# Proposta de Dissertação de Mestrado

## Uma Ferramenta para Avaliação de Algoritmos de Rearranjo de Genomas e sua Aplicação ao Problema da Ordenação por Transposições de Prefixo

Gustavo Rodrigues Galvão e Zanoni Dias

Instituto de Computação, UNICAMP

13 de Maio de 2011

# Agenda

# Rearranjo de Genomas

- Problema: determinar a distância evolutiva entre indíviduos.

# Rearranjo de Genomas

- Problema: determinar a distância evolutiva entre indíviduos.
- Solução: encontrar a menor sequência de eventos de rearranjo que levou o genoma de um indivíduo a se transformar no outro.

# Genomas e Eventos de Rearranjo

- Genomas, cromossomos e genes.

# Genomas e Eventos de Rearranjo

- Genomas, cromossomos e genes.
- Eventos de rearranjo conservativos e não conservativos.

# Genomas e Eventos de Rearranjo

- Genomas, cromossomos e genes.
- Eventos de rearranjo conservativos e não conservativos.
- Exemplos: *reversão*, *transposição*, *block-interchange*, *translocação*, *remoção*, *inserção* e *duplicação*.

# Genomas e Eventos de Rearranjo

- Genomas, cromossomos e genes.
- Eventos de rearranjo conservativos e não conservativos.
- Exemplos: *reversão*, *transposição*, *block-interchange*, *translocação*, *remoção*, *inserção* e *duplicação*.
- Este trabalho é focado em genomas unicromossais e em eventos conservativos.

