

**TRABALHO PRÁTICO 2**  
**GRAFOS TEMPORAIS****Observações:**

1. Comece a fazer este trabalho imediatamente. Você nunca terá tanto tempo para resolvê-lo quanto agora!
  2. **Data de entrega:** não antes de 20 de junho de 2024.
  3. **Submissão:** Faça a submissão deste trabalho no Moodle, conforme instruções postadas lá.
  4. **Plataforma computacional:** O seu trabalho será executado na plataforma VPL do Moodle.
  5. **Linguagem:** Você deve escrever o seu programa obrigatoriamente na linguagem de programação C++.  
Não será aceita outra linguagem.
  6. **Documentação:** Veja instruções no Moodle (aguardando a definição do monitor).
  7. **Testes:** O seu programa será avaliado conforme descrito no Moodle da disciplina.
-

## Grafos Temporais

**Introdução.** Antes de falarmos de grafos temporais, vamos falar brevemente de modelagem de sistemas que têm um “comportamento” que varia ao longo do tempo. Por exemplo, suponha uma pessoa usando um telefone celular. Se ela vai se afastando para longe da torre usada naquele momento para se comunicar e não existe outra torre nessa vizinhança, então ela perderá toda a conectividade com outras pessoas, i.e., não poderá fazer chamadas telefônicas, nem ter acesso à Internet, i.e., acessar aplicações que trocam dados (veja Figura 1a).

Suponha agora que temos milhares e milhares de pessoas nesse cenário. O que temos é um sistema dinâmico e complexo onde em momentos específicos podemos ter pessoas sem acesso ou com acesso (veja Figura 1b). Mais ainda, ao longo do tempo as pessoas podem ter acesso à rede de telefonia celular através de diferentes torres e, novamente, temos mais esse aspecto dinâmico. **um bom exemplo é o acesso na internet na rede da UFMG**

(que um n° de IP limitado)



Figura 1: Rede de telefonia celular é um sistema dinâmico

**Tipos de grafo usados na modelagem de sistemas dinâmicos.** Existem pelo menos três tipos de grafos usados na modelagem de sistemas dinâmicos. São eles:

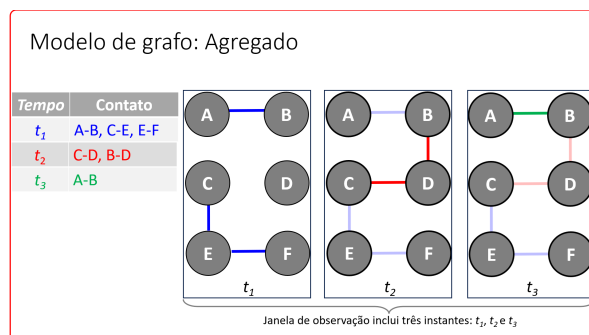
### 1. Agregado:

- Esse modelo cria um grafo de contatos durante um período de observação chamado de janela. Contato aqui representa alguma propriedade do sistema modelado. No caso da telefonia celular, esse contato pode representar a conectividade entre o telefone celular e uma torre de comunicação. No exemplo da Figura 2a, temos três instantes ( $t_1$ ,  $t_2$  e  $t_3$ ) que representam a janela de observação. No instante  $t_1$ , são criadas as arestas azuis entre os vértices A-B, C-E e E-F, que são mantidas nos próximos instantes, mas que não existem posteriormente a não ser a aresta A-B no instante  $t_3$ . O mesmo acontece com as arestas vermelhas entre os vértices C-D e B-D no instante  $t_2$  e depois posteriormente.
- O grafo resultante ao final da janela de observação, neste caso o grafo em  $t_3$ , é usado para modelar o sistema dinâmico durante um período (janela). Assim, o modelo tenta aproximar o que aconteceu no sistema, mas como “agrega” (inclui) todas as arestas que ocorrem durante a janela de observação, apresenta um sistema que não é realista.
- Esse modelo é fácil de processar, mas precisa de mais recursos por conta da agregação.
- O modelo tem uma janela de tempo fixo. No exemplo da Figura 2a temos “três” instantes e a próxima janela agregaria as arestas existentes nos instantes  $t_4$ ,  $t_5$  e  $t_6$ , e, assim, sucessivamente.
- Questão fundamental: qual é o tamanho da janela de tempo?

