolvendo pessoas dentro e fora de um determinado grupo. O Relatório descritivo encontra-se dividido em 3 partes contendo a descrição da solução utilizada , a análise teorica do problema e a avaliação experimental dos resultados. A metodologia utilizada foi baseada em pesquisa na internet e utilização do material de apoio disponivel na pagina da cadeira, mais informação disponivel na bibliografia. A solução do problema foi desenvolvida em C por opção dos elementos do grupo.

Descrição da solução

Para resolver o problema introduzido pelos os docentes, usamos como dito previamente na introdução o algoritmo Tarjan, que nos ajudou a agrupar as pessoas como é dito no enunciado do problema. Era nos pedido, para o nosso programa retornar 3 tipos de outputs:

1. Uma linha com o número de grupos máximos de pessoas que partilham informação;

Este ponto corresponde em linguagem corrente a obter o numero de todos os grupos de pessoas que partilham informação, considerando por grupo um conjunto de elementos em que a partir de qualquer elemento do grupo conseguimos atraves de uma ou mais ligações atingir qualquer outro elemento do grupo. Começamos por criar um stack onde os elementos são colocados após a execução do algoritmo tarjan , a contagem dos grupos descritos acima foi feita atraves de contador que é incrementado cada vez que temos o POP de um grupo (SCC) completo, sabemos isto quando acontece o POP da cabeça do grupo ou seja , o elemento do grupo cujo $d = low$, quando isto acontece sabemos portanto que demos POP da cabeça do grupo e portanto podemos incrementar o contador de SCC pois o ultimo elemento deste grupo ja saiu da pilha e devido ao algoritmo tarjan sabemos que uma SCC esta completa.

2. Uma linha com o número do maior grupo máximo de pessoas que partilham informação;

Este ponto corresponde em linguagem corrente a obter o tamanho do maior grupo de pessoas que partilham informação entre sim , o metodo de resolução foi bastante parecido ao do ponto anterior , apenas com uma pequena diferença , nós sabemos que o algoritmo tarjan apenas começa a dar pop da lista quando encontra uma SCC completa , com base nisto podemos colocar um contador que é incrementado cada vez que existe o POP de um elemento , quando é dado POP da cabeça do grupo ou seja o elemento cujo $d = low$, sabemos que aconteceu POP do SCC completo , com isto ficamos a saber por quantos elementos era composto esse SCC , de seguida efectuamos uma comparação do nosso contador com uma variavel global inicializada a 0 e contendo a informação do maior grupo ja encontrado ate agora, ou seja caso este grupo que acabou de dar POP seja maior que o nosso registo de maior grupo ja encontrado , o valor do registo é alterado para o tamanho do grupo que acabou de dar POP , caso o valor que ja se encontra nessa variavel seja maior , quer dizer que um grupo maior ja aconteceu e portanto não fazemos nada , mantendo o valor da variavel inalterado.

3. Uma linha com o número de grupos máximos de pessoas que partilham informação apenas dentro do grupo;

Este ponto, foi o mais complicado de chegar a uma conclusão e não conseguimos, sequer achar a melhor em termos de tempo e eficiência, mas a nossa solução foi, quando dávamos POP dos elementos à procura de SCC's completos, iamos guardando estes nós numa lista do tipo FILO e activando um flag(1) dentro de cada nó que nos avisava que esse nó estava num SCC, quando encontrávamos o SCC iamos verificar se cada elemento da lista que tínhamos enchido não trocava informação com um elemento com a flag a 0. Se não trocasse incrementamos uma variável global para ir para o output.

Análise teórica

Para este projecto, optamos por usar o algoritmo Tarjan, que agrupa elementos de um grafo dirigido em grupos intitulados por SCC's. O que se segue é uma descrição do algoritmo como o implementamos no nosso projecto.

Descrição do algoritmo: O algoritmo começa a analisar determinado nó de um grafo, no nosso caso vai começar sempre no nó 1, este vai percorrendo os nós marcando-os com a distância(d) percorrida até chegar aquele nó assim como um low que é a menor distância para chegar do nó inicial até ao SCC criado, embora no início antes de serem encontrados grupos o low vai ser igual ao d e igual ao visited, que é um inteiro incrementado sempre que um nó é visitado. Em seguida o nó é inserido numa stack, para ser reutilizado mais tarde, em seguida vê-se o primeiro nó adjacente ao nó que está a ser visitado, se esse nó ainda não foi visitado, é feita outra chamada recursiva à função com esse nó para ser marcado como visitado. Sempre que se encontra um nó já visitado o low dos dois nós, em questão vai passar a ser o menor entre eles, assim descobrindo um ciclo. Quando não houver, mais caminhos não explorados a partir do nó inicial, o algoritmo passa para um ciclo onde vai dar POPs da stack, que foi sendo ocupada previamente por nós que foram visitados, e sempre que se der POP e se encontrar um nó com a $d = low$ encontra-se a cabeça de um SCC e assim funciona o Tarjan.

Avaliação Experimental dos Resultados

```
6
1
1

real    0m0.001s
user    0m0.000s
sys     0m0.000s

1
6
1

real    0m0.001s
user    0m0.000s
sys     0m0.000s

1
10000
1

real    0m0.007s
user    0m0.000s
sys     0m0.004s

9996
5
3646

real    0m0.008s
user    0m0.004s
sys     0m0.000s
```

Bibliografia

- Slides aulas teoricas (disponivel na pagina da disciplina)
- Wikipedia (tarjan's Strongly connected components algorithm)
- www.cprogramming.com
- www.eventhelix.com
- Foruns para duvidas relacionadas com a linguagem