# Genoma Mitocondrial de Artrópodes

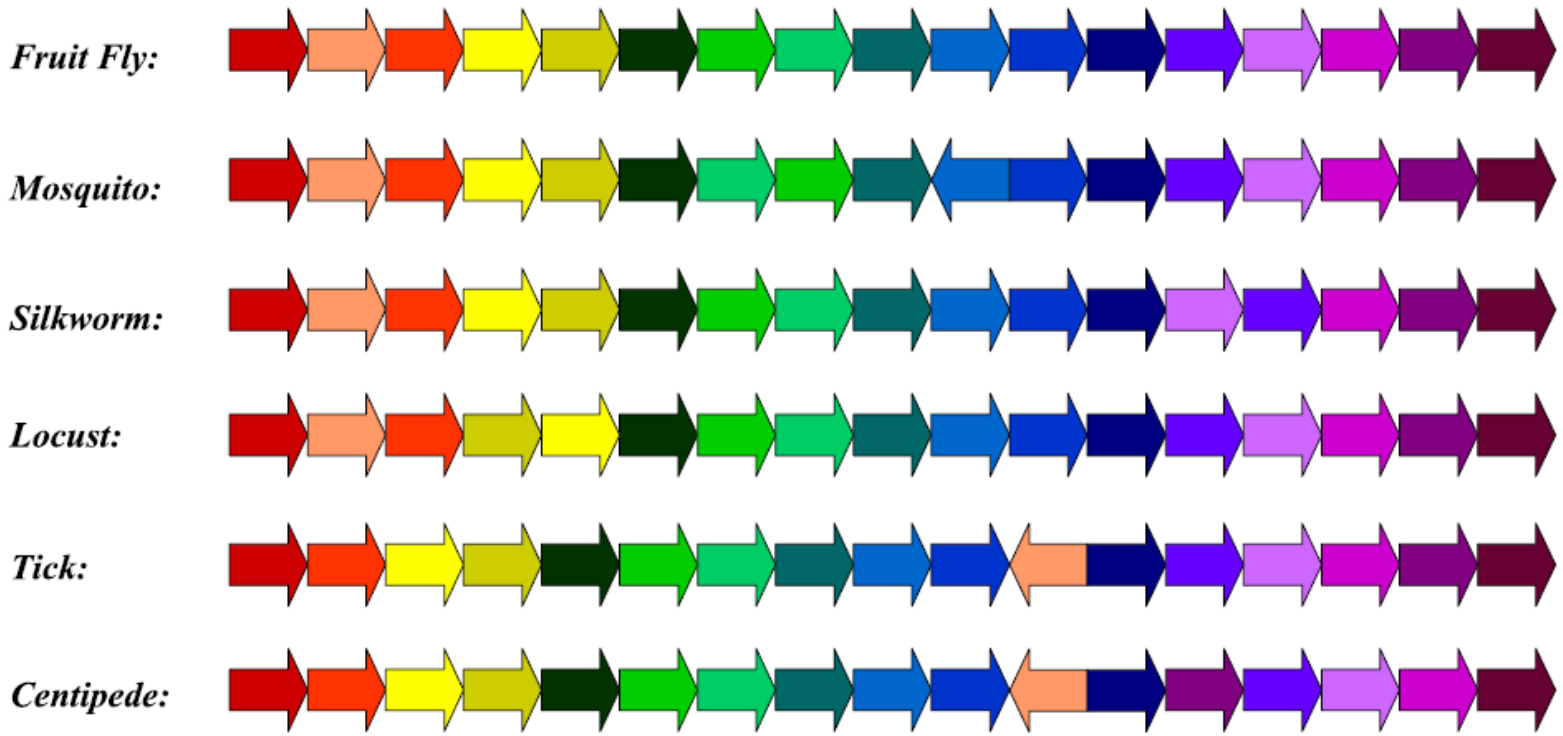



Figura: Adaptada de *Bergeron, A.; Stoye, J. (2003) On the similarity of sets of permutations and its applications to genome comparison (ISSN 0946-7831;Report 2003-01) Bielefeld: Universität Bielefeld.*

# Definição Formal de um Cromossomo

- Um cromossomo é classicamente representado como uma $n$-tupla.

# Definição Formal de um Cromossomo

- Um cromossomo é classicamente representado como uma $n$-tupla.
- Caso não haja repetição dos genes, esta $n$-tupla é uma permutação $\pi = (\pi_1\ \pi_2\ \ldots\ \pi_n)$, $1 \leq |\pi_i| \leq n$ e $|\pi_i| \neq |\pi_j| \leftrightarrow i \neq j$.