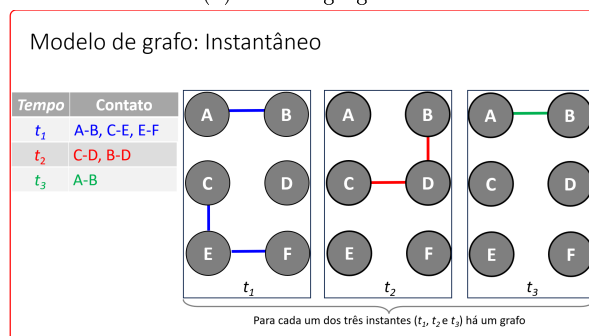
### 2. Instantâneo:

- Esse modelo analisa a topologia em momentos específicos. No exemplo da Figura 2b, há três grafos, um para cada um dos instantes  $t_1$ ,  $t_2$  e  $t_3$ .
- Esses “instantâneos” (*snapshots*) capturam a conectividade existente em um determinado momento e são usados para modelar o sistema dinâmico. Observe que a quantidade de grafos usados nesse processamento pode ser muito maior que no caso do modelo agregado.
- No entanto, é um modelo fácil de processar e que consome menos recursos pois cria o grafo resultante de um *snapshot*.

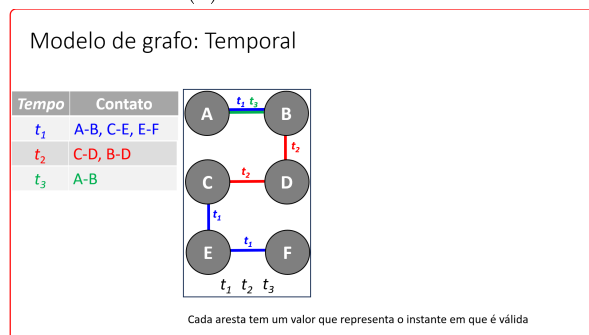
- Questão fundamental: quais são os momentos a serem analisados?
3. **Variável no tempo (grafo temporal):**
- Esse modelo considera as restrições temporais na análise da topologia. No exemplo da Figura 2c, cada aresta possui um valor que indica o intervalo de tempo no qual é válida, ou seja, os instantes  $t_1$ ,  $t_2$  e  $t_3$ .
  - Assim, esse modelo representa exatamente a “realidade” que ocorre no sistema dinâmico e seu processamento reflete o que efetivamente acontece no sistema.
  - Por outro lado, esse modelo tende a ser mais caro em termos de tempo e espaço pois processa tudo o que ocorreu no sistema.



(a) Grafo agregado



(b) Grafo instantâneo



(c) Grafo temporal

Figura 2: Tipos de grafo usados na modelagem de sistemas dinâmicos

Alguns comentários:

- “Instantâneo” pode ser visto como um caso especial de “Agregado” quando a janela de tempo é apenas uma unidade.
- “Agregado” usa uma janela de tempo fixo enquanto “Temporal” usa uma janela deslizante, ou seja, uma janela que considera todos os instantes de tempo que uma aresta existe.
- “Agregado” e “Instantâneo” não modelam a realidade (representam possíveis aproximações, geralmente irrealistas), mas “Temporal” sim, representa fielmente a realidade!

## Detalhamento de grafos temporais

**O que são grafos temporais?** Grafos temporais, também conhecidos como *grafos dinâmicos* ou *grafos com variáveis de tempo*, são um tipo de grafo que representa as relações entre entidades e como essas relações mudam ao longo do tempo. Em contraste com os grafos estáticos tradicionais, onde as relações entre os nós são consideradas constantes, os grafos temporais capturam a natureza evolutiva das interações e dependências.

