**ESTRUTURA DE DADOS AVANÇADA**

**Configurações do computador usado para realizar as operações:**

**Fabricante/Modelo do Sistema:** Dell Inc. / Vostro 14-5480 **Sistema Operacional:** Linux Ubuntu 16.04  
**Processador:** Intel Core I7-5500U CPU @ 2.40GHz (4 CPUs), ~3.0GHz  
**Memória:** 8192MB RAM  
**Placa de Vídeo:** NVIDIA GeForce 830M

Abaixo segue a tabela de operações com o algoritmo de floresta de conjuntos disjuntos utilizando os métodos find\_set e link\_set da forma mais inteligente.

|  |  |  |  |
| --- | --- | --- | --- |
| **Arquivo** | **Soma das Arestas** | | **Tempo em ms** |
| tb8ch60\_0.txt | 2900 | | 7 |
| tb8ch60\_1.txt | 2793 | | 1 |
| tb8ch60\_2.txt | 2763 | | 4 |
| tb8ch60\_3.txt | 2895 | | 1 |
| tb8ch100\_0.txt | 3876 | | 2 |
| tb8ch100\_1.txt | 3611 | | 2 |
| tb8ch100\_2.txt | 3566 | | 3 |
| tb8ch100\_3.txt | 3649 | | 3 |
| tb8ch200\_0.txt | 5225 | | 10 |
| tb8ch200\_1.txt | 4956 | | 6 |
| tb8ch200\_2.txt | 4945 | | 7 |
| tb8ch200\_3.txt | 5201 | | 6 |
| tb8ch300\_0.txt | 6232 | | 17 |
| tb8ch300\_1.txt | 6179 | | 19 |
| tb8ch300\_2.txt | 6040 | | 10 |
| tb8ch300\_3.txt | 6247 | | 15 |
| tb8ch400\_0.txt | 7187 | | 22 |
| tb8ch400\_1.txt | 7152 | | 52 |
| tb8ch400\_2.txt | 7036 | | 41 |
| tb8ch400\_3.txt | 7069 | | 26 |
| tb8ch500\_0.txt | 7942 | | 53 |
| tb8ch500\_1.txt | 7977 | | 62 |
| tb8ch500\_2.txt | 7808 | | 63 |
| tb8ch500\_3.txt | 7716 | | 59 |
| tb8ch600\_0.txt | 8458 | | 85 |
| tb8ch600\_1.txt | 8790 | | 107 |
| tb8ch600\_2.txt | 8557 | | 110 |
| tb8ch600\_3.txt | 8452 | | 93 |
| tb8ch700\_0.txt | 9155 | | 161 |
| tb8ch700\_1.txt | 9397 | | 154 |
| tb8ch700\_2.txt | 9171 | | 153 |
| tb8ch700\_3.txt | 9114 | | 137 |
| tb8ch800\_0.txt | 9818 | | 289 |
| tb8ch800\_1.txt | 10057 | | 244 |
| tb8ch800\_2.txt | 9857 | | 222 |
| tb8ch800\_3.txt | 9672 | | 214 |
| tb8ch900\_0.txt | 10459 | | 360 |
| tb8ch900\_1.txt | 10521 | | 374 |
| tb8ch900\_2.txt | 10473 | | 337 |
| tb8ch900\_3.txt | 10138 | | 428 |
| **Média dos tempos:** | | 96,125 | |

