

# Relatório 1º projecto ASA 2019/2020

**Grupo:** al026

**Aluno(s):** António Romeu Pinheiro (92427) e Mariana Cosme Cintrão (93737)

## Descrição do Problema

O problema descrito no enunciado trata o algoritmo que prevê as notas dos alunos na disciplina de ASA tendo em conta a nota que obtiveram no primeiro teste. Para além disso, para o cálculo das notas, é necessário ter em conta as relações entre alunos uma vez que, um aluno que tenha uma relação de amizade com um colega cuja nota é mais alta, fica com a nota desse colega.

A partir de um ficheiro de entrada, que contém informação acerca do número de estudantes (vértices), número de amizades (arestas), relações entre estudantes e notas de cada estudante, retorna-se a estimativa das notas de cada aluno.

## Descrição da Solução

Primeiramente é criado um grafo, que é uma classe, através da informação dada no input. Para solucionar o problema do enunciado aplicámos primeiro o algoritmo de Tarjan que, para além de guardar um grafo compacto (no qual cada SCC é representado como se fosse apenas um vértice), propaga as notas dos estudantes.

Quando um SCC é descoberto, a sua nota é adicionada a um vetor de inteiros (de tamanho  $n$  SCCs) no índice respetivo e são também adicionados a uma lista de adjacências os índices dos SCCs adjacentes a este. Após a aplicação do Tarjan modificado, aplicamos um algoritmo DFS sobre o grafo compacto para as notas serem atualizadas. Existe também um vetor de inteiros de tamanho  $n$  estudantes, que associa os estudantes ao respetivo SCC.

## Análise Teórica

Admitindo  $V$  como o número de vértices (estudantes) e  $E$  como o número de arestas (amizades) a solução proposta possui as seguintes complexidades:

- Complexidade Temporal:
  - Leitura do input:  $O(V+E)$
  - Guardar as notas dos alunos:  $O(V)$  - `std::vector::push_back` tem complexidade de  $O(1)$  e é executado  $V$  vezes
  - Guardar relações na lista de adjacências:  $O(E)$  - `std::vector::push_back` tem complexidade  $O(1)$  e é executado  $E$  vezes
  - Aplicação do algoritmo Tarjan:  $O(V+E)$

# Relatório 1º projecto ASA 2019/2020

**Grupo:** al026

**Aluno(s):** António Romeu Pinheiro (92427) e Mariana Cosme Cintrão (93737)

- Aplicação do algoritmo DFS:  $O(V+E)$
- Apresentação dos dados:  $O(V)$  - percorrer o vetor de inteiros que guarda as notas, de tamanho  $V$
- Total:  $O(5V+4E) = O(V+E)$
- Complexidade Espacial:
  - Vector de inteiros que guarda as notas:  $O(V)$
  - Array de vector de inteiros (lista de adjacências de vértices):  $O(V+E)$
  - Vetor de vetor de inteiros (lista de adjacências de SCCs):  $O(V+E)$ ;
  - Vetores que guardam notas de cada SCC, inteiros (disc), inteiros (lows) e booleanos (stackMember):  $O(V)$ ;
  - Array de inteiros que relaciona estudantes com a respetiva SCC:  $O(V)$ ;
  - Total:  $O(8V+2E) = O(V+E)$
- Total Global:  $O(V+E)$

## Avaliação Experimental dos Resultados

1ª experiência: O grafo gerado é composto por igual número de vértices e de arestas.



2ª experiência: O grafo gerado tem um número de vértices constantes, igual a 10000, sendo que o número de arestas vai incrementando.

De acordo com a nossa análise teórica, os resultados obtidos, em ambas as experiências, vão de acordo ao esperado uma vez que, tal como podemos observar através dos gráficos, os tempos de execução vão aumentando linearmente com os números de vértices e arestas.