**Características essenciais dos grafos temporais.** Os grafos temporais são grafos valorados onde as arestas capturam o aspecto temporal. Esses grafos possuem as seguintes características:

- **Dimensão temporal:** Os grafos temporais incorporam uma dimensão adicional de tempo, representando explicitamente quando as relações entre os nós existem ou mudam. Essa dimensão é representada por intervalos de tempo associados às arestas.
- **Relações dinâmicas:** As arestas (conexões) em um grafo temporal podem mudar ao longo do tempo, refletindo a natureza evolutiva das relações sendo modeladas.
- **Aplicações em diversos domínios:** Grafos temporais são usados em uma ampla gama de aplicações, incluindo redes sociais, redes de comunicação, redes biológicas e redes financeiras.

**Componentes de um grafo temporal.** Os grafos temporais possuem os seguintes elementos:

1. **Vértices:** representam as entidades ou objetos de interesse no sistema que está sendo modelado.
2. **Arestas (conexões):** representam as relações entre os nós, indicando que há alguma interação ou dependência entre eles.
3. **Timestamps** ou intervalos de tempo: associam a cada aresta a um timestamp, indicando quando a relação entre os nós correspondentes começa e termina (intervalo de tempo).
4. Outras informações (opcional): informações adicionais sobre as relações, como força, tipo ou direção.

**Modelagem da dinâmica temporal.** Os grafos temporais podem ser modelados usando várias abordagens, incluindo:

- Tempo **explícito:** as arestas são explicitamente rotuladas com timestamps, indicando seus horários exatos de início e término.
- Tempo **discreto:** o intervalo de tempo é dividido em intervalos discretos, e as arestas são associadas aos intervalos durante os quais estão ativas.
- Tempo **contínuo:** o intervalo de tempo é considerado contínuo, e as arestas são modeladas usando funções matemáticas que representam sua ativação ao longo do tempo.

**Aplicações dos grafos temporais.** Os grafos temporais têm uma ampla gama de aplicações, incluindo as seguintes que só conseguem fornecer resultados com qualidade se forem modeladas dessa forma:

- **Análise de redes sociais:** compreender a evolução das conexões sociais, identificar indivíduos influentes e rastrear a disseminação de informações.
- **Análise do tráfego de rede:** monitorar o congestionamento da rede, detectar anomalias e otimizar o desempenho da rede.
- **Análise de redes biológicas:** modelar redes regulatórias de genes, entender as interações de proteínas e prever a progressão de doenças.
- **Análise de redes financeiras:** identificar risco sistêmico, detectar fraudes e compreender a dinâmica do mercado.

**Desafios em trabalhar com grafos temporais.** A análise de grafos temporais apresenta vários desafios, como:

- **Coleta de dados:** coletar e organizar dados temporais pode ser complexo e demorado.
- **Complexidade computacional:** algoritmos para análise de grafos temporais podem ser computacionalmente exigentes, especialmente para grafos grandes e dinâmicos.
- **Visualização:** visualizar efetivamente a evolução de um grafo temporal pode ser desafiador. Uma das estratégias usadas é criar um vídeo de sua evolução.

Os grafos temporais fornecem uma ferramenta poderosa para modelar e analisar sistemas em diferentes áreas do conhecimento que apresentam **relações dinâmicas** ao longo do tempo (e.g., redes sociais, sistemas biológicos, sistemas de transporte inteligentes e sistemas financeiros). Esses grafos permitem obter valiosas informações sobre interações complexas e padrões em evolução dentro desses sistemas. À medida que a pesquisa em análise de grafos temporais continua a avançar, podemos esperar aplicações ainda mais inovadoras e uma compreensão mais profunda dos fenômenos do mundo real.

## Objetivo do trabalho

Neste trabalho, vamos **exercitar tópicos de alguns problemas de otimização em grafos**. Para tal, trabalharemos com **grafos temporais**, como explicado acima.

Serão **fornecidos casos de teste** para que você possa avaliar seu programa, mas eles **não são exaustivos**! Pode 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**.

