

# ATIVIDADE 003: Códigos em C, C++ e Assembly

---

## Paradigmas de Linguagens de Programação



- Anna Alicia Milani
- Caio Andrade Mota
- Fabrizio Honda
- José Jordan Alcantara

# DESCRIÇÃO DA ATIVIDADE

- 1) Faça um programa que receba dois números inteiros e gere os números inteiros que estão no intervalo compreendido por eles.
- 2) Faça um Programa que peça dois números e imprima o maior deles.
- 3) Faça um Programa que peça a temperatura em graus Fahrenheit, transforme e mostre a temperatura em graus Celsius.  
$$C = 5 * ((F - 32) / 9).$$
- 4) Faça um programa que resolva a MÉDIA DA UEA

# IMPLEMENTAÇÃO: QUESTÃO 1

```
#include<stdio.h>

int main(){
    int n1, n2;
    printf("Insira o numero 1: ");
    scanf("%d", &n1);
    printf("Insira o numero 2: ");
    scanf("%d", &n2);
    for(int i = (n1+1); i < n2; i++){
        printf("%d ", i);
    }
    printf("\n");
    return 0;
}
```

```
#include<bits/stdc++.h>

using namespace std;

int main(){
    int n1, n2;
    cout << "Insira o numero 1: ";
    cin >> n1;
    cout << "Insira o numero 2: ";
    cin >> n2;
    for(int i = (n1+1); i < n2; i++){
        cout << i << " ";
    }
    cout << endl;
    return 0;
}
```

```
;nasm -felf64 intervalo_num.asm 66 gcc -o intervalo_num intervalo_num.o -no-pie 66 ./intervalo_num
section .data
    msg1: db "Insira o primeiro numero: ", 0
    msg1len: equ $-msg1
    msg2: db "Insira o segundo numero: ", 0
    msg2len: equ $-msg2
    msg3: db "%d", 10, 0
    formatin: db "%d", 0
    msgfinalin: db "fim do programa", 0
    msgfinalllen: equ $-msgfinal
    n1: times 4 db 0;; numeros de 32 bits
    n2: times 4 db 0

section .text
    extern printf
    extern scanf
    global main

    main:

        push rbx

        ; solicita numero 1
        mov eax, 4
        mov ebx, 1
        mov ecx, msg1
        mov edx, msg1len
        int 80h

        ; le numero 1
        mov rdi, formatin
        mov rsi, n1
        mov al, 0
        call scanf

        ; msg leitura num 2
        mov eax, 4
        mov ebx, 1
        mov ecx, msg2
        mov edx, msg2len
        int 80h

        ; le numero 2
        mov rdi, formatin
        mov rsi, n2
        mov al, 0
        call scanf

        mov eax, [n1]
        mov ebx, [n2]
        cmp eax, ebx
        jge fim
        mov ebx, [n1]
        jmp laco

    laco:
        inc ebx
        mov rdi, msg3
        mov esi, ebx
        xor eax, eax
        call printf
        cmp ebx, [n2]
        jge fim
        jmp laco

    fim:
        mov eax, 4
        mov ebx, 1
        mov ecx, msgfinal
        mov edx, msgfinalllen
        int 80h
```

# IMPLEMENTAÇÃO: QUESTÃO 2

```
#include<iostream>
#include<algorithm>

using namespace std;

int main(){

    int a, b;
    cout << "Insira o primeiro número: ";
    cin >> a;
    cout << "Insira o segundo número: ";
    cin >> b;

    int result = (a>b) ? a : b;
    cout << "O maior número é: " << result << endl;
    return 0;
}
```

```
#include <stdio.h>

int main(){
    int a, b;

    printf("Insira o primeiro número: ");
    scanf("%d", &a);
    printf("Insira o segundo número: ");
    scanf("%d", &b);

    int result = (a > b) ? a : b;
    printf("O maior número é: %d\n", result);

    return 0;
}
```

```
.intel_syntax noprefix
.text
.section .rodata.str1.1,"aMS",@progbits,1
.LC0:
.string "Insira o primeiro numero: "
.LC1:
.string "%d"
.LC2:
.string "Insira o segundo numero: "
.LC3:
.string "O maior numero e: %d\n"
.section .text.startup,"ax",@progbits
.p2align 4
.globl main
.type main, @function
main:
.LFB23:
.cfi_startproc
endbr64
sub rsp, 24
.cfi_def_cfa_offset 32
lea rsi, .LC0[rip]
mov edi, 1
xor eax, eax
call __printf_chk@PLT
lea rsi, 8[rsp]
lea rdi, .LC1[rip]
xor eax, eax
call __isoc99_scanf@PLT
lea rsi, .LC2[rip]
mov edi, 1
xor eax, eax
call __printf_chk@PLT
lea rsi, 12[rsp]
lea rdi, .LC1[rip]
xor eax, eax
call __isoc99_scanf@PLT
mov edx, DWORD PTR 8[rsp]
cmp DWORD PTR 12[rsp], edx
lea rsi, .LC3[rip]
cmovge edx, DWORD PTR 12[rsp]
mov edi, 1
xor eax, eax
call __printf_chk@PLT
xor eax, eax
add rsp, 24
.cfi_def_cfa_offset 8
ret
.cfi_endproc

1:
.align 8
.long 0xc0000002
.long 3f - 2f

2:
.long 0x3

3:
.align 8

4:
```

# IMPLEMENTAÇÃO: QUESTÃO 3

```
#include <bits/stdc++.h>
using namespace std;

int main()
{
    int f, c;

    cout << "Temperatura em graus Fahrenheit:"
          << "\n";
    cin >> f;

    c = 5 * ((f - 32) / 9);

    cout << "A temperatura " << f
          << " em graus Fahrenheit representa "
          << c << " em graus Celsius.";

    return 0;
}
```

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int f, c;

    printf("Insira temperatura em graus Fahrenheit: \n");
    scanf("%d", &f);

    c = 5 * ((f - 32) / 9);

    printf("A temperatura %d em graus Fahrenheit ",
           "representa %d em graus Celsius.", f, c);
    return 0;
}
```

```
name "celsi"

org 100h

jmp start

tc db 10    ; t celsius.
tf db 0     ; t fahrenheit.

result1 db ? ; result in fahrenheit.
result2 db ? ; result in celsius.

start:

; convert celsius to fahrenheit

mov cl, tc
mov al, 9
imul cl
mov cl, 5
idiv cl
add al, 32
mov result1, al

mov bl, result1
call print ; print bl

ret ; return to the operating system.

; procedure prints the binary value of bl
print proc near

pusha

; print result in binary:
mov cx, 8
p1: mov ah, 2 ; print function.
    mov dl, '0'
    test bl, 10000000b ; test first bit.
    jz zero
    mov dl, '1'
zero: int 21h
    shl bl, 1

loop p1
; print binary suffix:
mov dl, 'b'
int 21h
; print carriage return and new line:
mov dl, 0Dh
int 21h
mov dl, 0Ah
int 21h

popa

ret ; return to the main program.
print endp
```



# IMPLEMENTAÇÃO: QUESTÃO 4

```
#include<stdio.h>
#include<stdlib.h>
#include<locale.h>

int main(){

    setlocale(LC_ALL,"Portuguese");

    int p1, p2, pf, med, final=0;

    printf("Informe suas notas de P1 e P2: ");
    scanf("%i %i", &p1, &p2);
    med = (p1+p2)/2;

    if(med >= 8)
        printf("Aprovado com média %i\n", med );
    else if(med >= 4){
        printf("Você está na PF! Sua média é %i\n", med);
        final = 1;
    } else if(med < 4)
        printf("Reprovado sem PF com média %i\n", med);

    if(final){
        printf("Você precisa de %i para passar\n", 18 - p1 - p2 );
        printf("Digite sua nota da PF\n");
        scanf("%i",&pf);
        med = (p1+p2+pf)/3;
    }

    if(med >= 6)
        printf("Aprovado com PF! Sua média é %i\n", med);
    else
        printf("Reprovado sem PF com média %i\n ");

}
```

```
#include<bits/stdc++.h>
using namespace std;

int main(){

    setlocale(LC_ALL,"Portuguese");

    int p1, p2, pf, med;
    bool final = false;

    cout << "Informe suas notas de P1 e P2: ";
    cin >> p1 >> p2;
    med = (p1+p2)/2;

    if(med >= 8)
        cout << "Aprovado com média " << med << endl;
    else if(med >= 4){
        cout << "Você está na PF! Sua média é " << med << endl;
        final = true;
    }
    else if(med < 4)
        cout << "Reprovado sem PF com média " << med << endl;

    if(final){
        cout << "Você precisa de " << 18 - p1 - p2 << " para passar" << endl;
        cout << "Digite sua nota da PF" << endl;
        cin >> pf;
        med = (p1+p2+pf)/3;
    }

    if(med >= 6)
        cout << "Aprovado com PF! Sua média é " << med << endl;
    else
        cout << "Reprovado sem PF com média " << med << endl;

}
```

```
global main
extern printf

section .text
main:
    mov rdi, informarnotas
    call printf

    mov ah, 01h
    int 21h
    mov [p1], al
    mov ah, 01h
    int 21h
    mov [p2], al

    mov dx, [p1]
    add dx, [p2]
    mov [soma], dx
    mov al, [soma]
    movsd xmm1, [dois]
    movsd xmm0, [soma]
    divsd xmm0, xmm1
    movsd [med], xmm0

    mov rax, [oito]
    cmp [med], rax
    jge cond2
    mov rdi, aprovadosempf
    call printf
    jmp fim

cond2:
    mov rax, [quatro]
    cmp [med], rax
    jge cond3
    mov rdi, valfazerpf
    call printf
    mov rax, [um]
    mov [final], rax
    jmp provafinal

cond3:
    mov rdi, reprovadosempf
    jmp fim

provafinal:
    mov rax, [um]
    cmp [final], rax
    je fim
    mov rdi, informarfina
    call printf

    mov ah, 01h
    int 21h
    mov [pf], al

    addsd xmm0, [pf]
    movsd [med], xmm0
    movsd xmm1, [tres]
    movsd xmm0, [med]
    divsd xmm1, xmm0
    movsd [med], xmm0

    mov rax, [seis]
    cmp [final], rax
    jge reprov
    mov rdi, aprovadocompf
    jmp fim

reprov:
    mov rdi, reprovadocompf
    call printf
    jmp fim

fim:
    ret

section .data:
informarnotas: db "Informe suas notas da P1 e P2: ", 0
aprovadosempf: db "Aprovado sem pf, parabéns!", 0
valfazerpf: db "Você está de PF!", 0
reprovadosempf: db "Você não alcançou a média para a prova final.", 0
informarfina: db "Informe a nota que você tirou na PF: ", 0
aprovadocompf: db "Parabéns, você foi aprovado com PF!", 0
reprovadocompf: db "Você não alcançou a média com a PF para passar", 0
p1: dq 0.0
p2: dq 0.0
pf: dq 0.0
final: dq 0.0
soma: dq 0.0
med: dq 0.0
um: dq 1.0
dois: dq 2.0
tres: dq 3.0
quatro: dq 4.0
seis: dq 6.0
oito: dq 8.0
```

# OBRIGADO PELA ATENÇÃO!

- Link do GitHub:  
<https://github.com/caioandrademota/estcmp005.git>

