

Experimento 06

Objetivo:

Projete e escreva um programa em linguagem assembly para ARM que irá determinar se os parênteses em uma string estão corretamente balanceados. Parenteses balanceados significa que cada símbolo de abertura tem um símbolo de fechamento correspondente e os pares de parênteses estão perfeitamente aninhados. Por exemplo, considere a sequência de parênteses balanceados mostrdos a seguir:

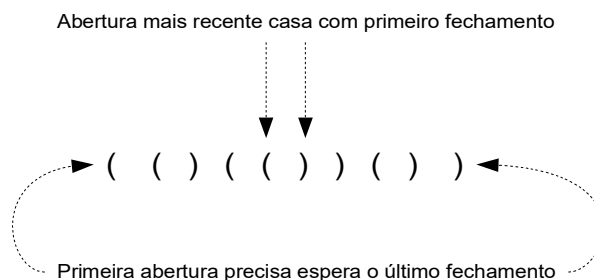
- `((()()))`
- `((()))`
- `((()((()()))))`

Comparados com os exemplos seguintes que são não balanceados:

- `(((((())`
- `()))`
- `((()())`

A habilidade de diferenciar entre parênteses que são corretamente balanceados é uma importante do reconhecimento de estruturas de linguagens de programação. O desafio é escrever um algoritmo que irá ler a string de parênteses da esquerda para a direita e decidir quando os simbolos estão balanceados.

Para resolver este problema é importante levar em conta a seguinte observação: durante o processamento dos símbolos da esquerda para a direita, o parêntese mais recentemente aberto precisa casar com o símbolo de fechamento mais próximo. Também, o primeiro símbolo aberto processado precisa ter que esperar até que o último símbolo de fechamento seja detectado para seu correto casamento. Símbolos de fechamento casam com símbolos de abertura numa ordem reversa; eles combinam de dentro para fora.



Isto é uma pista de que o uso de pilha pode ser usado para resolver este problema. Uma saída é iniciar com uma pilha vazia e iniciar o processamento da esquerda para a direita. Se um símbolo é um parêntese de abertura, coloque-o na pilha. Se um símbolo de parêntese de fechamento aparecer, então tire um símbolo de abertura da pilha. Se todos os parênteses estão corretamente balanceados, então a pilha deverá terminar vazia.

Utilize o aplicativo **µVision5** para implementar e depurar o programa. O programa final deve ser postado em uma tarefa do Moodle.