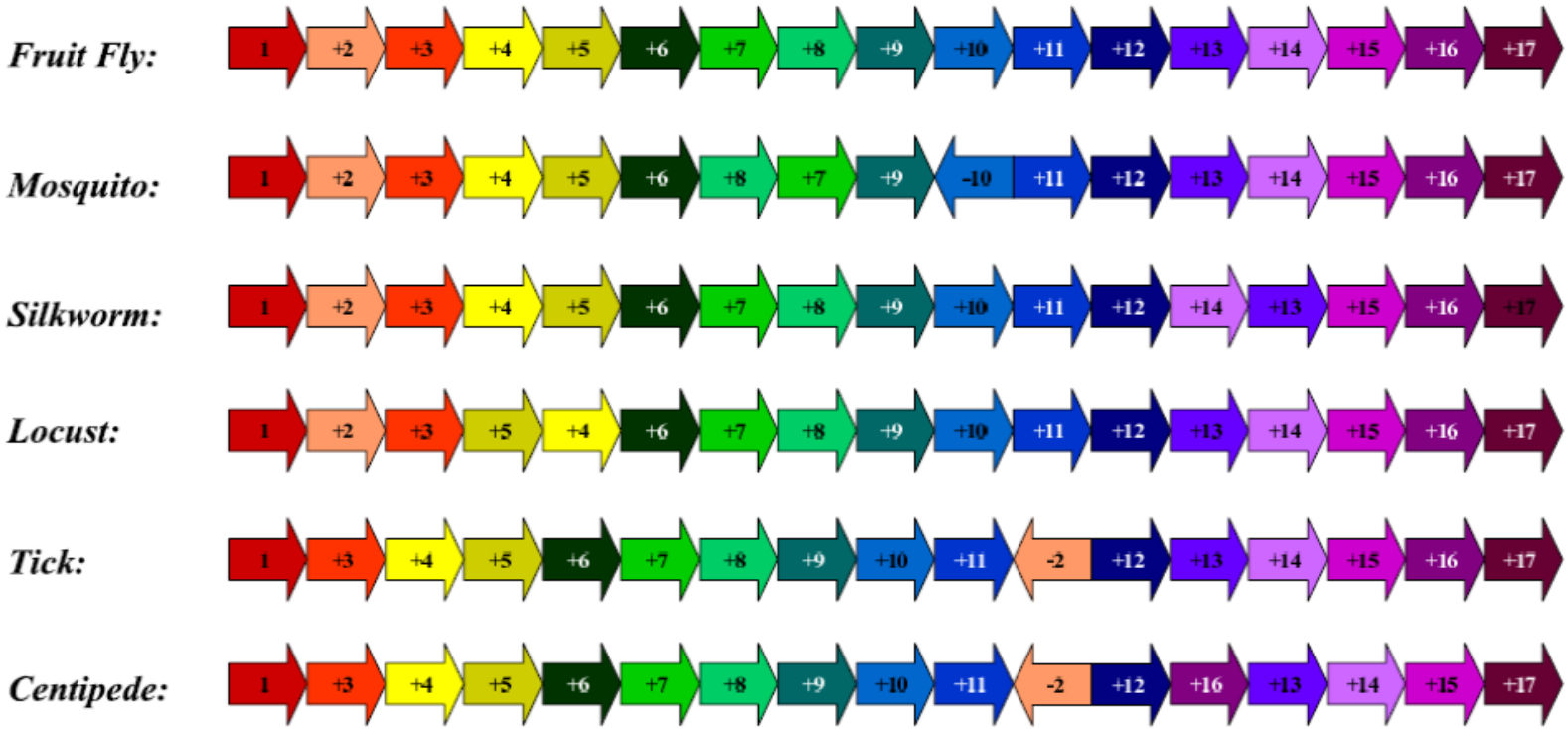
# Definição Formal de um Cromossomo

- Um cromossomo é classicamente representado como uma $n$-tupla.
- Caso não haja repetição dos genes, esta $n$-tupla é uma permutação $\pi = (\pi_1\ \pi_2\ \ldots\ \pi_n)$, $1 \leq |\pi_i| \leq n$ e $|\pi_i| \neq |\pi_j| \leftrightarrow i \neq j$.
- Cada elemento $\pi_i$ possui um sinal, $+$ ou $-$, que indica a orientação do gene que ele representa.

# Definição Formal de um Cromossomo

- Um cromossomo é classicamente representado como uma $n$-tupla.
- Caso não haja repetição dos genes, esta $n$-tupla é uma permutação $\pi = (\pi_1\ \pi_2\ \ldots\ \pi_n)$, $1 \leq |\pi_i| \leq n$ e $|\pi_i| \neq |\pi_j| \leftrightarrow i \neq j$.
- Cada elemento $\pi_i$ possui um sinal, $+$ ou $-$, que indica a orientação do gene que ele representa.
- Quando não há informação sobre a orientação dos genes, o sinal é omitido.

# Representação do Genoma Mitocondrial de Artrópodes



# Distância de Rearranjo

- Modelo de rearranjo: conjunto de eventos de rearranjo permitidos para transformar um genoma em outro.

# Distância de Rearranjo

- Modelo de rearranjo: conjunto de eventos de rearranjo permitidos para transformar um genoma em outro.
- Dadas duas permutações $\pi$ e $\sigma$ e um modelo de rearranjo $M$, podemos encontrar diversas sequências de eventos de rearranjos pertencentes a $M$ que transformam $\pi$ em $\sigma$.

# Distância de Rearranjo

- Modelo de rearranjo: conjunto de eventos de rearranjo permitidos para transformar um genoma em outro.
- Dadas duas permutações $\pi$ e $\sigma$ e um modelo de rearranjo $M$, podemos encontrar diversas sequências de eventos de rearranjos pertencentes a $M$ que transformam $\pi$ em $\sigma$.
- A distância de rearranjo entre as permutações $\pi$ e $\sigma$ com respeito ao modelo $M$ é igual ao tamanho da menor sequência.

