

# Relatório 1º projecto ASA 2021/2022

**Grupo:** tp015

**Aluno(s):** José Cutileiro (99097) e João Costa(99088)

---

## Descrição do Problema e da Solução

**Entender o problema:** É nos dada uma árvore genealógica, com ela devemos encontrar o ancestral comum mais próximo entre dois vértices (à escolha do user).

**Descrição da solução:** A solução obtida é baseada num sistema de cores para cada 'node'. Esta abordagem ajudou bastante na depuração e aperfeiçoamento do algoritmo desenvolvido durante o processo de codificação. De modo a nunca perder a semântica do problema o código está quebrado em várias etapas.

### Descrição por etapas:

1. Leitura do input: Aqui criamos um grafo transposto ao grafo representante da árvore genealógica original e damos 'init' nos nodes que compõem a árvore. (init: Todos os nodes começam com a cor branca)
2. Verificações: Primeiro verificamos que nenhum dos nodes tem mais que dois filhos (reparem que filhos no grafo original representam pais, nas próximas descrições de etapas estaremos a descrever o problema em relação ao grafo transposto). Caso a primeira verificação esteja correta, passamos para a segunda verificação -> verificar que o grafo não tem ciclos. Para isto basta utilizar o algoritmo DFS com uma verificação de cores (i.e Caso estejamos a visitar um node cinzento, a árvore terá um ciclo necessariamente)
3. Avançar com o primeiro vértice: Todos os filhos e descendentes dos filhos (e descendentes dos descendentes ...) do primeiro vértice inserido passam a cor VERMELHO. Para isto é possível usar uma outra variante da DFS
4. Avançar com o segundo vértice: Percorrer os filhos do segundo vértice, desta vez com um pouco mais de cuidado i.e Se estivermos num vértice branco passar para os filhos; Se estivermos num vértice vermelho passar o vértice para verde, e os filhos, do novo verde, para preto; Se estivermos num vértice preto, passar os seus filhos para preto; Se estivermos num vértice verde, ignorar.
5. Resposta final: Reparar que os vértices verdes são os ancestrais comuns mais próximos da árvore original, como queremos por ordem basta percorrer o vetor de vértices uma última vez, devolvendo os verdes.

# Relatório 1º projecto ASA 2021/2022

**Grupo:** tp015

**Aluno(s):** José Cutileiro (99097) e João Costa(99088)

---

## Análise Teórica

Definições: V - Representa o número de vértices

E - Representa o número de aresta

- Leitura dos dados de entrada:
  - Inicializar vértices:  $O(V)$
  - Escrever por cima dos vértices:  $O(E)$
- Verificação de ciclos no grafo transposto usando uma DFS:  $O(V + E)$
- Percorrer uma DFS partindo do primeiro vértice:  $O(V + E)$
- Percorrer uma DPS partindo do segundo vértice:  $O(V + E)$

Complexidade global da solução:  $O(V+E)$

## Avaliação Experimental dos Resultados

É fácil reparar que os resultados obtidos vão de acordo com os resultados esperados. A complexidade da solução esperada é de  $O(V+E)$  ou seja linear. Já a curva também é linear, comprovando a análise teórica.

