

# Relatório - Trabalho 3

## Organização e Arquitetura de Computadores - Turma C

Gabriel Vieira de Arimatéa, 15/0126956

<sup>1</sup>CIC – Universidade de Brasília (UnB)

vieira.arimatea@gmail.com

### 1. Descrição do problema

O trabalho consiste em implementar um código em *assembly* MIPS que realize operações sobre uma árvore binária de busca usando funções recursivas.

### 2. Descrição sucinta das funções implementadas

As principais funções implementadas são a função **menu**, **input**, **inserção**, **busca**, e **contaNos**:

#### 2.1. menu

Função para criar uma interface básica com o usuário, apresentando as opções de inserir um nó, buscar um nó, contar quantos nós a árvore possui e sair do programa;

#### 2.2. input

Função que recebe a resposta do usuário referente à função **menu**;

#### 2.3. inserção

Função que insere um nó na árvore. Sempre será adicionado de forma que todos os nós possuam seus filhos com valor maior à direita e com valor menor à esquerda. Os nós são inseridos de forma que a raiz está armazenada no endereço armazenado em **\$gp** e sempre os filhos de um nó estão localizados nos endereços  $2 * n + 1$  e  $2 * n + 2$ , sendo **n** o endereço do nó analisado subtraído de um *offset* (no caso o *offset* é o valor armazenado em **\$gp**);

#### 2.4. busca

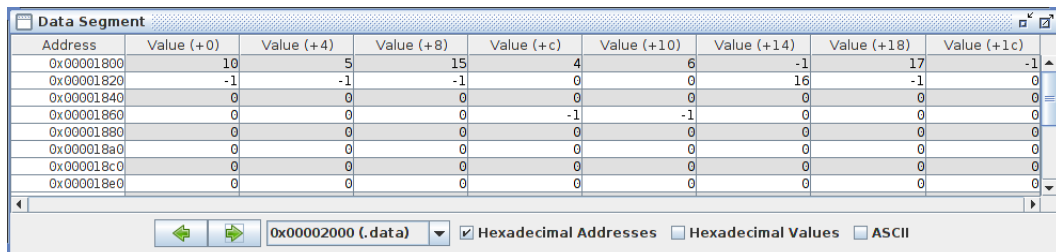
Consiste em uma função que busca um nó desejado, retornando se o encontrou ou se não;

#### 2.5. contaNos

Função que realiza a contagem de quantos nós a árvore possui.

### 3. Testes e resultados

Para o teste, inseriu-se os valores 10, 5, 15, 17, 4, 6, 16 nessa ordem. Os dados foram armazenados da seguinte forma na memória de dados:



Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x00001800	10	5	15	4	6	-1	17	-1
0x00001820	-1	-1	-1	0	0	16	-1	0
0x00001840	0	0	0	0	0	0	0	0
0x00001860	0	0	0	-1	0	0	0	0
0x00001880	0	0	0	0	0	0	0	0
0x000018a0	0	0	0	0	0	0	0	0
0x000018c0	0	0	0	0	0	0	0	0
0x000018e0	0	0	0	0	0	0	0	0

Figura 1. Nós armazenados

Os valores -1 indicam nós vazios, ou seja, são os "filhos" das folhas da árvore.

Em seguida, foi testada a função contaNos. O valor retornado pode ser visualizado na imagem abaixo:

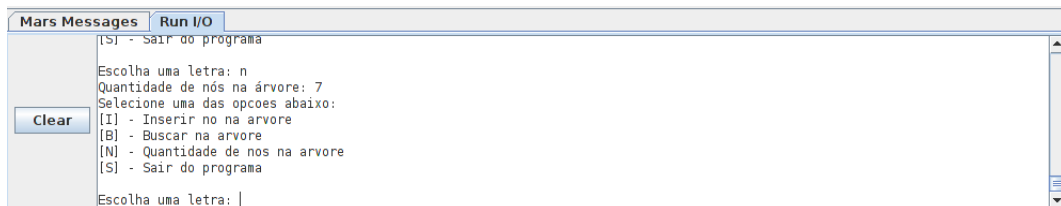


Figura 2. Retorno da função contaNo

A função seguinte a ser testada foi a busca. Para isso procuramos primeiramente o valor 5 (existente na árvore) e depois o valor 2 (não existente na árvore).

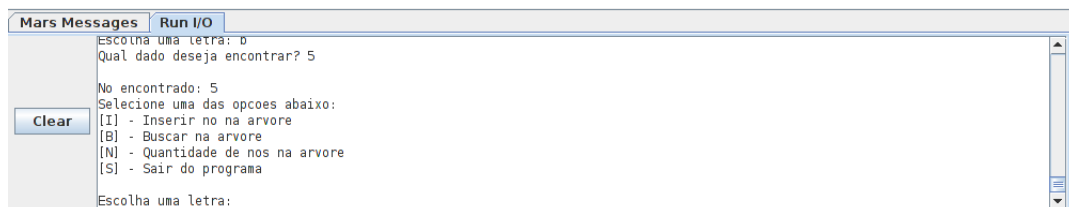
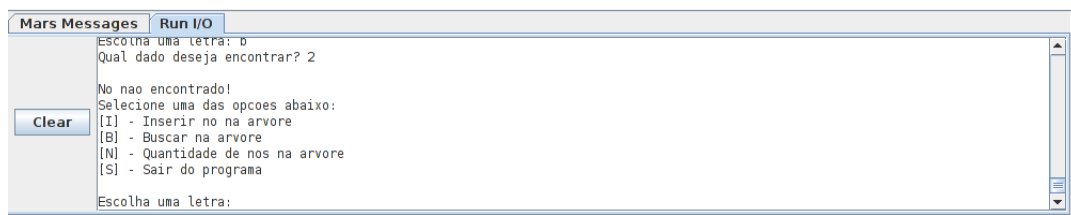


Figura 3. Retorno da busca com nó existente



**Figura 4. Retorno da busca com nó inexistente**

A função contaNos foi a mais trabalhosa para se implementar, uma vez que sua contagem inicialmente não estava correta, mas esse erro foi corrigido.

Com isso todos os testes foram concluídos.