# Distância de Rearranjo

- Modelo de rearranjo: conjunto de eventos de rearranjo permitidos para transformar um genoma em outro.
- Dadas duas permutações $\pi$ e $\sigma$ e um modelo de rearranjo $M$, podemos encontrar diversas sequências de eventos de rearranjos pertencentes a $M$ que transformam $\pi$ em $\sigma$.
- A distância de rearranjo entre as permutações $\pi$ e $\sigma$ com respeito ao modelo $M$ é igual ao tamanho da menor sequência.
- A maior distância de rearranjo entre duas permutações de tamanho $n$ com respeito ao modelo $M$ é chamada de diâmetro de rearranjo.

# Problema da Ordenação de Genomas

- Seja $I_n = (1\ 2\ \ldots\ n)$ a permutação identidade.

# Problema da Ordenação de Genomas

- Seja $I_n = (1\ 2\ \ldots\ n)$ a permutação identidade.
- Ordenar uma permutação $\alpha$ significa transformá-la na permutação $I_n$ considerando eventos de rearranjo pertencentes a um modelo $M$.

# Problema da Ordenação de Genomas

- Seja $I_n = (1\ 2\ \ldots\ n)$ a permutação identidade.
- Ordenar uma permutação $\alpha$ significa transformá-la na permutação $I_n$ considerando eventos de rearranjo pertencentes a um modelo $M$.
- Processo equivalente ao de transformar a permutação $\pi$ na permutação $\sigma$ se tomarmos $\alpha = \pi\sigma^{-1}$.

# Problema da Ordenação de Genomas

- Seja $I_n = (1\ 2\ \ldots\ n)$ a permutação identidade.
- Ordenar uma permutação $\alpha$ significa transformá-la na permutação $I_n$ considerando eventos de rearranjo pertencentes a um modelo $M$.
- Processo equivalente ao de transformar a permutação $\pi$ na permutação $\sigma$ se tomarmos $\alpha = \pi\sigma^{-1}$.
- Determinar a distância de rearranjo entre dois genomas reduz-se ao Problema da Ordenação de Genomas.

# Estado da Arte de Algumas Variações

| Modelo de Rearranjo | Orientação | Melhor Solução Teórica |
|---|---|---|
| Reversão | Sim | Algoritmo exato $O(n^{\frac{3}{2}}\sqrt{\log n})$ |
| Reversão | Não | Algoritmo 1.375-aproximado |
| Transposição | Não | Algoritmo 1.375-aproximado |
| Transposição e Transreversão | Sim | Algoritmo 1.5-aproximado |
| Reversão de Prefixo | Não | Algoritmo 2-aproximado |
| Transposição de Prefixo | Não | Algoritmo 2-aproximado |
| Reversão e Transposição | Sim | Algoritmo 2-aproximado |

# Estado da Arte de Algumas Variações

| Modelo de Rearranjo | Orientação | Melhor Solução Teórica |
|---|---|---|
| Reversão | Sim | Algoritmo exato $O(n^{\frac{3}{2}}\sqrt{\log n})$ |
| Reversão | Não | Algoritmo 1.375-aproximado |
| Transposição | Não | Algoritmo 1.375-aproximado |
| Transposição e Transreversão | Sim | Algoritmo 1.5-aproximado |
| Reversão de Prefixo | Não | Algoritmo 2-aproximado |
| Transposição de Prefixo | Não | Algoritmo 2-aproximado |
| Reversão e Transposição | Sim | Algoritmo 2-aproximado |

# Uma Ferramenta de Auditoria

- As melhores soluções para a maior parte das variações do Problema de Ordenação de Genomas são aproximações ou heurísticas.

# Uma Ferramenta de Auditoria

- As melhores soluções para a maior parte das variações do Problema de Ordenação de Genomas são aproximações ou heurísticas.
- Necessidade de criar mecanismos que auxiliem o processo de análise quantitativa de tais soluções.

# Uma Ferramenta de Auditoria

- As melhores soluções para a maior parte das variações do Problema de Ordenação de Genomas são aproximações ou heurísticas.
- Necessidade de criar mecanismos que auxiliem o processo de análise quantitativa de tais soluções.
- Construção de uma ferramenta para automatizar e padronizar a avaliação de algoritmos de rearranjo de genomas.

