

Ordenação topológica usando filas

Ordenação Topológica

Suponha o problema de se vestir pela manhã. Existe uma lista de roupas/objetos que devem ser usados e certamente deve haver alguma ordem na sua apropriação. Suponhamos a seguinte lista:

- * meia esquerda
- * meia direita
- * sapato esquerdo
- * sapato direito
- * galocha esquerda
- * galocha direita
- * cueca ou calcinha
- * calça ou saia
- * camiseta
- * camisa
- * paletó
- * pulseira
- * anéis
- * luva direita
- * luva esquerda
- * relógio
- * óculos
- * tiara
- * chapéu ou gorro
- * cinto
- * sutiã (se for o caso)

O que se pode afirmar sobre este problema é a existência de duplas de tarefas, onde a primeira deve ser feita antes da segunda. Por exemplo: (calça,cinto) ou (camiseta,camisa) ou ainda (meia esquerda, sapato esquerdo). Obtida a lista completa de dependências, escrever uma sequência de tarefas que as obedeçam (às dependências) é obter uma ordenação topológica das tarefas. A ordenação topológica é usada sempre que se precisa colocar objetos/eventos em ordem parcial.

x e y estão em ordem parcial se x deve ser feito antes de y

A ordem parcial apresenta as seguintes propriedades:

- 1. Se *x* deve ser feito antes de *y* e *y* deve ser feito antes de *z*, então: *x* deve ser feito antes *z* (propriedade transitiva).
- 2. Se *x* dsfa *y* então *y* NÃO dsfa *x* (propriedade assimétrica).
- 3. *x* NÃO dsfa *x* (propriedade irreflexiva).

São exemplos de atividades que devem sofrer ordenação parcial:
Montagem dos componentes de um veículo;
Execução de diversas disciplinas de um curso universitário com pré-requisitos;
Construção de uma ponte;
Enchimento do carrinho de supermercado,

O algoritmo de ordenação topológica trata de extrair uma ordenação linear de um conjunto de objetos/eventos que deve ser parcialmente ordenado. Isto é: Se A_j dsfa A_k então $j < k$.
Note-se que a ordenação parcial não necessariamente é única, razão pela qual o exercício a seguir sugere uma ordenação inicial, a fim de que o gabarito possa ser gerado e usado na correção.

Um pouco de história Antes de estudar o algoritmo, vale a referência de que ele está por trás do método PERT/CPM, uma das grandes conquistas da logística humana, desenvolvido na indústria bélica americana na década de 50 (especificamente para o projeto POLARIS, que tinha as seguintes características: 250 empreiteiros; 9.000 sub-empreiteiros; 70.000 peças diferentes nunca antes fabricadas; Atraso no projeto inaceitável, e que foi concluído com sucesso notável).
Para os que não lembram, (ou não estudaram, já que não eram nascidos ainda), o mundo vivia

em sobressalto, motivado pelo conflito leste oeste (capitalismo versus comunismo), origem da guerra fria que tinha seu epicentro em Berlim Ocidental, cujo mapas vemos aqui:



Chamou-se guerra (pois foi um efetivo confronto humano, no qual morreram milhões de seres humanos) mas foi também fria, já que os belicosos sempre foram terceirizados, nunca os dois blocos se confrontaram diretamente, (mas foi por muito pouco). Alguns episódios desta guerra: Revolução Chinesa (1949); Guerra da Coreia (1950-1953); Hungria (1956) Cuba (1958); Guerra do Vietnam (1961-73); Golpe no Brasil (1964); Primavera de Praga (1968); Golpe no Chile (1973); Golpe na Argentina (1976); Invasão no Afeganistão (1980), para ficarmos só nos principais.

Este conflito foi diferente de todos os havidos anteriormente na história da humanidade, porque neste apareceram as armas atômicas. Começando com a bomba de Hiroshima em 1945, pelo desenvolvimento americano da bomba H, cerca de 500 vezes mais potente, em 1951 e soviético em 1954. Peritos calcularam que nos anos 80, o arsenal americano e soviético juntos poderiam destruir o nosso planeta 1200 vezes. A situação só se descomprimiu um pouco no final da década de 80, quando o sistema soviético ruiu por completo e a URSS desapareceu, surgindo em seu lugar cerca de 15 repúblicas na região. A maior é a Rússia.

