



Universidade de São Paulo - USP  
Instituto de Ciências Matemáticas e de Computação – ICMC  
Departamento de Sistemas de Computação – SSC  
SSC0112 Organização de Computadores Digitais I  
1o. Trabalho Prático – disponibilizado em 15/03/2018

## Implementação de uma Árvore de Busca Digital Binária

O objetivo deste trabalho é implementar em Assembly MIPS uma Árvore de Busca Digital Binária (Árvore Digital Binária, *Binary Tree* ou *Bitwise Trie*), para armazenar e recuperar números binários. O programa em assembly MIPS feito será executado no simulador SPIM ou no MARS.

As opções previstas para esta aplicação são: inserção, remoção, busca e visualização da Árvore Digital Binária. Estas quatro opções são acessadas via um menu, acrescido apenas da opção de finalização da aplicação, como última opção. A visualização da árvore deve ser feita por amplitude, onde cada linha impressa contém os nós de um determinado nível da árvore (começando pela raiz). A operação de busca deve retornar o valor da chave pesquisada e o caminho usado na árvore para a encontrar, ou -1 caso a chave pesquisada não tenha sido encontrada.

A árvore deve ser implementada com alocação dinâmica e não estática (vetores). Os dígitos das chaves inseridas não devem ser armazenados nas estruturas de dados de cada nó da árvore, pois os próprios links (0 e 1) dos filhos dos nós representam tais dígitos. Os demais campos da estrutura de cada nó da árvore não são especificados no trabalho, propositalmente.

Ao entrar nas opções de inserção, remoção e busca, o usuário deve fornecer uma chave e realizar a operação solicitada. Ao final destas operações, uma nova chave deve ser solicitada. O retorno ao menu de opções só ocorre quando o valor -1 (menos um) for fornecido. A opção de visualização retorna ao menu de opções após exibir a Árvore de Busca Digital Binária completamente.

As chaves para inserção, remoção e busca, devem ser fornecidas diretamente em binário. Esses dígitos binários podem ser tratados como uma *string* (cadeia de caracteres), porém, não devem permitir a inserção de dígitos diferentes de **0** e **1**. Não devem ser permitidas inserções de chaves replicadas na Árvore Digital Binária.

A correção levará em conta a execução correta do algoritmo e a qualidade do código fonte feito (qualidade inclui, mas não está limitada a: indentação, comentários corretos, nome significativos para rótulos, espaços entre porções do código, etc).

O trabalho deverá ser feito em grupo e submetido no Moodle até a data já definida.

Quaisquer características omissas a esta especificação deverão ser reportadas ao professor para orientação de como proceder na construção dos algoritmos.

## Referências Iniciais

<https://www.ime.usp.br/~pf/estruturas-de-dados/aulas/tries.html>

<http://www.mathcs.emory.edu/~cheung/Courses/323/Syllabus/Text/trie01.html>

Jaime Szwarcfiter; Lilian Markenzon "Estruturas de Dados e Seus Algoritmos", LTC, 2014.

Aaron M. Tenenbaum; Yedidyah Langsam; Moshe J. Augenstein. "Estruturas de Dados Usando C", Makron Books., 2005.