# Aplicando a Ferramenta

- Avaliação de algoritmos de aproximação e heurísticas que viermos a desenvolver para resolver o Problema da Ordenação por Transposições de Prefixo.

# Aplicando a Ferramenta

- Avaliação de algoritmos de aproximação e heurísticas que viermos a desenvolver para resolver o Problema da Ordenação por Transposições de Prefixo.
- Problema não muito explorado, por isso nosso interesse em estudá-lo.

# Ideia Geral

- Algoritmos aproximados e heurísticas não garantem que a distância calculada é a distância de rearranjo.

# Ideia Geral

- Algoritmos aproximados e heurísticas não garantem que a distância calculada é a distância de rearranjo.
- Obtenção de estatísticas comparando a distância retornada por um algoritmo de aproximação ou heurística e a distância de rearranjo para um grande conjunto de permutações.

# Ideia Geral

- Algoritmos aproximados e heurísticas não garantem que a distância calculada é a distância de rearranjo.
- Obtenção de estatísticas comparando a distância retornada por um algoritmo de aproximação ou heurística e a distância de rearranjo para um grande conjunto de permutações.
- Distâncias de rearranjo obtidas por meio de um algoritmo de busca em largura.

# Estatísticas Consideradas

- **Distância média**: média aritmética das distâncias retornadas pelo algoritmo.

# Estatísticas Consideradas

- **Distância média**: média aritmética das distâncias retornadas pelo algoritmo.
- **Diâmetro**: valor da maior distância retornada pelo algoritmo.

# Estatísticas Consideradas

- **Distância média**: média aritmética das distâncias retornadas pelo algoritmo.
- **Diâmetro**: valor da maior distância retornada pelo algoritmo.
- **Fator médio**: média aritmética dos "fatores de aproximação" calculados para cada genoma.

# Estatísticas Consideradas

- **Distância média**: média aritmética das distâncias retornadas pelo algoritmo.
- **Diâmetro**: valor da maior distância retornada pelo algoritmo.
- **Fator médio**: média aritmética dos "fatores de aproximação" calculados para cada genoma.
- **Fator máximo**: valor do maior "fator de aproximação" produzido pelo algoritmo.

# Estatísticas Consideradas

- **Distância média**: média aritmética das distâncias retornadas pelo algoritmo.
- **Diâmetro**: valor da maior distância retornada pelo algoritmo.
- **Fator médio**: média aritmética dos "fatores de aproximação" calculados para cada genoma.
- **Fator máximo**: valor do maior "fator de aproximação" produzido pelo algoritmo.
- **Igualdade**: porcentagem de genomas para os quais a distância retornada pelo algoritmo é igual a distância de rearranjo.

# Características

- Avaliação será feita para todas as permutações sem sinal de tamanho até 13 e todas as permutações com sinal de tamanho até 10.

# Características

- Avaliação será feita para todas as permutações sem sinal de tamanho até 13 e todas as permutações com sinal de tamanho até 10.
- A ferramenta será implementada sob uma arquitetura cliente-servidor.

# Características

- Avaliação será feita para todas as permutações sem sinal de tamanho até 13 e todas as permutações com sinal de tamanho até 10.
- A ferramenta será implementada sob uma arquitetura cliente-servidor.
- Os modelos de rearranjo cobertos pela ferramenta serão aqueles abordados pela literatura que se aplicam apenas a genomas unicrossomais.

# Definição Formal

## Transposição de Prefixo

*É definida como um evento* $\rho(j, k)$ *sobre uma permutação* $\pi = (\pi_1\ \pi_2\ \ldots\ \pi_n)$ *tal que* $\rho(j, k) \cdot (\underline{\pi_1\ \ldots\ \pi_{j-1}}\ \underline{\pi_j\ \ldots\ \pi_{k-1}}\ \pi_k\ \ldots\ \pi_n) = (\underline{\pi_j\ \ldots\ \pi_{k-1}}\ \underline{\pi_1\ \ldots\ \pi_{j-1}}\ \pi_k\ \ldots\ \pi_n)$, *sendo* $2 \leq j < k \leq n+1$.

