

# ***Linguagem de Programação e Aplicações***

## ***Trabalho 2***

***Profº Adenilso Simão***

### **Introdução:**

Este trabalho tem por objetivo explorar nossos conhecimentos na manipulação de árvores em linguagem C, tanto como árvores binárias de busca ou não.

O trabalho consiste basicamente em dois problemas: o primeiro nos pede para construirmos uma árvore binária de busca, e permitirmos ao usuário realizar uma lista de operações nesta árvore. O segundo problema, por sua vez, nos pede a construção de uma árvore genealógica, seguida da possibilidade do usuário realizar uma série de operações nesta.

Alunos envolvidos neste trabalho:

1- Bruno Henrique de Souza – nº USP: 9311685

2- Bruno Ávila – nº USP: 9012667

3- Evandro de Castro Pinto – nº USP: 9312856

### **Descrição do Projeto:**

Os códigos fonte dos programas foram compilados em C (.c), no ambiente de desenvolvimento Dev-C++, versão 5.10.0.0;

O compilador dos executáveis (.exe) foi TDM-GCC 4.8.1 64-bit Release;

Os parâmetros de compilação são os padrões do ambiente;

As plataformas de compilação da versão binária foram MinGW64\bin e MinGW54\x86\_64-w64-mingw32\bin;

As bibliotecas utilizadas em todos os programas foram stdio.h e stdlib.h, além da biblioteca string.h no programa do exercício 2;

No repositório encontra-se uma pasta chamada "Entradas-exemplo". Nela estão escritas, em formato .txt, exemplos de entradas para serem testadas pelos programas, além de desenhos de algumas entradas-exemplo para que o usuário possa ter uma imagem do que se trata a árvore em questão;

Os códigos fonte e seus executáveis se encontram no repositório desse trabalho;

Para a compilação, os programas devem ser compilados da maneira padrão do ambiente Dev-C++. Para tal, basta ir ao menu Executar, e clicar em Compilar. Outros ambientes podem se dar por meios diferentes, como a tecla "Build", ou outros.

Contudo, o princípio da compilação é o mesmo. Uma vez compilado, o programa está pronto para ser executado.

## **Tutoriais:**

### **1) Operações básicas em árvores:**

1.1- O programa imprimirá uma mensagem perguntando ao usuário quantos elementos terá a sua árvore. Depois de informado ao programa, uma mensagem aparecerá pedindo ao usuário que diga quais serão esses elementos;

1.2- O usuário deve então entrar com os elementos que deseja ter em sua árvore, separando cada elemento por Enter. É preciso lembrar que esta árvore armazena valores inteiros, então é preciso que o usuário entre com valores inteiros para que o programa funcione apropriadamente;

1.3- Uma vez que todos os elementos foram inseridos, o programa já vai ter feito a construção da árvore binária de busca. Neste momento, uma lista aparecerá ao usuário, perguntando qual daquelas operações o usuário deseja realizar;

1.4- A primeira opção é Inserção. Aqui, semelhantemente ao início do programa, será perguntado quantos elementos serão inseridos e quais são tais elementos;

1.5- A segunda opção é Remoção. Nesta operação, será perguntado ao usuário qual elemento ele deseja remover da árvore. Depois de informado, o programa irá remover tal elemento de rearranjar a árvore para que ela continue sendo uma árvore binária de busca. O programa funciona inclusive para a remoção da raiz;

1.6- A terceira opção é Busca. Aqui, o programa perguntará ao usuário qual elemento ele deseja buscar na árvore. Caso não haja tal valor na árvore, o programa imitará uma mensagem afirmando isso. Caso contrário, uma mensagem aparecerá dizendo que o valor digitado foi encontrado na árvore;

1.7- A quarta opção é impressão em Pré-Ordem. Nesta operação, os elementos da árvore serão impressos de acordo com a ordenação conhecida como pré-ordem;