Algoritmo de ordenação topológica

O funcionamento do algoritmo é simples: Busca-se um elemento que não tenha predecessor (sempre o haverá), retira-se-o do conjunto, colocando-o na saída. O conjunto restante ainda segue uma ordem parcial.

Repete-se o processo até que todos os objetos/eventos estejam na saída. A entrada do algoritmo é um conjunto de pares (a,b) que deve ser lido como "a deve ser feito antes de b".

Construa uma estrutura de dados na forma de uma tabela formada por

coluna 1 identificação da tarefa na ordem sugerida pelo exercício

coluna 2 quantidade de vezes que esta tarefa é "b" nos pares (a,b)

cols 3-10 valores de "b" quando o valor da coluna 1 é "a" em (a,b)

Por exemplo (12 nodos e 15 arestas):

(6 9) (2 4) (3 8) (2 3) (2 5) (8 9) (5 7) (6 7) (9 12) (8 11) (1 3) (3 4) (1 2) (5 10) (5 6)

Com estes dados a matriz ficaria (col 1 sugerida: números naturais)

1	0	3	2						
2	1	4	3	5					
3	2	8	4						
4	2								
5	1	7		10	6				
6	1	9	7						
7	2								
8	1	9	11						
9	2	12							
10	1								
11	1								
12	1								

Por exemplo, acompanhemos a montagem da primeira linha. Ela se refere à tarefa 1:

A coluna 2 é igual a zero, não há nenhuma antecessora para esta tarefa (não há nenhum par x,1)

Já as colunas 3..10 têm os valores 3 e 2, pois existem os pares (1,3) e (1,2), nesta ordem. (ou seja, 1 deve ser feita antes da 3 e da 2)

Obs: Para os algoritmos de ENFILEIRA e DESENFILEIRA use aqueles tradicionais vistos em sala de aula (linear/circular/encadeada, tanto faz).

```
1: Algoritmo RESOLVE-ORDENAÇÃO-TOPOLOGICA
2: para I de 1 até TAM
3:   se TAB[I;2] = 0
4:     ENFILEIRA (F,TAB[I;1])
5:   fim{se}
6: fim{para}
7: enquanto nãovazia(F)
8:   VEZ ← DESENFILEIRA (F)
9:   imprima VEZ
10:  enquanto VEZ tiver sucessores
11:    subtraia 1 do sucessor {os sucessores estão nas cols 3..10}
12:    se TAB[sucessor;2] = 0
13:      ENFILEIRA (F,sucessor)
14:    fim{se}
15:  fim{enquanto}
16: fim{enquanto}
17: fim algoritmo
```

Resposta para este caso:

1, 2, 3, 5, 8, 4, 10, 6, 11, 9, 7, 12

Outro caso exemplo: (12 nodos, 15 arestas)

(13 8) (8 7) (20 6) (19 3) (4 1) (7 11) (19 11) (3 20) (15 6) (6 14) (1 6) (7 19) (8 4) (4 15) (3 14)

Sequência sugerida:

13, 8, 7, 19, 11, 3, 20, 4, 15, 1, 6, 14

terá como resposta:

13, 8, 7, 4, 19, 1, 15, 3, 11, 20, 6, 14

OBS: 1. A sequência sugerida, JA É A COLUNA 1.

2. Como se pode ver no exemplo, em caso de empate (mais de uma tarefa com zero na coluna 2), enfileire ANTES a tarefa de baixo na tabela.

Para você fazer

Conjunto com 12 nodos e 15 arestas. Sejam os pares:

(8,12) (18,20) (15,16) (3,16) (20,10) (12,15) (18,3) (12,19) (4,18) (19,13) (4,19) (11,8) (15,18) (10,13) (8,4)

Coluna 1 sugerida:

11 8 12 4 15 18 3 19 16 20 10 13

Resposta:

--	--	--	--	--	--	--	--	--	--	--	--



Bibliografia recomendada: Szwarcfiter, J e Markenzon L. ED e seus algoritmos, pag. 47-51 Knuth, D. The Art of Computer Programming. v1. pags 258-265