**Projeto AED**

**Eleições e contagem de Votos**

Alunos: Jaime Victor, Ana Paula e Andressa

**Classes e objetos:**

Existem 4 objetos implementados:

· Nó:

Este objeto possui os atributos:

-Pai

É um ponteiro que referencia um nó pai.

-Filho esquerdo

É um ponteiro que referencia um outro nó, menor do que o observado.

-Filho direito

É um ponteiro que referencia um outro nó, maior do que o observado.

-Valor

Guarda o conteúdo do nó em si, o próprio valor.

-Vermelho

Funciona com uma variável booleana para definir a cor do nó, True para de o nó for vermelho, False para se o nó for preto. Todo nó é inicialmente vermelho.

\*\*Todos os atributos possuem os métodos get e set.

· Nil:

Funciona como um nó substituto para o None

Possui 2 atributos:

-Valor

Sempre None.

-Vermelho

Sempre False, funciona da mesma forma que no nó, por convenção.

\*\*possui apenas o método getvermelho(), usado na verificação da árvore.

Árvore binária:

Possui 2 atributos:

-Nil

Marca as extremidades da árvore.

-Raiz

Indica onde é o primeiro elemento da árvore.

\*\*Possui os devidos métodos get e set, para ambos.

Possui os métodos:

-ehvazio()

Diz se a árvore é vazia ou não.

-ehfilhoesquerdo()

Diz se um nó é filho esquerdo do pai.

-ehfilhodireto()

Diz se um nó é filho direito do pai.

-inserir()

Navega pelos elementos existentes (olhando para esquerda, caso o nó a ser inserido seja menor ou igual ao nó analisado, caso seja maior, navega para direita. Até um nó folha), e insere o novo nó como um nó folha.

-procurar()

Recebe um valor e retorna None se o nó não estiver na árvore, e retorna o nó com aquela chave caso esteja.

-preorder()

Exibe o nó, em seguida o seu nó esquerdo, e então seu direito.

-posorder()

Exibe o lado esquerdo do nó primeiro, em seguida todo lado direito, por fim o nó.

-inorder()

Exibe o lado esquerdo, em seguida o nó, só então exibe o lado direito.

-minimo()

Exibe o menor valor de a partir de um nó, usualmente a raiz para exibir o menor valor da árvore.

-maximo()

Exibe o maior valor de a partir de um nó, usualmente a raiz para exibir o maior valor da árvore.

-sucessor()

Exibe o menor número maior que o nó recebido.

-predecessor()

Exibe o maior número que é menor que o nó recebido.

-remover()

Remove completamente o nó da árvore.

Árvore Vermelho e preta (Herda a árvore binária):

A árvore VP implementada possui os seguintes métodos:

-inserirVP()

O método inserir é igual a um método inserir de uma árvore binária, porém define os filhos direito e esquerdo como Nil, e chama o método de ajustar inserção da árvoreVP.

-rotacaoesquerda()

A rotação esquerda age como uma das ferramentas para a correção do balanceamento da árvore, ela move uma dupla de nós reorganizando seus filhos.

-rotacaodireita()

A rotação esquerda age como uma das ferramentas para a correção do balanceamento da árvore, ela move uma dupla de nós reorganizando seus filhos.

-ajustarinsercao()

Esse método usa das rotações para manter as propriedades de uma árvore VP, o algoritmo inicia a visualização das regras no nó inserido, caso alguma regra tenha sido quebrada ele faz as devidas correções se utilizando das rotações.

-ajustarremocao()

Esse método funciona da mesma forma que o método de ajustar inserção.

-procurarVP()

Recebe um valor e retorna None se o nó não estiver na árvore, e retorna o nó com aquela chave caso esteja.

-preorderVP()

Exibe o nó, em seguida o seu nó esquerdo, e então seu direito.

-posorderVP()

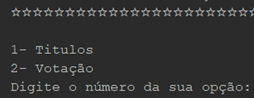
Exibe o lado esquerdo do nó primeiro, em seguida todo lado direito, por fim o nó.

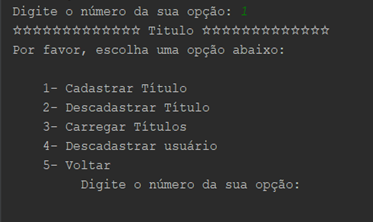
-inorder()

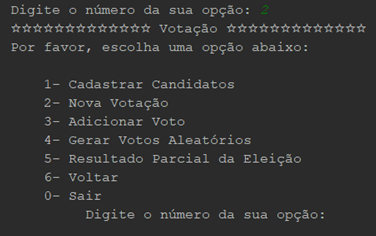
Exibe o lado esquerdo, em seguida o nó, só então exibe o lado direito.

Interface:

Aqui é possível ver o menu inicial do projeto (1), o menu de gerenciamento de títulos (2) e o menu de gerenciamento de votação (3)

(1)

(2)

(3)

Funcionamento:

\*\*Com o intuito de evitar repetição serão referenciadas as imagens acima com os respectivos números

(1) Menu inicial-

1-Essa opção leva à tela (2)

2-Essa opção leva à tela (3)

(2) Menu Títulos –

1- Cadastrartitulo()

O programa pede uma entrada, o número do título, e verifica se esse título já existe, caso exista ele exibe uma mensagem de erro, do contrário o título será cadastrado e o valor do título será armazenado na árvore vermelha e preta.

2-DescadastrarTitulo()

Pede um título como entrada, verifica se existe, se existir, remove o título da árvore vermelha e preta.

3-CadastrarTitulo()

Preenche a árvore de títulos aleatoriamente.

4-Voltar

Retorna ao menu inicial

(3) Menu votação –

1-CadastrarCandidatos()

Exige como entrada o nome e o título do candidato, então verifica se ele já se cadastrou (se o candidato estiver na lista de cadastrados), caso esteja, ele acusa que o candidato ja foi cadastrado, se não, ele o cadastra.

2-NovaVotacao()

Apaga todos os elementos da árvore.

3-AdicionarVoto()

Exige como entrada o título do candidato, então verifica se ele já realizou a votação (se o título estiver na arvoreVP), caso esteja, ele acusa que o voto já foi realizado, caso não, ele adiciona à árvore o título do candidato.

4-GerarVotosAleatorios()

Simula uma votação aleatória.

5-ResultadoParcial()

Mostra a quantidade de votos de cada candidato até o momento.

6- Sair()

Apaga todo conteúdo das árvores e fecha o programa.