

## Microcontroladores – exercício extra

### Considerações Iniciais

Para poder executar o exercício desta aula, é necessário possuir o seguinte conjunto de conhecimentos:

- conceito de stack, sua utilidade e implementação física no contexto do i8051

No final da aula deverá ser capaz de

- escrever programas que façam uso intensivo de stack
- dominar a utilização da stack e compreender o mecanismo de gestão da mesma através do stack pointer

### Objectivo

Exemplificar a utilização da Stack

### Introdução

A notação PostFix (também conhecida por polonesa inversa), muito popular em algumas marcas conceituadas de máquinas calculadoras, assenta num processamento com utilização intensiva da Stack da máquina.

Os adeptos deste tipo de notação, invocam como uma das principais vantagens, a rapidez da introdução de dados.

A utilização da notação acenta no facto de que a introdução dos operandos é contígua, e seguida do operador que os relaciona. Vejam-se os seguintes exemplos:

$$\begin{array}{rcl} 3\ 5\ + & = & 3 + 5 \\ 3\ 5\ 7\ +\ * & = & (7 + 5) * 3 \\ 3\ 5\ 7\ +\ * \ 3\ +\ 5\ * & = & (((7+5)*3)+3)*5 \end{array}$$

### Funcionamento

O processador de notação polonesa inversa deve aceitar um apontador para memória externa, que indica o início da string, e efectuar o processamento da mesma deixando o resultado no topo da Stack.

Fundamentalmente, o processador funciona do seguinte modo:

- i) se o carácter em análise fôr um dígito, ele coloca-o na Stack
- ii) se o carácter fôr um operador matemático, ele retira dois operandos da Stack, efectua a operação e coloca o resultado na Stack
- iii) prossegue com esta metodologia até alcançar o final da string.

Considere que a string termina com o carácter fffh, que por esta razão não é um operando válido.

Construa o algoritmo para processamento de notação PostFix  
Construa um programa capaz de processar strings em PostFix

OBS: admita que cada operando é apenas constituído por um dígito decimal