

Universidade Federal de Pernambuco
Centro de Tecnologia e Geociências
Departamento de Engenharia Mecânica

Lista de Exercícios #01 - 2024.1

O objetivo desta lista é programar alguns algoritmos simples em assembler utilizando o **VisUAL ARM Emulator**¹ e com isso entender melhor como funcionam os códigos em um nível mais baixo de abstração.

Instruções Gerais

- **Sempre comente seu código e identifique o que ele faz e quem foi que fez (você, no caso);**
- Salve o arquivo de cada questão no padrão: LE02_Qx_SEU_NOME.asm; em que x é o número da questão e SEU_NOME é o seu nome;
- Sempre teste os códigos para diferentes valores de entrada, quando isso for possível (questões 3, 4, 5, e 6);

Exercícios

1. **Usando valores imediatos e acesso a memória:** faça um código que:
 - carregue os valores {0x12, 0x12AB, 0x12AB34, 0x12AB34CD} nos registradores R0 ao R3, respectivamente;
 - Após isso, salve esses registradores R0 ao R3 na memória de dados nos endereços 0x100 ao 0x10C, respectivamente;
 - Por fim, leia a memória de dados e salve nos registradores R8 ao R11.
2. **Usando laços:** faça um código que calcule os 30 primeiros termos da sequência de Fibonacci e os guarde na memória de dados, para isso utilize um laço, sabendo que os dois primeiros termos da sequência são $F(0) = 1$ e $F(1) = 1$ e que o n-ésimo termo é dado por $F(n) = F(n-1) + F(n-2)$.
3. Para o código implementado na Questão 2, até que termo da sequência é possível calcular sem que haja *overflow* do resultado?
4. **Multiplicação em baixo nível:** faça um código que calcule a multiplicação entre dois números e salve a resposta na **memória de dados**.
5. **Reutilizando código:** faça um código que calcula o fatorial de um número, para isso, reaproveite o código feito para calcular a multiplicação entre dois número. Salve o resultado na **memória de dados**.

¹Disponível em: <https://salmanarif.bitbucket.io/visual/index.html>

6. **Uso de algoritmos clássicos:** faça um código que calcule a divisão euclidiana entre dois números e salve os valores de quociente e resto na memória de dados.

OBS: Divisão Euclidiana: $\frac{d_i}{d_v}$

$$d_i = d_v \times q + r \quad (1)$$

em que:

- d_i = dividendo;
- d_v = divisor;
- q = quociente;
- r = resto;

$\{d_i, d_v, q, r\} \in \mathbb{N}$ e $d_v \neq 0$.

7. Implemente o algoritmo de Euclides para calcular o MDC entre dois números inteiros m e n , em que $0 \leq n < m$, cuja uma implementação em **C**²:

```
1 // Para calcular o mdc(m,n) para 0 <= n < m
2 int euclides_mdc(int m, int n)
3 {
4     if (n==0)
5     {
6         return m;
7     }
8     return euclides_mdc (n, m % n);
9 }
```

e salve o resultado na **memória de dados**.

Links Importantes

- Página do VisUAL ARM Emulator: <https://salmanarif.bitbucket.io/visual/>
- Lista de instruções suportadas pelo emulador: https://salmanarif.bitbucket.io/visual/supported_instructions.html
- User Guide do emulador: https://salmanarif.bitbucket.io/visual/user_guide/index.html
- O Algoritmo de Euclides: <https://www.ime.usp.br/~coelho/mac0338-2004/aulas/mdc/index.html>

²FONTE: <https://www.ime.usp.br/~coelho/mac0338-2004/aulas/mdc/index.html>