



Trabalho 06

– Procedimentos Recursivos, serviços do sistema, índices e ponteiros –

Trabalho individual. Resolva os problemas seguintes e entregue até o dia 26/10/2016 09/11/2016.

1. Escreva um procedimento não recursivo, em linguagem de montagem, para calcular os números de Fibonacci (ou série de Fibonacci), de 0 até n . O valor de n é entrado pelo usuário do programa. Calcule F_8 . A sequência de Fibonacci é definida pela seguinte relação de recorrência:

$$F_n = F_{n-1} + F_{n-2}, \text{ com } F_0 = 0 \text{ e } F_1 = 1$$

2. Repita o problema anterior escrevendo um procedimento recursivo.
3. Escreva um procedimento que gere 10 números aleatórios entre 0 e 100. O procedimento deverá salvar os números em um arquivo. Dica: use os serviços do sistema.
4. Escreva um programa, em linguagem de montagem para o MIPS, para ordenar um vetor de inteiros. Use índices para ler e escrever os elementos do vetor.
5. Reescreva o programa do item 2, usando ponteiros para acessar os elementos do vetor.
6. Converta as seguintes instruções em linguagem de montagem do MIPS, para instruções em linguagem de máquina. A primeira instrução deve estar no endereço 0x00400014. Faça a conversão manualmente.

```
fact:
    sub    $sp, $sp, 8
    sw     $ra, 4($sp)
    sw     $a0, 0($sp)
    slti   $t0, $a0, 1
```

7. Escreva as correspondentes instruções em linguagem de montagem para as seguintes instruções em linguagem de máquina:

Endereço	Código (em binário)
0x00400000	00000000100011011111000000100000
0x00400004	0000100000010000000000000000100
0x00400008	10101110010010010000010011010010
0x0040000c	0000100000010000000000000000101
0x00400010	0000100000010000000000000000010
0x00400014	00000000111000010001100000101010
0x00400018	00001000000100000000000000000000

8. Converta as seguintes instruções em linguagem de montagem do MIPS, para instruções em linguagem de máquina. Faça a tradução manualmente. Utilize o MARS somente para verificar a sua resposta

```
loop1:
    sw $zero, 0($t0)
    slt $t3, $t0, $t2
    bne $t3, $zero, loop2
loop2:
    addi $t0, $t0, 4
    j loop1
```