

Aluno: Thiago Pereira Camerato

Matrícula: 2212580

Introdução: O programa desenvolvido tem como objetivo selecionar dois competidores, A e C, que irão competir com o competidor B em rodadas de natação. A seleção é baseada na altura dos competidores, seguindo critérios específicos. O competidor A deve ser o mais alto entre os que são mais baixos que B, e o competidor C deve ser o mais baixo entre os que são mais altos que B. Para isso, foi utilizada uma Árvore Binária de Busca (ABB) para armazenar as alturas dos competidores e realizar as seleções de forma eficiente.

Estrutura do Programa:

O programa consiste em três arquivos **trabalho2_main.c**, **trabalho2_aux.c**, e **trabalho2_aux.h**.

- **trabalho2_main.c** é o arquivo principal que contém a função **main** onde o programa começa sua execução. Ele é responsável por criar a ABB, inserir os competidores na árvore, solicitar a altura do competidor B ao usuário e chamar a função **findCompetitors** para realizar a seleção dos competidores A e C. Também imprime as alturas dos competidores em ordem e a quantidade de nós visitados.
- **trabalho2_aux.c** contém as implementações das funções auxiliares necessárias para a manipulação da ABB e a seleção dos competidores. Estas funções incluem a inicialização da árvore, inserção de nós, busca dos competidores A e C, impressão dos nós em ordem e liberação de memória.
- **trabalho2_aux.h** contém as declarações das estruturas de dados e protótipos de funções utilizadas no programa.

Solução:

A solução consiste em criar uma ABB para armazenar as alturas dos competidores e, em seguida, utilizar a função **findCompetitors** para selecionar os competidores A e C com base na altura do competidor B fornecida pelo usuário.

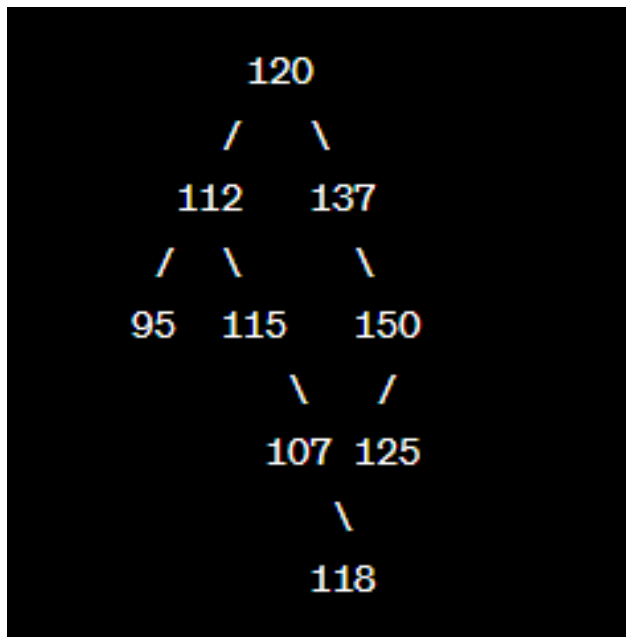
- **Inicialização da ABB:** A função **iniciatetree** inicializa uma árvore vazia.
- **Inserção dos Competidores:** No arquivo **main.c**, são inseridos os valores das alturas dos competidores na ABB com a função **insertNode**.
- **Seleção dos Competidores A e C:** A função **findCompetitors** é responsável por encontrar os competidores A e C com base na altura do competidor B. Ela percorre a árvore de forma eficiente, contando o número de nós visitados. Os competidores A e C são selecionados de acordo com os critérios mencionados na introdução.
- **Impressão dos Resultados:** O programa imprime as alturas dos competidores em ordem, a altura do competidor B, e as alturas dos competidores A e C selecionados. Caso não seja possível selecionar competidores que atendam aos critérios, é exibida a mensagem "Competição inválida".

Observações e Conclusões:

- A implementação da ABB e das funções auxiliares permite uma busca eficiente dos competidores A e C com um número mínimo de nós visitados.

- A estrutura da ABB é adequada para armazenar e buscar as alturas dos competidores de forma eficiente, o que é essencial para a seleção dos competidores.
- As mensagens de "Competição inválida" são exibidas corretamente quando não é possível encontrar competidores que atendam aos critérios.
- Em geral, a solução implementada atende aos requisitos do problema de forma eficiente e clara, proporcionando uma seleção justa dos competidores de acordo com os critérios estabelecidos.

Árvore após as inserções:



Output do programa:

- Teste 1:

```

$ ./trabalho2.exe
Alturas dos competidores em ordem:
95 107 112 115 118 120 125 137 150

Digite a altura do competidor B:
112
Competitor B: 112
Competitor A: 107
Competitor C: 115

Numero de nos visitados: 2
  
```

- Teste 2:

```
$ ./trabalho2.exe
Alturas dos competidores em ordem:
95 107 112 115 118 120 125 137 150

Digite a altura do competidor B:
150
Competitor B: 150
Competitor A: 137
Competition invalid

Numero de nos visitados: 3
```

- Teste 3:

```
$ ./trabalho2.exe
Alturas dos competidores em ordem:
95 107 112 115 118 120 125 137 150

Digite a altura do competidor B:
107
Competitor B: 107
Competitor A: 95
Competitor C: 112

Numero de nos visitados: 4
```