# Principais Avanços

- Dias e Meidanis, em 2002, apresentaram dois algoritmos aproximativos, um com fator de aproximação 3 e outro com fator de aproximação 2, para resolver o problema e um limite inferior de $\frac{n}{2}$ e um limite superior de $n - 1$ para o diâmetro de transposição de prefixo, $D_{tp}(n)$.

# Principais Avanços

- Dias e Meidanis, em 2002, apresentaram dois algoritmos aproximativos, um com fator de aproximação 3 e outro com fator de aproximação 2, para resolver o problema e um limite inferior de $\frac{n}{2}$ e um limite superior de $n - 1$ para o diâmetro de transposição de prefixo, $D_{tp}(n)$.
- Fortuna e Meidanis, em 2005, apresentaram um algoritmo para ordenar $R_n$, $n \geq 4$, com $\lceil \frac{3n}{4} \rceil$ transposições de prefixo e estudaram uma família de permutações ditas fáceis.

# Principais Avanços

- Dias e Meidanis, em 2002, apresentaram dois algoritmos aproximativos, um com fator de aproximação 3 e outro com fator de aproximação 2, para resolver o problema e um limite inferior de $\frac{n}{2}$ e um limite superior de $n - 1$ para o diâmetro de transposição de prefixo, $D_{tp}(n)$.
- Fortuna e Meidanis, em 2005, apresentaram um algoritmo para ordenar $R_n$, $n \geq 4$, com $\lceil \frac{3n}{4} \rceil$ transposições de prefixo e estudaram uma família de permutações ditas fáceis.
- Chitturi e Sudborough, em 2008, demonstraram que $\frac{2n}{3} \leq D_{tp}(n) \leq n - \log_8 n$.

# Principais Avanços

- Dias e Meidanis, em 2002, apresentaram dois algoritmos aproximativos, um com fator de aproximação 3 e outro com fator de aproximação 2, para resolver o problema e um limite inferior de $\frac{n}{2}$ e um limite superior de $n-1$ para o diâmetro de transposição de prefixo, $D_{tp}(n)$.
- Fortuna e Meidanis, em 2005, apresentaram um algoritmo para ordenar $R_n$, $n \geq 4$, com $\lceil \frac{3n}{4} \rceil$ transposições de prefixo e estudaram uma família de permutações ditas fáceis.
- Chitturi e Sudborough, em 2008, demonstraram que $\frac{2n}{3} \leq D_{tp}(n) \leq n - \log_8 n$.
- Labarre, em 2008, demonstrou que $D_{tp}(n) \geq \lfloor \frac{3n+1}{4} \rfloor$.

# Principais Avanços

- Dias e Meidanis, em 2002, apresentaram dois algoritmos aproximativos, um com fator de aproximação 3 e outro com fator de aproximação 2, para resolver o problema e um limite inferior de $\frac{n}{2}$ e um limite superior de $n-1$ para o diâmetro de transposição de prefixo, $D_{tp}(n)$.
- Fortuna e Meidanis, em 2005, apresentaram um algoritmo para ordenar $R_n$, $n \geq 4$, com $\lceil \frac{3n}{4} \rceil$ transposições de prefixo e estudaram uma família de permutações ditas fáceis.
- Chitturi e Sudborough, em 2008, demonstraram que $\frac{2n}{3} \leq D_{tp}(n) \leq n - \log_8 n$.
- Labarre, em 2008, demonstrou que $D_{tp}(n) \geq \lfloor \frac{3n+1}{4} \rfloor$.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

1. Obtenção dos créditos obrigatórios e defesa do Exame de Qualificação do Mestrado (EQM).

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

2. Implementação do algoritmo de busca em largura para calcular as distâncias de rearranjo.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

3. Cálculo das distâncias de rearranjo relativas ao modelos cobertos pela ferramenta.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

4. Implementação da ferramenta.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

5. Estudo de novas famílias de permutações que podem ser ordenadas por transposições de prefixo em tempo polinomial.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

