

# Trabalho Prático

## Algoritmos de Pesquisa

### Profa. Jaqueline Faria de Oliveira

#### Alunos:

- Ivan Paulovich Pinheiro Gomes RA11323689
- Josiele Alves de Araújo Durães RA11412030

#### Árvores binárias de busca

Assim como árvores binárias generalizam a ideia de listas encadeadas, árvores binárias de busca (= binary search trees = BSTs) generalizam a ideia de listas encadeadas crescentes.

Uma árvore binária de busca serve para o armazenamento de dados na memória do computador e a sua subsequente recuperação.

Em uma árvore binária de busca cada nó contém um campo chamado key, podendo haver outras informações, além dos ponteiros left e right. O campo key especifica em geral uma chave que identifica de forma única um determinado registro ou informação.

Exemplos de chaves: número de identidade, número CPF, etc. Assim, suporemos que todos os valores de key são distintos. Dado um valor qualquer, deseja-se localizar o nó da árvore, se houver, cujo key é igual ao valor dado.

Característica da árvore binária de busca: Para todo nó  $x$  da árvore binária de busca, as seguintes condições são verificadas.  $key(x) > key(y)$ , para todo nó  $y$  da subárvore esquerda.  $key(x) < key(y)$ , para todo nó  $y$  da subárvore direita.

#### Hash

Em ciência da computação a tabela hash (de hashing, no inglês), também conhecida por tabela de espalhamento, é uma estrutura de dados especial, que associa chaves de pesquisa (hash) a valores. Seu objetivo é, a partir de uma chave simples, fazer uma busca rápida e obter o valor desejado. É algumas vezes traduzida como tabela de escrutínio.

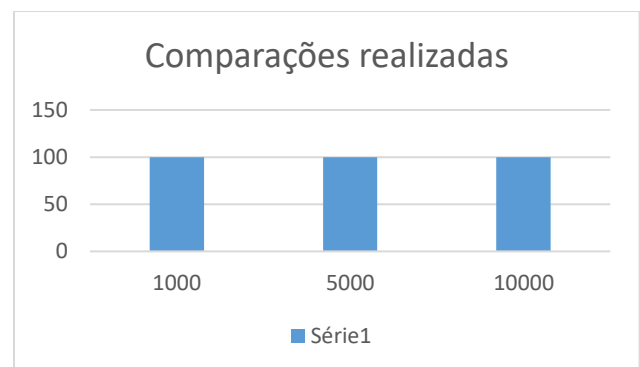
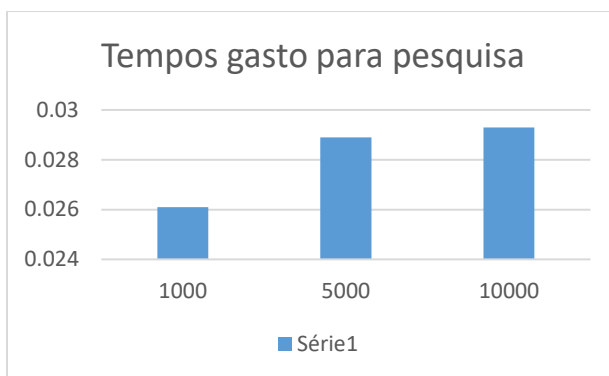
#### Código Fonte

O código fonte acompanha este documento e foi desenvolvido com C#, .NET Core e Visual Studio 2017.

## Resultados

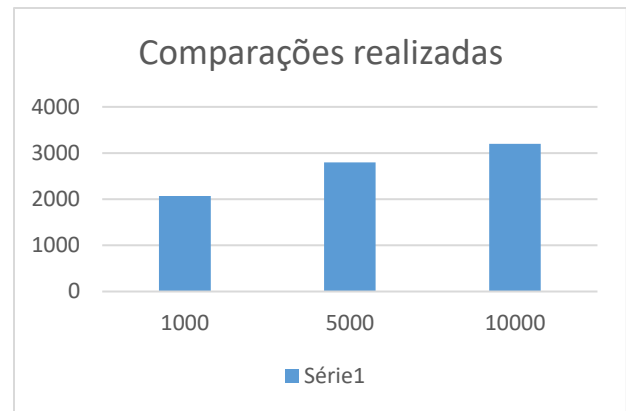
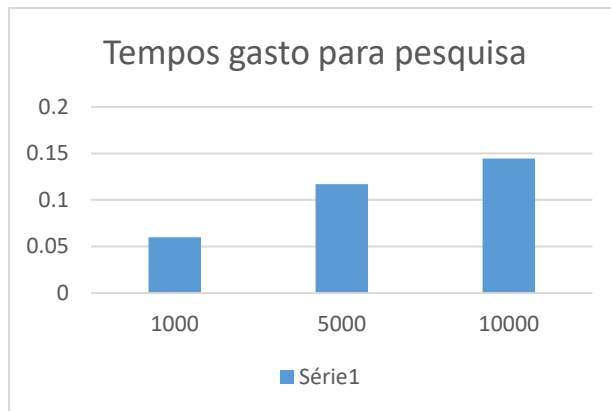
Select C:\Program Files\dotnet\dotnet.exe					
Tempo (ms)			Comparacoes		
	Hash	BST	Hash	BST	
1k	0.0261	0.0601	100	2098	
5k	0.0289	0.117	100	2800	
10k	0.0293	0.1446	100	3202	

## Hash



A pesquisa em uma estrutura de dados Hash é bastante rápida e a variação de tempo foi de 0.02 centésimos de segundos. Em relação a quantidade de comparações, o valor se manteve constante.

## Árvore Binária de Pesquisa



A pesquisa em uma estrutura de dados Árvore Binária de Pesquisa é bastante rápida e a variação de tempo foi de 0.1 centésimos de segundos. Em relação a quantidade de comparações, o valor aumentou em árvores com mais dados.

## Referências

- Árvore Binária de Busca  
<https://www.ime.usp.br/~song/mac5710/slides/06bst.pdf>
- A simple Binary Search Tree written in C#  
<https://www.codeproject.com/Articles/18976/A-simple-Binary-Search-Tree-written-in-C>
- A simple Binary Search Tree written in C#  
<https://www.codeproject.com/Articles/18976/A-simple-Binary-Search-Tree-written-in-C>
- Introduction To Hashing and the HashTable Class: Part 2  
<http://www.c-sharpcorner.com/UploadFile/a85b23/introduction-to-hashing-and-the-hashtable-class-part-2/>