

Relatório EP3 – MAC0216 – Técnicas de Programação:**1. Especificações do hardware utilizado nos testes:**

- Sistema:

Kernel: 5.4.0-91-generic x86_64 bits: 64 Compilador: gcc v: 9.3.0

Desktop: Cinnamon 5.2.7 Distro: Linux Mint 20.3 Una

Base: Ubuntu 20.04 focal

- Máquina:

Laptop Dell Inspiron 5458

- CPU:

Intel Core i5-5200U, Dual Core

2. Tabela com o desempenho de cada função:

| Função | Média | Mínimo | Máximo |
|--------------------|-------|--------|--------|
| ehPar | 0.25 | 0.00 | 1.00 |
| ehPrimo | 0.15 | 0.00 | 1.00 |
| ehQuadradoPerfeito | 0.20 | 0.00 | 2.00 |
| ehCuboPerfeito | 0.30 | 0.00 | 2.00 |
| ehFibonacci | 0.20 | 0.00 | 1.00 |
| ehFatorial | 0.25 | 0.00 | 1.00 |
| criaVetorAleatorio | 55.60 | 43.00 | 76.00 |

Tabela 1: desempenho de cada função em termos de tempo médio, mínimo e máximo de execução em microssegundos usando a função `gettimeofday()`.

3. Conclusão:

Os tempos de execução para cada função fazem sentido quando comparados, pois as funções com o tempo de execução menor, por exemplo, a função `ehPrimo()` tem condição de saída se o número é dividido por outro além dele mesmo e por 1, portando, se logo nas primeiras passagens do *loop* o número já é divisível por algum que não os dois anteriores, a função logo retorna o valor 0. O mesmo ocorre para a função `ehPar()`, que somente precisa realizar a divisão do número de entrada por 2 e verificar o resto, e também para as funções restantes.

Ou seja, as funções da biblioteca `propriedadesnumericas.h` possuem condições de saídas do *loop* e retorno do valor 0 que são logo satisfeitas quando os números não comprem as características de serem primos, quadrados perfeitos, cubos perfeitos, fatoriais, etc.

Já na função de maior tempo de execução, a função *criaVetorAleatorio()*, da biblioteca *criavetoraleatorio.h*, os tempos observados são compreensíveis, pois esta função além de ter que criar um vetor com 10 números, precisa realizar diversos cálculos matemáticos que estão definidos na função *rand()*, utilizada dentro da função inicial, para a geração de cada um destes números pseudoaleatórios.