1.8- A quinta opção é impressão em Pós-Ordem. Esta operação irá imprimir os elementos da árvore na ordenação conhecida como pós-ordem;

1.9- A sexta opção é impressão Em Ordem. Aqui, o programa imitará os valores da árvore em ordem. Numa árvore binária de busca, como é o caso da nossa em questão, os valores aparecem em ordem crescente nesta operação;

1.10- A sétima opção é Labelled Bracketing. Esta operação imprime os valores da árvore binária de busca numa ordenação conhecida como Labelled Bracketing, caracterizada pela presença de colchetes numa forma de representar a árvore;

1.11- Caso o usuário digite um valor de opção que não seja uma dessas sete, o programa dirá a ele que entrou com um valor inválido. Ao fim de cada operação, o programa perguntará ao usuário se ele deseja continuar no programa. Caso a resposta for afirmativa, a lista aparecerá novamente para que uma nova operação possa ser realizada. Caso contrário, o programa irá se despedir, e ser encerrado.

## 2) Árvore genealógica:

2.1- Neste programa, será montada uma árvore genealógica. Para tal, o programa irá perguntar ao usuário quantas tuplas ele deseja ter em sua árvore. Uma tupla é um trio composto por uma pessoa e seus pais. Por exemplo, “Pedro João Maria” é uma tupla, onde Pedro é o filho, e João e Maria são seus pais. Uma segunda tupla deve informar os avós de Pedro, ligando-os com João (avós paternos) ou com Maria (avós maternos), por exemplo, “João José Marília”, onde João é considerado o filho de José e de Marília;

2.2- Depois de informado o número de tuplas na árvore, o programa pedirá ao usuário para informar tais tuplas. O programa lembra o usuário que a ordem que ele digita os nomes é importante, sendo que o primeiro deve ser o filho, o segundo o pai e o terceiro a mãe;

2.3- Depois de terminada a alocação dos nomes na árvore, aparecerá uma lista de opções que o usuário pode fazer em sua árvore genealógica;

2.4- A primeira opção imprimirá os membros da família por gerações. Membros que estão na mesma geração serão impressos na mesma linha na saída do programa;

2.5- A segunda opção irá imprimir os antepassados de um dado indivíduo. O programa pedirá ao usuário que diga qual pessoa da família ele quer conhecer os antepassados. Depois de informado, o programa mostrará todos os antepassados dessa pessoa, não necessariamente numa ordem que informe as gerações a partir de tal pessoa. Se não aparecer nome algum, significa que tal pessoa não tem antepassados representados na árvore genealógica em questão;

2.6- A terceira opção é labelled bracketing. Aqui, a árvore genealógica será impressa na ordenação labelled bracketing, semelhante a usada no exercício 1;

2.7- A quarta opção é para se determinar o grau de parentesco. Esta operação irá perguntar ao usuário entre quais pessoas ele deseja saber o grau de parentesco. É preciso ter em mente que o programa só irá funcionar apropriadamente se as pessoas realmente estiverem inseridas na árvore. Caso essa condição seja atendida, o programa irá mostrar uma mensagem afirmando qual é o grau de parentesco entre tais pessoas;

2.8- A quinta opção é Inserção. Ela perguntará ao usuário qual é a tupla que ele deseja inserir. É importante manter em mente que o filho dessa nova tupla deverá ser um membro da família, e mais especificamente algum membro que não tenha os pais já inscritos na árvore;

2.9- Caso nenhuma das cinco opções acima seja escolhida pelo usuário, o programa imitará uma mensagem falando que uma opção inválida foi digitada. Ao fim de cada operação, será mostrada uma mensagem perguntando ao usuário se ele deseja continuar no programa. Caso a resposta for positiva, a lista aparecerá novamente. Caso contrário, o programa irá se despedir, e será encerrado;

Obs: O programa só funciona com tuplas completas. Não irá adiantar se o usuário souber apenas o filho e um dos pais. É necessário que o filho, o pai e a mãe sejam informados para que possam ser inseridos.