**Informações importantes.** Veja o Moodle da disciplina para informações sobre a submissão.

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. Você terá **20 tentativas** para conseguir a nota total de execução; apenas a última submissão será considerada 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. Escreva seu próprio código de maneira **legível** e **com comentários** apropriados; **ele pode ser útil no futuro próximo**.

## 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 um grande centro para o desenvolvimento da combinatória devido aos grandes investimentos iniciados por Bacon Grafo, e que agora finalmente estão encontrando aplicações nas mãos de Bacon Construtura.

Em um momento de puro brilhantismo (Bacon Construtura), 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 Bacon 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, **sabe-se quanto tempo cada via gasta para ser percorrida e o seu custo de construção**. Por exemplo, na Figura 3, temos que o canal entre as vilas *a* e *b* será completado ao final do ano 3 (começará no primeiro dia do ano 1, i.e., gastará 3 anos), seu tempo de travessia é de 4 horas, e o custo de construção é de 5 baconzitos (unidade monetária de Baicônia).

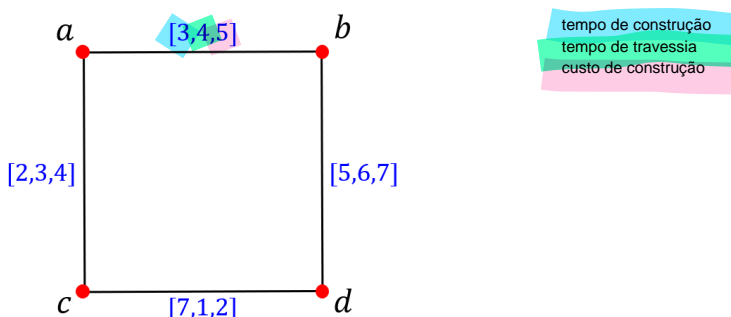


Figura 3: As quatro vilas de Baicônia e os parâmetros de cada conexão projetada por Bacon Construtura

Para dar início a seu grandioso projeto de infraestrutura, Bacon Construtura precisa da 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:

palacio = primeira "vila"?

cada vila tem um palacio diferente? pq o a é o ano final da construção

- Após a conclusão do projeto, qual será a distância mínima do palácio real, localizado na vila de índice  $a$ , para cada outro vila?
  - Qual o primeiro ano em que todas essas distâncias podem ser realizadas ao mesmo tempo?
- Qual o primeiro ano em que todo o reino alcançável a partir do palácio real?
- 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 Construtura recebeu essas exigências do Conselho e prontamente lhe passou essa tarefa, aluno de Algoritmos 1. 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!

### Casos de teste

**Formato 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 formam caminhos entre todo par de vilas, i.e., o grafo resultante é conexo. Você pode assumir que  $1 \leq N, M \leq 10^6$ , e que as vilas são numeradas com elementos distintos do conjunto  $\{1, 2, \dots, N\}$ . Cada uma das  $M$  linhas seguintes descreve uma conexão entre um par de vilas, e contém cinco inteiros  $(u, v, a, l, c)$ :

- $u$  e  $v$  são vilas com identificadores  $1 \leq u, v \leq N$  e  $u \neq v$ ;
- $a$  é o ano final de construção da via ( $1 \leq a \leq 10^8$ );
- $l$  representa o tempo de travessia em horas da via ( $1 \leq l \leq 10^5$ );
- $c$  representa seu custo de construção em baconzitos ( $1 \leq c \leq 10^5$ ).



Assuma que não exista nenhuma outra conexão entre vilas e que Bacon Construtura estava com seu plano brilhante no primeiro dia do primeiro ano.

A entrada deve ser lida da entrada padrão.

CONSTRUÇÃO COMEÇA NO PRIMEIRO DIA DO PRIMEIRO ANO

**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 à 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 linhas 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 conectar todo o reino.

A saída deve ser escrita na saída padrão.

**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 serão penalizados.

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.

### Exemplos

**Exemplo 1.** No trecho abaixo, o vértice 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
	1° ano q todas as conexões estarão feitas
	5
	1°ano palacio -> qualquer vila
	11
	menor custo para conectar todo o reino

**Exemplo 2.** No trecho abaixo, novamente, 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