

Relatório 2º projecto ASA 2021/2022

Grupo: al104

Aluno(s): Manuel Pereira (98580)

Descrição do Problema

Neste relatório pretendemos analisar a solução proposta para o segundo projeto da unidade curricular de Análise e Síntese de Algoritmos.

O problema apresentado tem por objetivo determinar se um dado grafo dirigido forma uma árvore genealógica e nesse caso, determinar o conjunto de ancestrais comuns mais próximos entre dois vértices dados.

Descrição da Solução:

De modo a resolvermos o problema apresentado, desenvolvemos uma solução que foi implementada em linguagem C++.

Nesta implementação cada nó representa uma pessoa e os vizinhos diretos de um nó correspondem aos seus filhos.

De forma a determinar se o grafo dado forma uma árvore genealógica, começamos por verificar o número de progenitores que cada filho possui, através de uma lista de adjacências do grafo transposto, onde verificamos, para cada nó, o número de nós adjacentes. Caso esse número fosse superior a 2, então chegamos a conclusão que o grafo não forma uma árvore genealógica. Caso o número de nós adjacentes fosse inferior ou igual a 2, nada podemos garantir, pois falta provar que o grafo não possui ciclos.

Para verificar a existência de ciclos foi realizada uma procura em profundidade utilizando o algoritmo DFS (Depth-First Search), com pequenas alterações. Inicialmente marcamos todos os nós como brancos (0). Para cada nó branco, começamos a DFS, marcando esse nó como cinzento (1) no início e como preto (2) no fim. Se durante a execução da DFS existe uma visita a um nó cinzento, então podemos concluir que existe um ciclo e portanto, o grafo não forma uma árvore genealógica. Caso após a execução da DFS não tenha sido identificado nenhum ciclo, concluimos que o grafo dado forma uma árvore genealógica.

Após a confirmação da existência de uma árvore genealógica precisamos de encontrar os ancestrais comuns mais próximos entre os nós que nos foram dados como input. Para tal, utilizando a lista de adjacências do grafo transposto, fizemos duas iterações do algoritmo DFS, uma para cada nó que pretendemos determinar os ancestrais comuns mais próximos, marcando os nós visitados na primeira iteração como azuis (4) e os nós visitados na segunda iteração como vermelhos (5). No final das duas iterações, verificamos todos os nós do grafo e caso o nó tenha sido visitado pelas duas DFS (≥ 9), aumentamos o valor de todos os nós adjacentes a esse nó. Por fim, os ancestrais comuns mais próximos entre os dois nós dados como input serão todos os nós em que o seu valor é igual a 9, na tabela dos nós visitados.

Relatório 2º projecto ASA 2021/2022

Grupo: al104

Aluno(s): Manuel Pereira (98580)

Análise Teórica

Leitura dados – $O(E)$

Aplicação do Algoritmo – $O(V + E)$

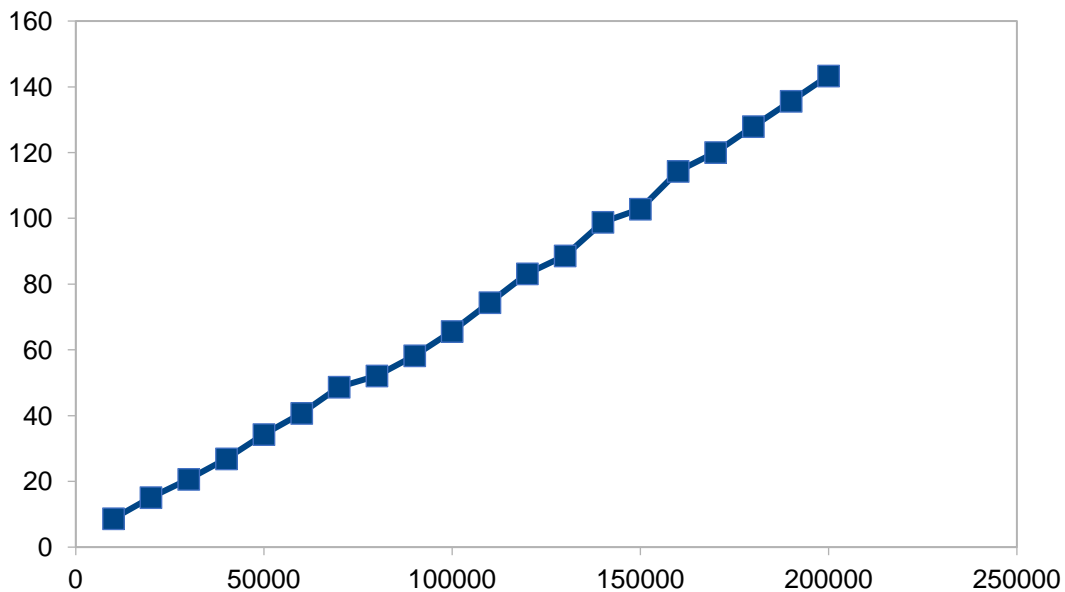
Apresentação Dados – $O(V)$

Complexidade Global Temporal – $O(V+E)$

Complexidade Espacial – $O(V+E)$

Avaliação Experimental dos Resultados

Foram geradas 20 instâncias, de tamanho incremental de 10000 elementos e com um tamanho inicial de 10000 elementos, sendo os tempos obtidos através da funcionalidade time do Linux.



O gráfico gerado aparenta ser de complexidade linear, pelo que podemos concluir que é concordante com a complexidade teórica prevista.