

Profa. Dra. Raquel C. de Melo-Minardi  
Departamento de Ciência da Computação  
Instituto de Ciências Exatas  
Universidade Federal de Minas Gerais

	0	1	2	3	4	5	6	7	8	9	10
0	*	←	*	←	*	←	*	←	*	←	*
1	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
2	*	←	*	←	*	←	*	←	*	←	*
3	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
4	*	←	*	←	*	←	*	←	*	←	*
5	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
6	*	←	*	←	*	←	*	←	*	←	*
7	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
8	*	←	*	←	*	←	*	←	*	←	*
9	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
10	*	←	*	←	*	←	*	←	*	←	*

# MÓDULO 4

## ALGORITMOS PARA BIOINFORMÁTICA

### Jogo das fichas

## JOGO DAS FICHAS

- ▶ Alice e Bob tem duas pilhas de fichas e estão a jogar o seguinte jogo:
  - ▶ A cada rodada um jogador pode retirar uma pedra de uma pilha ou retirar uma pedra de cada uma das duas pilhas;
  - ▶ A seguir, é a vez do outro jogador;
  - ▶ Ganha a partida quem tirar a última pedra
  - ▶ Alice é a primeira a jogar

### **Desafio**

Pense sobre o problema e nos diga se:

1. Há uma estratégia para vencer o jogo?
2. Algum dos jogadores (o primeiro ou o segundo) tem alguma vantagem?

## SISTEMATIZANDO AS JOGADAS

- ▶ Bob tentar sistematizar o conhecimento sobre o jogo e reflete sobre as possíveis variações de jogos que podem ser jogados com pilhas de até 10 fichas (10+10)
- ▶ Ele nota que é bastante complexo e decide, tomando uma abordagem mais reducionista, refletir sobre o jogo com 2 fichas em cada pilha (2+2)
- ▶ Ele nota rapidamente que o segundo jogador, no caso ele, sempre ganha e decide então, escrever a receita para vencer sempre o jogo:
  - ▶ Se Alice pegar uma ficha de cada pilha, ele pegará as 2 restantes e vencerá o jogo
  - ▶ Se Alice pegar apenas uma ficha de uma das pilhas, ele pegará apenas 1 ficha da mesma pilha. Assim, Alice só terá uma pilha para pegar apenas uma pedra, deixando a última para Bob que vencerá o jogo novamente

## SISTEMATIZANDO AS JOGADAS

- ▶ Bob presume que ele **vencerá todo jogo  $n+n$**
- ▶ Ele continua jogando e aplicando seu algoritmo para vitória e **perde**
- ▶ Ele começa a desenvolver uma estratégia para o jogo  $3+3$  e nota que há diversas estratégias possíveis ficando sem esperança de conseguir um algoritmo para o jogo  $10+10$  pois Alice pode tomar uma enorme variedade de possíveis jogadas e estratégias

## SISTEMATIZANDO AS JOGADAS

- ▶ Enquanto isso Alice, que já cursou a disciplina de Algoritmos, nota que sempre perde o jogo 2+2 e não desiste de encontrar uma receita para vencer jogos 3+3
- ▶ Ela nota que a forma como Bob definiu seu algoritmo de forma textual não ajudou muito pois há muitas possibilidades
- ▶ Ela tenta organizar suas idéias usando um outro tipo de modelo: o matricial
- ▶ Cada posição da  $(i, j)$  da matriz descreve uma possível jogada que ela fará em um jogo  $i+j$  ( $i$  pedras da pilha  $A$  e  $j$  na pilha  $B$ ). Ela começa a preencher a matriz com os símbolos:
  - ▶  $*$  : não vale a pena jogar pois sempre perderá se o oponente conhecer a regra da vitória
  - ▶  $\leftarrow$  : deve retirar uma ficha da pilha  $B$
  - ▶  $\uparrow$  : deve retirar uma ficha da pilha  $A$
  - ▶  $\nwarrow$  : deve retirar uma ficha de cada pilha

# SISTEMATIZANDO AS JOGADAS

▶ Veja a matriz que Alice gerou com esse algoritmo para vencer o jogo

	0	1	2	3	4	5	6	7	8	9	10
0	*	←	*	←	*	←	*	←	*	←	*
1	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
2	*	←	*	←	*	←	*	←	*	←	*
3	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
4	*	←	*	←	*	←	*	←	*	←	*
5	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
6	*	←	*	←	*	←	*	←	*	←	*
7	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
8	*	←	*	←	*	←	*	←	*	←	*
9	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
10	*	←	*	←	*	←	*	←	*	←	*

▶ Por exemplo, se ela enfrenta um jogo 3+3, ela encontra o símbolo “↖” dentro da célula (3, 3) indicando que ela deve pegar uma ficha de cada topo de pilha

# SISTEMATIZANDO AS JOGADAS

- ▶ Veja a matriz que Alice gerou com esse algoritmo para vencer o jogo

	0	1	2	3	4	5	6	7	8	9	10
0	*	←	*	←	*	←	*	←	*	←	*
1	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
2	*	←	*	←	*	←	*	←	*	←	*
3	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
4	*	←	*	←	*	←	*	←	*	←	*
5	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
6	*	←	*	←	*	←	*	←	*	←	*
7	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
8	*	←	*	←	*	←	*	←	*	←	*
9	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
10	*	←	*	←	*	←	*	←	*	←	*

