
Análise e Síntese de Algoritmos 2016/2017

1º Projecto - Relatório

Christian Sousa - 67381

João Mak - 69613

Grupo 96



Introdução

Neste projecto o objectivo é criar um programa que faça a ordenação de fotografias (em que cada uma é identificada por um identificador único) recebendo no input várias linhas onde a primeira é referente ao número de fotografias a ordenar bem como o número de pares para os quais é conhecida a ordem e as restantes linhas contêm os pares de fotografias para os quais é conhecida a ordenação significando que o primeiro do par ocorreu antes da segunda.

Solução

- Fazer uma DFS de forma a obter uma árvore DF (que corresponde a uma organização temporal das fotos);
- Se for encontrado um back edge, o input é incoerente (informação contraditória, impossível de obter hierarquia temporal total);
- Se for encontrado um cross edge (mais que uma árvore), o input é insuficiente para estabelecer uma hierarquia temporal total;
- Se forem encontrados cross E back edges, prioridade é dada á incoerência;
- Em caso contrário, o output corresponde á ordem das fotos (da mais antiga á mais recente).

Análise Teórica

Começámos por tentar descrever o problema em termos algorítmicos, de forma a facilitar a modelação da melhor solução. Decidimos que este problema podia ser resumido a um problema de análise e operações sobre grafos (mais especificamente, grafos dirigidos acíclicos).

Podemos então criar um paralelismo:

- Cada foto corresponde a um nó de um grafo;
- Cada arco (dirigido) corresponde a uma hierarquia temporal entre os nós correspondentes (nó origem corresponde à foto mais antiga, nó destino correspondente à foto mais recente).
- Uma árvore DF corresponde a uma ordenação temporal das fotos, sendo a raíz a foto mais antiga.

Depois de estabelecido este paralelismo, o próximo passo foi perceber qual a melhor solução para o problema neste contexto. Chegámos a algumas conclusões:

- Se existe um back edge (na árvore DF), corresponde a uma incoerência (foto simultaneamente mais recente e mais antiga que outra).
- Se existe um cross edge, corresponde a uma insuficiência de informação (existe pelo menos um par de fotos para os quais não é possível estabelecer uma hierarquia temporal).

Assim sendo, o algoritmo para resolver o problema torna-se trivial:

- Fazemos uma pesquisa em profundidade (DFS) sobre o input de N nós e L arcos (1º linha do input), tomando para cada arco as linhas 2 a $L+1$;
- Se encontramos um back edge, termina e devolve 'Incoerente'.
- Termina a DFS e de seguida verifica se a organização topológica é única. Se não for, devolve Insuficiente.
- Em caso contrário, devolve a ordenação topológica.

Complexidade:

- Criar grafo: $O(E)$
- DFS: $O(V+E)$
- Verificação da ordem topológica: $O(V)$
- Complexidade total: $O(V+E)$

Avaliação Experimental dos Resultados

Para realizar a avaliação experimental foram utilizados os exemplos dados no enunciado do projecto assim como o gerador de instâncias disponível na página da disciplina.

Para testar os casos em que o programa deverá ter como output **"Incoerente"** foram utilizados os seguintes inputs:

Input 1: (4,4),(1,2),(2,3),(3,2),(3,4)

Input 2: (5,6),(1,2),(2,3),(3,2),(3,4),(1,5),(5,4)

Para testar os casos em que o programa deverá ter como output **"Insuficiente"** foram utilizados os seguintes inputs:

Input 3: (4,4),(1,2),(3,1),(3,4),(4,2)

Input 4: (5,3),(4,1),(3,2),(5,4)

Para testar os casos em que o programa deverá ter como output a **única ordenação possível** dos inputs, foi utilizado:

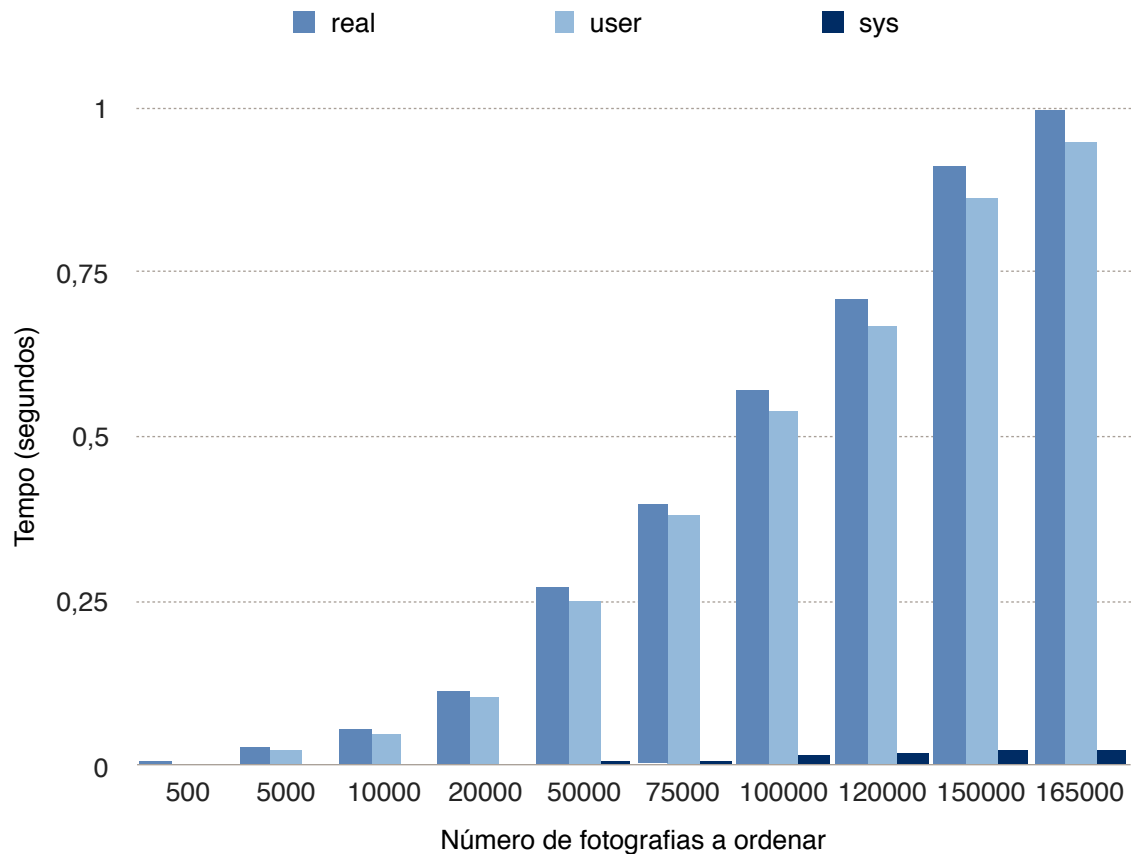
Input 5: (5,4),(1,2),(2,3),(3,4),(4,5)

Input 6: (10,9),(8,1),(6,5),(10,2),(1,4),(2,8),(7,6),(3,7),(9,3),(4,9)

No caso do input 5 o output foi: 1 2 3 4 5

No caso do input 6 o output foi: 10 2 8 1 4 9 3 7 6 5

Foram também testados inputs nos quais a ordenação é única, mas neste caso contendo 500, 5000, 10000, 20000, 50000, 75000, 100000, 120000, 150000 e 165000 fotografias para ordenar, e recorrendo ao comando de terminal *time* foram medidos os tempos(em segundos).



real: tempo real usado pelo processo, em segundos.

user: tempo total de segundos do CPU usados pelo processo em modo utilizador.

sys: numero total de segundos de CPU usado em modo núcleo.