

TP2 - Grafos temporais

17 pontos

Entrega: 22/10/2023

1 Objetivo do trabalho

Neste trabalho, vamos exercitar tópicos de alguns problemas de otimização em grafos. Para tal, trabalharemos com *grafos temporais*, uma modelagem que permite estudar a evolução de alguns sistemas.

Serão fornecidos alguns casos de teste para que você possa testar seu programa, mas eles não são exaustivos! Podem haver situações que não são ilustradas por eles; cabe a você pensar em novos casos e garantir que seu programa esteja correto e implemente um algoritmo de complexidade adequada.

1.1 Informações importantes

O código fonte do seu trabalho deve estar contido em um **único** arquivo na linguagem C++ e deve ser submetido via Moodle na tarefa **Entrega TP2** até o dia 22/10/2023. Você terá **20 tentativas** para conseguir a nota total de execução; apenas a última submissão será levada em conta para fins de avaliação. Você não terá acesso a todos os casos de teste; determinar estratégias para testar seu programa e suas ideias faz parte do trabalho. Envios com atraso serão aceitos; leia a Seção 4 para a política de atrasos.

Plágio de qualquer natureza não será tolerado. Caso qualquer cola seja encontrada, seu trabalho será zerado e as demais providências cabíveis serão tomadas. Escreva seu próprio código, de maneira legível e com comentários apropriados; ele pode ser útil no futuro próximo.

2 Definição do problema

A Baicônia é uma nação idílica, governada por uma dinastia magnânima de capivaras fundada por Bacon - o Grande - há várias gerações. Além de sua vasta riqueza cultural construída a partir da filosofia Deboísta, que advoga pela manutenção da tranquilidade em todas as relações sociais, a Baicônia também é uma grande centro para o desenvolvimento da combinatória devido aos grandes investimentos iniciados por Bacon - o Grafo - e que agora finalmente estão encontrando aplicações nas mãos de Bacon - a Construtura.

Em um momento de puro brilhantismo, nossa heroína vislumbrou uma vasta rede de canais e estradas que pudessem rapidamente interconectar todo o reino, trazendo prosperidade econômica e maximizando o tempo que sua população poderia praticar a atividade favorita do reino: ficar de boa na lagoa. A Construtura, porém, sabe que essa será uma empreitada de vários anos, e já determinou em qual ano que cada conexão entre vilas será finalizada. Além disso, Bacon sabe quanto tempo cada via gasta para ser percorrida e o custo de construção da mesma. Por exemplo, na figura abaixo, temos que o canal entre as vilas a e b será completado no ano 3, seu tempo de travessia é de 4 horas, e o custo de construção é de 5 dinheiros.

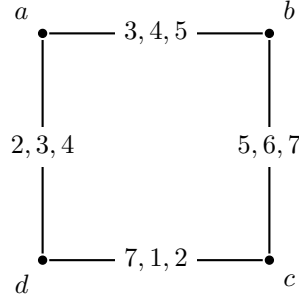


Figure 1: As quatro vilas da Baicônia e os parâmetros de cada conexão projetada por Bacon - a Construtura.

Para dar início a seu grandioso projeto de infraestrutura, Bacon - a Construtura - precisa de aprovação do Conselho das Capivárias, uma organização da sociedade civil responsável por gerir o orçamento do reino e garantir que a monarquia não se torne uma tirania absolutista. O Conselho gostaria das seguintes informações:

1. Após a conclusão do projeto, qual será a distância mínima do palácio real (localizado na vila de índice 1) para cada outro vila?
 - (a) Qual o primeiro ano em que todas essas distâncias podem ser realizadas ao mesmo tempo?
2. Qual o primeiro ano em que todo o reino alcançável a partir do palácio real?
3. Se o projeto fosse alterado para apenas garantir que seja possível chegar de qualquer vila a qualquer outra, qual seria o menor custo possível?

