

LABORATORIO 2 - Performance de Algoritmos para Raiz Quadrada

Para fazer esse laboratório eu elaborei um código em C que importava os códigos Assembly compilados e executava as 5 formas de calcular a raiz quadrada. No começo tentei executar os laços FOR com 1 milhão de repetições, mas o código não executa, então tive que baixar para 100 mil repetições.

Após fazer isso obtive os seguintes resultados:

Número de Clocks por Execução e Implementação					
Execução	Raiz quadrada ASM 1	Raiz quadrada ASM 2	Raiz quadrada C	Função sqrt()	Raiz quadrada FPU
1	5	5	19	2	0
2	4	5	18	3	0
3	12	5	18	3	0
4	4	5	19	3	0
5	3	6	18	2	0
6	5	4	18	3	0
7	10	5	18	3	0
8	4	4	18	1	1
9	4	5	20	0	0
10	5	5	20	3	0
11	5	5	19	0	0
12	6	5	18	2	0
13	5	5	18	3	0

Com isso obtive as seguintes médias:

	Raiz quadrada ASM 1	Raiz quadrada ASM 2	Raiz quadrada C	Função sqrt()	Raiz quadrada FPU
Média	5,5385	4,9231	18,5385	2,1538	0,0769

Como podemos concluir ao ver o número de clocks utilizados por cada método, o cálculo utilizando a unidade de ponto flutuante do processador foi a mais rápida, muitas vezes nem sendo medida pelo código em C.

Logo depois veio a implementação SQRT da math.h, o que faz sentido já que esse comando possui várias otimizações.

Depois vieram os métodos em assembly e por fim, bem mais lento veio a implementação própria em C.