Abaixo segue a tabela de operações com o algoritmo de floresta de conjuntos disjuntos usando os métodos find\_set1 e link\_set1 da forma menos utilizada, ou seja, a forma “mais burra”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Arquivo** | **Soma das Arestas** | | **Tempo em ms** |
| tb8ch60\_0.txt | 2900 | | 9 |
| tb8ch60\_1.txt | 2793 | | 1 |
| tb8ch60\_2.txt | 2763 | | 2 |
| tb8ch60\_3.txt | 2895 | | 2 |
| tb8ch100\_0.txt | 3876 | | 2 |
| tb8ch100\_1.txt | 3611 | | 3 |
| tb8ch100\_2.txt | 3566 | | 2 |
| tb8ch100\_3.txt | 3649 | | 2 |
| tb8ch200\_0.txt | 5225 | | 9 |
| tb8ch200\_1.txt | 4956 | | 6 |
| tb8ch200\_2.txt | 4945 | | 6 |
| tb8ch200\_3.txt | 5201 | | 5 |
| tb8ch300\_0.txt | 6232 | | 13 |
| tb8ch300\_1.txt | 6179 | | 13 |
| tb8ch300\_2.txt | 6040 | | 9 |
| tb8ch300\_3.txt | 6247 | | 14 |
| tb8ch400\_0.txt | 7187 | | 30 |
| tb8ch400\_1.txt | 7152 | | 38 |
| tb8ch400\_2.txt | 7036 | | 60 |
| tb8ch400\_3.txt | 7069 | | 68 |
| tb8ch500\_0.txt | 7942 | | 107 |
| tb8ch500\_1.txt | 7977 | | 101 |
| tb8ch500\_2.txt | 7808 | | 106 |
| tb8ch500\_3.txt | 7716 | | 107 |
| tb8ch600\_0.txt | 8458 | | 107 |
| tb8ch600\_1.txt | 8790 | | 109 |
| tb8ch600\_2.txt | 8557 | | 104 |
| tb8ch600\_3.txt | 8452 | | 92 |
| tb8ch700\_0.txt | 9155 | | 99 |
| tb8ch700\_1.txt | 9397 | | 119 |
| tb8ch700\_2.txt | 9171 | | 126 |
| tb8ch700\_3.txt | 9114 | | 122 |
| tb8ch800\_0.txt | 9818 | | 219 |
| tb8ch800\_1.txt | 10057 | | 212 |
| tb8ch800\_2.txt | 9857 | | 218 |
| tb8ch800\_3.txt | 9672 | | 216 |
| tb8ch900\_0.txt | 10459 | | 362 |
| tb8ch900\_1.txt | 10521 | | 379 |
| tb8ch900\_2.txt | 10473 | | 345 |
| tb8ch900\_3.txt | 10138 | | 421 |
| **Média dos tempos:** | | 98,185 | |

Abaixo segue a tabela de operações com o algoritmo de floresta de conjuntos disjuntos usando os métodos link\_set e find\_set1 (forma inteligente/forma burra).

|  |  |  |  |
| --- | --- | --- | --- |
| **Arquivo** | **Soma das Arestas** | | **Tempo em ms** |
| tb8ch60\_0.txt | 2900 | | 19 |
| tb8ch60\_1.txt | 2793 | | 2 |
| tb8ch60\_2.txt | 2763 | | 1 |
| tb8ch60\_3.txt | 2895 | | 1 |
| tb8ch100\_0.txt | 3876 | | 4 |
| tb8ch100\_1.txt | 3611 | | 11 |
| tb8ch100\_2.txt | 3566 | | 2 |
| tb8ch100\_3.txt | 3649 | | 1 |
| tb8ch200\_0.txt | 5225 | | 6 |
| tb8ch200\_1.txt | 4956 | | 7 |
| tb8ch200\_2.txt | 4945 | | 8 |
| tb8ch200\_3.txt | 5201 | | 8 |
| tb8ch300\_0.txt | 6232 | | 15 |
| tb8ch300\_1.txt | 6179 | | 18 |
| tb8ch300\_2.txt | 6040 | | 12 |
| tb8ch300\_3.txt | 6247 | | 15 |
| tb8ch400\_0.txt | 7187 | | 32 |
| tb8ch400\_1.txt | 7152 | | 27 |
| tb8ch400\_2.txt | 7036 | | 29 |
| tb8ch400\_3.txt | 7069 | | 68 |
| tb8ch500\_0.txt | 7942 | | 82 |
| tb8ch500\_1.txt | 7977 | | 51 |
| tb8ch500\_2.txt | 7808 | | 50 |
| tb8ch500\_3.txt | 7716 | | 49 |
| tb8ch600\_0.txt | 8458 | | 83 |
| tb8ch600\_1.txt | 8790 | | 94 |
| tb8ch600\_2.txt | 8557 | | 119 |
| tb8ch600\_3.txt | 8452 | | 95 |
| tb8ch700\_0.txt | 9155 | | 160 |
| tb8ch700\_1.txt | 9397 | | 149 |
| tb8ch700\_2.txt | 9171 | | 143 |
| tb8ch700\_3.txt | 9114 | | 126 |
| tb8ch800\_0.txt | 9818 | | 217 |
| tb8ch800\_1.txt | 10057 | | 209 |
| tb8ch800\_2.txt | 9857 | | 213 |
| tb8ch800\_3.txt | 9672 | | 211 |
| tb8ch900\_0.txt | 10459 | | 358 |
| tb8ch900\_1.txt | 10521 | | 374 |
| tb8ch900\_2.txt | 10473 | | 342 |
| tb8ch900\_3.txt | 10138 | | 434 |
| **Média dos tempos:** | | 96,975 | |

