# Linguagem de montagem 6

**Desenvolvida por**: Raul A. Gonzalez Augusto, RA.: 211023698

# 1 Somar três variáveis inteiras

#include <stdio.h>

int main()

{

short int x = 10, y = 20, z = 30;

short int soma;

asm(" MOV CX, %[x]\n"

" ADD CX, %[y]\n"

" ADD CX, %[z]\n"

" MOV %[soma], CX"

: [soma] "=r"(soma)//saidas

: [x] "r"(x), [y] "r"(y), [z] "r"(z)//entradas

: "cx");

printf("%d\n", soma);

return 0;

}

# 2 Identificar se um número inteiro (com sinal) é negativo e se for, transforme-o em positivo

#include <stdio.h>

int main()

{

int n=-10, newN;

printf("Numero antes da checagem: %d\n",n);

asm("MOV EAX, 0 \n"

" CMP EAX, %[n] \n"

" MOV EAX, %[n] \n"

" JLE final \n"

" NEG EAX \n"

" final: \n"

" MOV %[ newN],EAX \n"

: [newN] "=r"( newN)//saidas

: [newN] "r"( newN)//entradas

: "eax");

printf("Numero apos a checagem: %d\n", newN);

return 0;

}

# 3 três variáveis inteiras (com sinal), contar quantas são maiores que 10

#include <stdio.h>

int main() {

int x = -5, y = 10, z = 30;

int count;

asm(" MOV EDX, 0 \n"

" MOV EAX, 10 \n"

" CMP EAX, %[x] \n"

" JGE testaY \n"

" INC EDX \n"

" testaY: \n"

" CMP EAX, %[y] \n"

" JGE testaZ \n"

" INC EDX \n"

" testaZ: \n"

" CMP EAX, %[z] \n"

" JGE fim \n"

" INC EDX \n"

" fim: \n"

" MOV %[count], EDX"

: [count] "=r"(count)//saida

: [x] "r"(x), [y] "r"(y), [z] "r"(z)//entradas

: "edx","eax");

printf("%d\n", count);

return 0;

}

# 4 Dado três variáveis inteiras (com sinal), somar as varieveis maiores que 10

#include <stdio.h>

int main() {

int x = -5, y = 10, z = 30;

int soma;

asm(" MOV EDX, 0 \n"

" MOV EAX, 10 \n"

" CMP EAX, %[x] \n"

" JGE testaY \n"

" ADD EDX,%[x] \n"

" testaY: \n"

" CMP EAX, %[y] \n"

" JGE testaZ \n"

" ADD EDX,%[y] \n"

" testaZ: \n"

" CMP EAX, %[z] \n"

" JGE fim \n"

" ADD EDX,%[z] \n"

" fim: \n"

" MOV %[soma], EDX"

: [soma] "=r"(soma)//saida

: [x] "r"(x), [y] "r"(y), [z] "r"(z)//entradas

: "edx","eax");

printf("%d\n", soma);

return 0;