- ▶ Isso faz com que Bob enfrente a seguir um jogo 2+2, marcado com “\*” indicando que, não importa a jogada que ele faça, Alice vencerá



# SISTEMATIZANDO AS JOGADAS

▶ Veja a matriz que Alice gerou com esse algoritmo para vencer o jogo

	0	1	2	3	4	5	6	7	8	9	10
0	*	←	*	←	*	←	*	←	*	←	*
1	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
2	*	←	*	←	*	←	*	←	*	←	*
3	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
4	*	←	*	←	*	←	*	←	*	←	*
5	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
6	*	←	*	←	*	←	*	←	*	←	*
7	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
8	*	←	*	←	*	←	*	←	*	←	*
9	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
10	*	←	*	←	*	←	*	←	*	←	*

▶ Suponha que, ao invés disso, Bob retire uma pedra da pilha *B*, deixando um jogo 2+1 para Alice que consultará a célula (2, 1) da tabela que lhe indicará que deve tirar uma outra ficha da pilha *B*, deixando apenas duas fichas na pilha *A* (Alice vencerá)

# SISTEMATIZANDO AS JOGADAS

▶ Veja a matriz que Alice gerou com esse algoritmo para vencer o jogo

	0	1	2	3	4	5	6	7	8	9	10
0	*	←	*	←	*	←	*	←	*	←	*
1	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
2	*	←	*	←	*	←	*	←	*	←	*
3	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
4	*	←	*	←	*	←	*	←	*	←	*
5	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
6	*	←	*	←	*	←	*	←	*	←	*
7	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
8	*	←	*	←	*	←	*	←	*	←	*
9	↑	↖	↑	↖	↑	↖	↑	↖	↑	↖	↑
10	*	←	*	←	*	←	*	←	*	←	*

▶ Se Bob, ao invés disso, retirar uma ficha da pilha *A*, Alice deverá consultar a célula (1, 2) que lhe indicará que deve retirar uma ficha da pilha *A* deixando duas fichas na pilha *B* para que vença novamente.

## SISTEMATIZANDO AS JOGADAS

- ▶ Bob ficou impressionado com a tabela construída por Alice mas não consegue construir uma matriz semelhante para um jogo  $20 \times 20$
- ▶ Bob não cursou a disciplina de Algoritmos
- ▶ Você deve estar se perguntando:
  - ▶ Mas o que esse jogo tem a ver comigo?
  - ▶ E o que tem a ver com alinhamento de sequências?
    - ▶ Esse jogo e o respectivo algoritmo nada mais são do que o clássico algoritmo para alinhamento de sequências

## ALINHAMENTO DE SEQUÊNCIAS

- ▶ Apesar de não parecer que as fichas tenham a ver com a sequência de DNA ou de aminoácidos, a idéia computacional usada para resolver ambos os problemas é a mesma
- ▶ Bob não é capaz de encontrar a estratégia para o jogo porque não entende como os algoritmos de alinhamento funcionam
- ▶ Mesmo que ele use o BLAST, ele não entende a idéia por trás do método e, como ele não conseguiu encontrar uma estratégia para o jogo de *10+10*, ele provavelmente falhará quando confrontado por um novo problema de alinhamento
- ▶ Bob precisa se tornar um bioinformata de verdade e aprender a **pensar algoritmicamente sobre os problemas biológicos**

**Paradigma** é um exemplo que serve como modelo; um padrão.

Dicionário do Google

- ▶ Em computação, **paradigmas** são formas ou técnicas de projeto de algoritmos comumente utilizadas, ou seja, para as quais há diversos exemplos de diferentes algoritmos.
- ▶ Alguns exemplos não tratados nesse curso incluem: indução, recursividade, algoritmos de tentativa e erro, divisão e conquista, balanceamento, algoritmos gulosos e algoritmos aproximados [Ziviani, 2004].

**Programação dinâmica** é um paradigma de projeto de algoritmos baseada no cálculo da solução de todos os possíveis subproblemas de um problema, partindo dos menores para os maiores, armazenando os resultados em uma matriz ou tabela.

Uma grande vantagem desse paradigma é que uma vez que um subproblema é resolvido e sua resposta é armazenada na matriz, a solução desse subproblema nunca mais será recalculada.

**Nívio Ziviani**

- ▶ Em computação, **paradigmas** são formas ou técnicas de projeto de algoritmos comumente utilizadas, ou seja, para as quais há diversos exemplos de diferentes algoritmos.
- ▶ Alguns exemplos não tratados nesse curso incluem: indução, recursividade, algoritmos de tentativa e erro, divisão e conquista, balanceamento, algoritmos gulosos e algoritmos aproximados [Ziviani, 2004].

## PROGRAMAÇÃO DINÂMICA

- ▶ Você percebe agora que o problema do jogo das fichas é resolvido usando programação dinâmica?
- ▶ Note que quando Bob quer resolver um problema  $20+20$ , ele necessariamente passará pela solução de um problema  $10+10$  e por um  $2+2$  respectivamente
- ▶ Uma vez que o menor subproblema é calculado, ele nunca será revisitado mas comporá as soluções dos problemas maiores
- ▶ Dizemos que um problema para poder ser resolvido através de programação dinâmica precisa ter subestrutura ótima, o que certamente é o caso no nosso jogo
- ▶ Interessantemente, um alinhamento par-a-par local de sequências (problema resolvido pelo BLAST), é um problema que tem subestrutura ótima
- ▶ Reflita sobre essa questão