Abaixo segue a tabela de operações com o algoritmo de floresta de conjuntos disjuntos usando os métodos link\_set1 e find\_set (forma burra/forma inteligente).

|  |  |  |  |
| --- | --- | --- | --- |
| **Arquivo** | **Soma das Arestas** | | **Tempo em ms** |
| tb8ch60\_0.txt | 2900 | | 43 |
| tb8ch60\_1.txt | 2793 | | 2 |
| tb8ch60\_2.txt | 2763 | | 3 |
| tb8ch60\_3.txt | 2895 | | 2 |
| tb8ch100\_0.txt | 3876 | | 10 |
| tb8ch100\_1.txt | 3611 | | 7 |
| tb8ch100\_2.txt | 3566 | | 5 |
| tb8ch100\_3.txt | 3649 | | 7 |
| tb8ch200\_0.txt | 5225 | | 13 |
| tb8ch200\_1.txt | 4956 | | 9 |
| tb8ch200\_2.txt | 4945 | | 11 |
| tb8ch200\_3.txt | 5201 | | 10 |
| tb8ch300\_0.txt | 6232 | | 14 |
| tb8ch300\_1.txt | 6179 | | 11 |
| tb8ch300\_2.txt | 6040 | | 9 |
| tb8ch300\_3.txt | 6247 | | 14 |
| tb8ch400\_0.txt | 7187 | | 37 |
| tb8ch400\_1.txt | 7152 | | 34 |
| tb8ch400\_2.txt | 7036 | | 30 |
| tb8ch400\_3.txt | 7069 | | 35 |
| tb8ch500\_0.txt | 7942 | | 58 |
| tb8ch500\_1.txt | 7977 | | 97 |
| tb8ch500\_2.txt | 7808 | | 115 |
| tb8ch500\_3.txt | 7716 | | 59 |
| tb8ch600\_0.txt | 8458 | | 89 |
| tb8ch600\_1.txt | 8790 | | 94 |
| tb8ch600\_2.txt | 8557 | | 94 |
| tb8ch600\_3.txt | 8452 | | 80 |
| tb8ch700\_0.txt | 9155 | | 194 |
| tb8ch700\_1.txt | 9397 | | 174 |
| tb8ch700\_2.txt | 9171 | | 152 |
| tb8ch700\_3.txt | 9114 | | 145 |
| tb8ch800\_0.txt | 9818 | | 239 |
| tb8ch800\_1.txt | 10057 | | 234 |
| tb8ch800\_2.txt | 9857 | | 233 |
| tb8ch800\_3.txt | 9672 | | 309 |
| tb8ch900\_0.txt | 10459 | | 381 |
| tb8ch900\_1.txt | 10521 | | 473 |
| tb8ch900\_2.txt | 10473 | | 363 |
| tb8ch900\_3.txt | 10138 | | 481 |
| **Média dos tempos:** | | 144,775 | |

**Comentários:**

Analisando as tabelas acima, podemos concluir que o algoritmo mais utilizado a implementação mais inteligente em Floresta de Conjuntos Disjuntos é o mais rápido de todos; comparando com a implementação mais lenta, no caso em que usamos os métodos link\_set1 e find\_set (forma burra/forma inteligente), o tempo usando o algoritmo inteligente é cerca de 33% menor; já comparando a implementação inteligente e a implementação menos utilizada (ou “burra”), temos que o tempo usando o algoritmo inteligente é cerca de 2,5% menor; já comparando os métodos “híbridos” (link\_set e find\_set1 / link\_set1 e find\_set), o tempo para o algoritmo usando a implementação link\_set e find\_set1 é 34% menor. Também podemos concluir que usando a implementação link\_set e find\_set1 é a segunda mais rápida de todos.