MAC–IME–USP

Carlos Eduardo Ferreira
Sala 108C Tel.: 3091 6079
e-mail cef@ime.usp.br

MAC 323 – Algoritmos e Estruturas de Dados II

Primeiro semestre de 2023

Reconstrução de DNAs – **Entrega: 12 de junho**

O objetivo deste exercício-programa é implementar algumas funções para manipular grafos, e testar algumas funções que vimos. O problema que usaremos como aplicação é o de reconstrução de sequências de DNA, um problema computacionalmente difícil.

Você deverá entregar, junto com o código que produzir, um relatório em que conta quais foram seus testes, e os resultados obtidos.

Tarefas básicas de grafos

Você deverá implementar funções que:

- lê um arquivo com o seguinte formato, e constroi uma estrutura para representar o grafo dirigido:

```
V E  // número de vértices e arcos
u_1 v_1
u_2 v_2
.
.
.
u_E v_E
```

- verifica se um arco $u - v$ está em um circuito
- encontra em um grafo acíclico um caminho de comprimento máximo.

Você pode implementar outras funções que desejar.

Descrição do problema

Um dos problemas encontrados em Biologia Computacional é o de reconstrução de sequências de DNA. As sequências de DNA, especialmente de organismos mais complexos, têm, muitas vezes, milhões de nucleotídeos, e pode ser vista como uma *string* formada pelas letras "A", "C", "G" e "T".

```
ACTCGTAAATACATAACGATAC
```

A "leitura" destas sequências perde muita precisão em pedaços grandes, com mais de 500 ou 1000 bases. Desta forma, uma técnica usada para reconstruir uma sequência (que pode ter milhões de bases) é a seguinte:

- multiplique a sequência de DNA com a ajuda de vírus (isso pode trazer problemas ao processo, como misturar o DNA do vírus, mas não vamos considerar isso aqui);
- quebre (mecanicamente, como em um liquidificador) as várias cópias do vírus em pedaços pequenos;
- tente "remontar o quebra-cabeças", ou seja, use as partes da sequência para remontar a sequência original.

Este método é conhecido como "método shotgun" para a reconstrução de DNA, e a tarefa complexa é a de montagem dos fragmentos.

No exemplo acima, poderíamos ter quebrado (as cópias da) sequência em 12 fragmentos pequenos:

```
(0) ACTCGT      (1) ATACATAA    (2) TAACGA     (3) ACGAT
(4) TCGTA       (5) AAATA       (6) ATAAC      (7) CGAT
(8) GTAAATA     (9) ACATAA     (10) GATAC     (11) GATAC
```

O problema que o sequenciador vai resolver é, a partir destes 12 fragmentos, reconstruir, da melhor maneira possível, a sequência original.

A abordagem que mostramos aqui é bastante semelhante a de alguns softwares usados em montagens de fragmentos de DNA.

A montagem do grafo

O grafo dirigido que construiremos a partir dos fragmentos tem n vértices, onde n é o número de fragmentos produzidos. O grafo terá um parâmetro k. Teremos um arco entre o vértice u e o vértice v se k ou mais últimas letras do fragmento correspondente a u são iguais às k ou mais primeiras letras de v.

Por exemplo: se $k = 2$ teremos arcos, por exemplo, entre os vértices:

- (0) ACTC**GT** e (8) **GT**AAATA
- (0) AC**TCGT** e (4) **TCGT**A
- (6) ATA**AC** e (9) **AC**ATAA
- (10) **GATAC** e (11) **GATAC**
- (11) **GATAC** e (10) **GATAC**

Encontrando uma solução para o problema

O problema de montagem de fragmentos é, dados os n fragmentos de várias cópias de uma mesma sequência de DNA, encontrar a sequência original. É difícil formular o problema de uma forma precisa, pois pode haver diversas sequências que poderiam dar origem aos fragmentos e não é possível, sem mais conhecimentos, decidir qual é a correta. Uma aproximação frequentemente usada é encontrar a maior sequência que pode ser obtida através da concatenação de fragmentos que têm intersecção (o final de um fragmento é igual ao começo do seguinte) de pelo menos k bases, para um dado k. Mesmo esta aproximação já dá origem a um problema NP-difícil (vocês aprenderão mais sobre estas classes de complexidade no próximo semestre).

Utilizando a modelagem em grafos proposta acima, dado um conjunto de fragmentos e um parâmetro k montamos um grafo. Um caminho neste grafo corresponde a uma sequência de fragmentos que podem ser concatenados para obter uma estimativa do DNA original. Dentro todos os caminhos possíveis, desejamos encontrar o maior caminho possível ou ainda o caminho cuja concatenação dá origem à maior sequência que, esperamos, é uma boa solução para nosso problema.

No exemplo acima, para $k = 2$, os vértices $0 - 8 - 6 - 9$ formam um caminho, que, quando concatenamos os fragmentos, obtemos:

ACTCGTAAATAACATAA

Este problema, infelizmente, é NP-difícil, a menos que o grafo seja acíclico... Assim, vamos ter de nos conformar em obter uma solução heurística para o problema. Se o grafo for acíclico, conhecemos algoritmos para encontrar um caminho de comprimento máximo. Se existirem arestas que estejam em circuitos podemos tentar removê-las. Mas, note que isso pode dar origem a várias soluções diferentes conforme a ordem em que estas arestas são removidas.

Mais uma vez o relatório que vocês prepararão é bastante importante. Diga como você gerou seus testes, e qual estratégia heurística utilizou para remover arestas de circuitos antes de encontrar um caminho máximo. Compare seus resultados com a resposta esperada (uma vez que você gerou testes, sabe a resposta do problema, né?).