**UNIVERSIDADE ESTADUAL DE SANTA CRUZ**

GUSTAVO BONFIM FRAGOSO

**PROJETO 3 B – COMPARAÇÃO DO TEMPO DE EXECUÇÃO DO ALGORITMO QUICKSORT NAS LINGUAGENS HASKELL, PROLOG E C.**

Ilhéus – BA

2016

**GUSTAVO BONFIM FRAGOSO**

**PROJETO 3 B – COMPARAÇÃO DO TEMPO DE EXECUÇÃO DO ALGORITMO QUICKSORT NAS LINGUAGENS HASKELL, PROLOG E C.**

Implementações em C, Haskell e prolog do algoritmo quicksort e comparação dos tempos de execução.

Prof.º César Bravo

Ilhéus – BA

2016

Sumário

[1. Compilação dos programas em linguagem C 1](#_Toc453170504)

[2. Execução 1](#_Toc453170505)

[3. Resultados 1](#_Toc453170506)

[3.1 Forma gráfica 1](#_Toc453170507)

[3.2 Forma analítica: 2](#_Toc453170508)

[4. Download 2](#_Toc453170509)

# Compilação dos programas em linguagem C

gcc -o permutation permutation.c

gcc -o quick quicksort.c

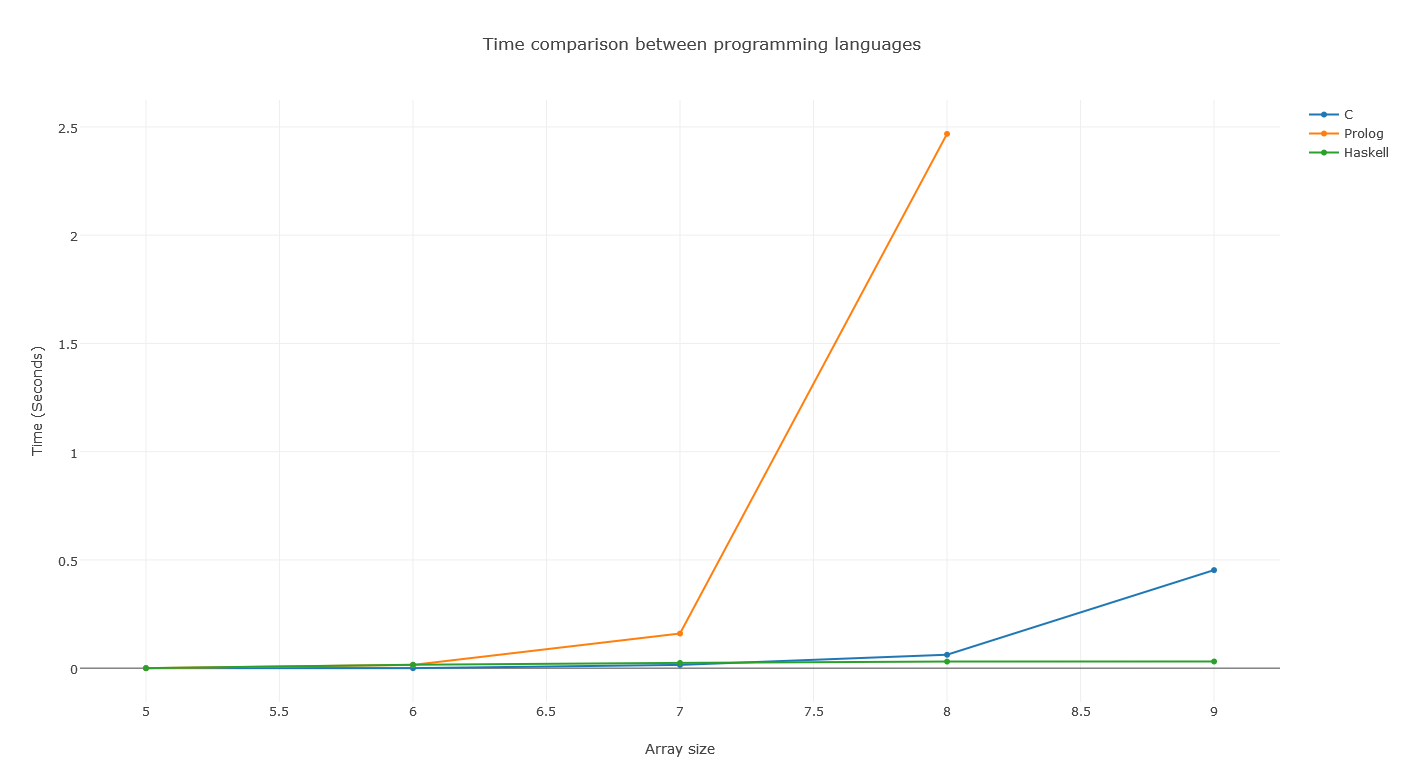
Obs.: Ver próxima seção antes de compilar

# Execução

1. Executar o programa permutation.c definindo o tamanho dos arrays que serão gerados na linha 4, exemplo: #define N 9 para tamanho 9;
2. Utilizar os programas:
   1. quicksort.c com ./quick (Alterando o arquivo a ser lido na linha 14)
   2. quicksortN.pl rodando a clausula “check”
   3. quicksort.hs colocando o nome do arquivo a ser lido na linha 14 e utilizando a função main;

# Resultados

## 3.1 Forma gráfica



## 3.2 Forma analítica:

Tempo de execução nas diferentes plataformas

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Linguagem | Quantidade de elementos por linha | | | | |
|  | 5 | 6 | 7 | 8 | 9 |
| C | 0,000000 s | 0,000000 s | 0,015000 s | 0,062000 s | 0,453000 s |
| Prolog | 0,000000 s | 0,015000 s | 0,160000 s | 2,468000 s | 30,70800 s \* |
| Haskell | 0,000000 s | 0,015701 s | 0,024324 s | 0,030638 s | 0,030791 s |

\*(Média de tempo para 3 amostras com 40.320 elementos multiplicado por 9)

# Download

https://github.com/gbfragoso/HaskellQuicksort