

2º Exercício

1º) Considerando o projeto desenvolvido, comente uma funcionalidade importante e mostre sua implementação de maneira detalhada. (2,0 pontos)

2º) Considerando a arquitetura x86, fale sobre:

- a) Os registradores de ponto flutuantes (formatos, modos de acesso). (0.75 ponto)
- b) O registrador de controle e de *flags* relativos ao processamento de ponto flutuante. (0.5 ponto)
- c) Explique os três modos de endereçamento das instruções de ponto flutuante. (0.75 ponto)
- d) Avalie o código abaixo, mostrando, linha-a-linha, os valores dos registradores e apresente o resultado. (1.5 pontos)

<pre>[SECTION .text] START: fld DWORD[m1] fld st0 fmulp st1,st0 fld DWORD[m2] fld st0 fmulp st1,st0 faddp st1,st0 fsqrt fst DWORD[m3]</pre>	<pre>mov eax, 1H int 80H [SECTION .data] m1 dd 1.0 m2 dd 2.0 m3 dd 0.0</pre>
---	---

3º) Considerando o código abaixo, implemente a função *obtain_result* (cuja implementação C é apresentada) em *assembly*, considerando a sintaxe NASM e ambiente operacional Linux. Explique o comportamento da pilha (ilustre com figuras), o processo de passagem dos parâmetros, da devolução do resultado da avaliação etc. (3,0 pontos)

```
#include <stdio.h>

int obtain_result(int a, int b, int (*compare)(int a, int b), char *s1)
{
    write(1,s1,13);
    return(compare(a, b));
}

int max(int a, int b)
{
    printf("Em max\n");
    return((a > b) ? a: b);
}
```