Bacon - a Construtura - recebeu essas exigências do Conselho e prontamente lhe passou essa tarefa. Claramente, ela tem mais o que fazer, como descobrir uma maneira de desafiar as leis da física e construir estradas que lhe permitam voltar no tempo.

3 Casos de teste

3.1 Formatado da Entrada

Cada caso de teste é composto por várias linhas. A primeira linha contém dois inteiros, N e M , que representam, respectivamente, o número de vilas e conexões da Baicônia. É garantido que as vias conectam formam caminhos entre todo par de vilas. Você pode assumir que $1 \leq N, M \leq 10^6$, e que as vilas são numeradas com elementos do conjunto $\{1, 2, \dots, N\}$. Cada uma das M linhas seguintes descreve uma conexão entre um par de vilas, e contém cinco inteiros u, v, a, ℓ, c , onde u e v são vilas ($1 \leq u \neq v \leq N$), a é o ano de construção da via ($1 \leq a \leq 10^8$), ℓ representa o tempo de travessia da via ($1 \leq \ell \leq 10^5$), e c representa seu custo de construção ($1 \leq c \leq 10^5$).

Assuma que não exista nenhuma outra conexão entre vilas e que Bacon - a Construtura - criou seu plano brilhante no ano 0.

A entrada deve ser lida da entrada padrão.

3.2 Formato da Saída

A saída contém múltiplas linhas. Na i -ésima das primeiras N linhas deve ser impresso um inteiro que representa o tempo mínimo necessário para sair do palácio real e chegar á vila i . Na $N + 1$ -ésima linha da saída deve ser impresso um único inteiro A_1 , que representa o primeiro ano no qual as distâncias listadas nas linha anteriores são mutuamente realizáveis. A $N + 2$ -ésima linha da saída contém um único inteiro A_2 , que representa o primeiro ano a partir do qual é possível chegar em qualquer vila do reino a partir do palácio real. Por fim, na linha $N + 3$, imprima um único inteiro, que representa o menor custo necessário para conectador todo o reino.

A saída deve ser escrita na saída padrão.

3.3 Limites de execução

Para qualquer caso de teste, seu código deve imprimir a resposta em no máximo 3 segundos. Seu programa deve usar menos de 100MB de memória. Estruturas de dados devem ser alocadas sob demanda; ou seja, não faça vetores estáticos gigantescos para entradas pequenas. Todas as avaliações serão feitas automaticamente via VPL. Programas que não aderirem a essas restrições para um teste terão a nota do mesmo zerada.

Lembre-se: você pode submeter uma solução para a tarefa no máximo 20 vezes e apenas a última submissão será levada em conta para fins de avaliação.

3.4 Exemplos

3.4.1 Exemplo 1

Na figura abaixo, a vila de número i corresponde à i -ésima letra do alfabeto; i.e. $4 = d$.

Entrada	Saída
4 4	0
1 2 3 4 5	4
2 3 5 6 7	4
3 4 7 1 2	3
4 1 2 3 4	7
	5
	11

3.4.2 Exemplo 2

Na figura abaixo, o vértice de número i corresponde à i -ésima letra do alfabeto; i.e. $4 = d$.

Entrada	Saída
6 5	0
1 2 1 7 2	7
2 3 2 8 1	15
3 4 10 1 4	16
4 5 1 3 2	19
5 6 3 9 2	28
	10
	10
	11

4 Atrasos

O trabalho pode ser entregue com atraso, mas será penalizado de acordo com a seguinte fórmula, onde d é o número de dias atrasados:

$$\Delta(d) = \frac{2^{d-1}}{0.32} \% \quad (1)$$

Por exemplo, com um atraso de quatro dias e uma nota de execução de 70% do total, sua nota final será penalizada em 25%, ficando assim igual a $70 \cdot (1 - \Delta(d)) = 52.5\%$. Note que a penalização é exponencial, e um atraso de 6 dias equivale a uma penalidade de 100%.