

AOC-1: Trabalho Prático 9

► Instruções

- Use apenas instruções vistas em aula até agora
- No MARS, use a seguinte configuração:
 - No menu *Settings*, HABILITE a opção *Permit extended (pseudo) instructions and formats*
 - No menu *Settings*, HABILITE a opção *Delayed Branching*
- Seus exercícios serão corrigidos com o MARS configurado da forma descrita acima
- Comente seu código
- Todos os exercícios são individuais
 - Cópias detectadas resultarão em nota zero para ambos os alunos
- Submeta os exercícios no AVA em um arquivo compactado até as 23:55 de **quinta-feira**, 12/11/2015. Não serão aceitos exercícios após a data/hora-limite.

AOC-1: Trabalho Prático 9

► Instruções

1. Escreva um programa equivalente ao código C ao lado em assembly.

Assim como no exemplo em C, utilize duas subrotinas (*soma3n* e *soma*).

```
main()
{
    int X=150;
    int Y=230;
    int Z=991;
    int R=0;
    R = soma3n(X, Y, Z);
}

int soma3n(int n1, int n2, int n3)
{
    return( soma(n3, soma(n1, n2)));
}

int soma(int A, int B)
{
    return( A+B );
}
```

AOC-1: Trabalho Prático 9

► Instruções

2. **Escreva um programa que calcule o fatorial de um número N inteiro sem sinal. Deve utilizar uma subrotina recursiva.**

O argumento para a subrotina deve ser passado através do registrador \$a0 e o resultado da subrotina deve ser retornado através do registrador \$v0.

AOC-1: Trabalho Prático 9

► Instruções

3. Implemente uma subrotina recursiva para imprimir na tela todas as permutações de uma *string*.

Para teste, escreva um programa que leia a *string* de entrada do teclado.

Como ponto de partida, utilize o código em C disponível no seguinte link:

<http://www.ime.usp.br/~pf/mac0122-2003/aulas/permut.html>