6. Obtenção de novos algoritmos aproximativos e heurísticas para o Problema de Ordenação por Transposições de Prefixo.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

7. Avaliação, utilizando a ferramenta, dos algoritmos produzidos.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

8. Escrita da dissertação.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

9. Revisão final do texto da dissertação.

# Agosto de 2010 a Julho de 2012

| | 2010 | | | | | 2011 | | | | | | | | | | | 2012 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | S | O | N | D | J | F | M | A | M | J | J | A | S | O | N | D | J | F | M | A | M | J | J |
| 1 | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 2 | ● | ● | ● | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | | | |
| 4 | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | | | | | |
| 5 | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | ● | ● | ● | ● | ● | ● | | | |
| 7 | | | | | | | | | | | | | | | | | | ● | ● | ● | ● | | | |
| 8 | | ● | ● | | | | | | | ● | ● | | | ● | ● | | | | | ● | ● | ● | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | ● | |
| 10 | | | | | | | | | | | | | | | | | | | | | | | | ● |

10. Defesa da dissertação.

# Cálculo das Distâncias de Rearranjo

- Algoritmo de busca em largura implementado de maneira a consumir a menor quantidade de memória possível.

# Cálculo das Distâncias de Rearranjo

- Algoritmo de busca em largura implementado de maneira a consumir a menor quantidade de memória possível.
- Resta calcular as distâncias de rearranjo das permutações sem sinal de tamanho 13 referentes ao modelo que cobre reversões e transposições.

# Cálculo das Distâncias de Rearranjo

- Algoritmo de busca em largura implementado de maneira a consumir a menor quantidade de memória possível.
- Resta calcular as distâncias de rearranjo das permutações sem sinal de tamanho 13 referentes ao modelo que cobre reversões e transposições.
- Resultados parciais foram publicados nos anais do *26th Symposium on Applied Computing* da ACM deste ano (SAC'2011).

# Implementação da Ferramenta

- A ferramenta foi implementada em Java. Utilizamos o *framework* Apache Axis2 para auxiliar na construção dos serviços web.

# Implementação da Ferramenta

- A ferramenta foi implementada em Java. Utilizamos o *framework* Apache Axis2 para auxiliar na construção dos serviços web.
- Construção de uma aplicação web para disponibilização da ferramenta e divulgação de resultados.

# Implementação da Ferramenta

- A ferramenta foi implementada em Java. Utilizamos o *framework* Apache Axis2 para auxiliar na construção dos serviços web.
- Construção de uma aplicação web para disponibilização da ferramenta e divulgação de resultados.
- Utilizamos o *framework* JSF para desenvolvê-la e o PostgreSQL como Sistema Gerenciador de Banco de Dados. Ela está hospedada em um servidor do Instituto de Computação da UNICAMP rodando o Apache Tomcat 6.0.

# Estratégias de Ataque

- Obtenção de novas famílias de permutações que podem ser ordenadas por transposições de prefixo em tempo polinomial.

# Estratégias de Ataque

- Obtenção de novas famílias de permutações que podem ser ordenadas por transposições de prefixo em tempo polinomial.

# Estratégias de Ataque

- Obtenção de novas famílias de permutações que podem ser ordenadas por transposições de prefixo em tempo polinomial.
- Melhorar o algoritmo 2-aproximado desenvolvido por Dias e Meidanis.

# Estratégias de Ataque

- Obtenção de novas famílias de permutações que podem ser ordenadas por transposições de prefixo em tempo polinomial.
- Melhorar o algoritmo 2-aproximado desenvolvido por Dias e Meidanis.
- Metodologia teórica similar à descrita na literatura para demonstrar resultados que viermos a obter.

# Análises

- Uma análise de complexidade será realizada para todos os algoritmos produzidos.

# Análises

- Uma análise de complexidade será realizada para todos os algoritmos produzidos.
- Quando for o caso, uma prova da aproximação será fornecida.

# Análises

- Uma análise de complexidade será realizada para todos os algoritmos produzidos.
- Quando for o caso, uma prova da aproximação será fornecida.
- Todos os algoritmos produzidos serão analisados quantitativamente com o auxílio da ferramenta.