

TDE 2 Resolução de problemas estruturados em computação.

O programa implementado funciona, semelhante aos quebra cabeças “Torre de Hanoi”. Permitindo o usuário realizar quantos movimentos achar necessário para preencher as torres, a partir de um tamanho de pilhas informado, ou selecionar uma opção para preenchimento automático, a partir de informado o tamanho das pilhas também. O programa gera números aleatórios entre 0 e 100 para preencher as pilhas e apresenta menus, permitindo o usuário preencher manualmente as pilhas, ou utilizar uma solução automática, a partir de um tamanho informado. Ao final da execução é informada a quantidade de jogadas que foram realizadas para ganhar o jogo, e a impressão do estado final das torres.

A implementação possui 3 classes, a Pilha, a classe Jogo e a classe Main.

. Classe Pilha:

- . Apresenta 3 dados informações, capacidade, topo e os dados.
- . No construtor da classe é passada a capacidade, o topo é iniciado em -1 e os dados recebe um array de inteiros com a capacidade.
- . Apresenta métodos getteres e setteres para a capacidade, topo e os dados.
- . Apresenta métodos pop, que remove valores da pilha, push -> que insere elementos na pilha, possui também os métodos pilhaCheia, pilhaVazia e ImprimePilha, esses métodos vão verificar se a pilha está cheia ou vazia e o método imprime vai imprimir a pilha.

. Classe Jogo:

- . A classe jogo possui 3 torres nomeadas, de torre 1, torre 2 e torre 3 do tipo pilha.
- . Possui um scanner para ler dados do teclado, um contador de jogadas do tipo inteiro e um construtor onde são inicializadas as pilhas.
- . Possui uma função iniciar jogo que só vai iniciar o jogo se a torre1 estiver vazia, se estiver vazia vai preencher ela com valores aleatórios e imprimir.
- . Possui uma função movimentação do tipo boolean, com parâmetros pilha origem e pilha destino. Onde vai ser realizado as movimentações dos valores das pilhas.
- . Possui um método menu1, que vai exibir as opções para o usuário escolher.
- . Possui um método menuMovimentação, que vai validar a opção escolhida pelo usuário e realizar as movimentações corretas entre as torres dependendo da opção escolhida pelo usuário. As opções fornecidas ao usuário estão no método açãoMovimentação.
- . Possui o método menuInicial, o qual, valida a opção escolhida pelo usuário no menu1.
- . Método validaFimDeJogo: Método que aparece após o usuário ou o computador terminar as jogadas, exibe uma mensagem de fim de jogo e em quantas jogadas foi realizada a ordenação, após isso imprime o estado final das pilhas.
- . