

AOC-1: Trabalho Prático 4

► Instruções

- Use apenas instruções vistas em aula até agora (slides)
- No MARS, use a seguinte configuração:
 - No menu *Settings*, desabilite as opções:
 - *Permit extended (pseudo) instructions and formats* e
 - *Delayed Branching*
- Seus exercícios serão corrigidos com o MARS configurado da forma descrita acima
- Utilize **EXATAMENTE** os registradores explicitados nos exercícios
 - Resultados armazenados em registradores diferentes serão considerados incorretos
- Comente seu código
- Todos os exercícios são individuais
 - Cópias detectadas resultarão em nota zero para ambos os alunos

AOC-1: Trabalho Prático 4

► Instruções

- O material deve ser entregue pelo AVA (<http://ava.ufpel.edu.br>) e deverá obedecer às seguintes regras:
 - Será um arquivo compactado (**obrigatoriamente** no formato **.zip**) contendo os códigos fonte dos TPs
 - Ex: *fulano_da_silva.zip*
 - Cada exercício deve ter o seguinte nome:
 - *matricula_tp{n}_e{m}.asm*
 - Onde:
 - *matricula* é a matrícula do aluno e
 - *{n}* é o número do TP
 - *{m}* é o número do exercício
 - Ex: *16100001_tp1_e1.asm*, *16100001_tp1_e2.asm*, ...
- Trabalhos que não seguirem as regras acima **NÃO SERÃO CORRIGIDOS!**
- O prazo de submissão do trabalho é até as **23:55** da próxima **sexta-feira**.
NÃO serão aceitos exercícios após a data/hora-limite

AOC-1: Trabalho Prático 4

► Instruções

1. Faça um programa que calcule a seguinte equação:

$$y = 32ab - 3a + 7b - 13$$

Utilize endereços de memória para armazenar o valor de *a*, *b* e o resultado *y*. Cada valor deve ocupar 4 bytes na memória (**.word**), assim como para o resultado (**.space 4**). Utilize as três primeiras posições da memória **.data** para armazenar, consecutivamente, *a*, *b* e *y*, iniciando o código com:

```
.data  
a: .word 3  
b: .word 5  
y: .space 4
```

AOC-1: Trabalho Prático 4

► Instruções

2. Faça um programa que calcule o seguinte polinômio usando o método de Horner:

$$y = 9a^3 - 5a^2 + 7a + 15$$

Utilize endereços de memória para armazenar o valor de a e o resultado y . Cada valor deve ocupar 4 bytes na memória (**.word**), assim como para o resultado (**.space 4**). Utilize as duas primeiras posições da memória **.data** para armazenar, consecutivamente, a e y , iniciando o código com:

```
.data  
a: .word 3  
y: .space 4
```

Observe como o método de Horner é mais eficiente (faz menos operações) que calcular o polinômio de forma sequencial.

AOC-1: Trabalho Prático 4

► Instruções

3. Faça um programa que calcule o seguinte polinômio usando o método de Horner:

$$y = -ax^4 + bx^3 - cx^2 + dx - e$$

Utilize endereços de memória para armazenar o valor de *a*, *b*, *c*, *d*, *e*, *x* e o resultado *y*. Cada valor deve ocupar 4 bytes na memória (**.word**), assim como para o resultado (**.space 4**). Utilize as sete primeiras posições da memória **.data** para armazenar, consecutivamente, *a*, *b*, *c*, *d*, *e*, *x* e *y*, iniciando o código com:

```
.data
a: .word -3
b: .word 7
c: .word 5
d: .word -2
e: .word 8
x: .word 4
y: .space 4
```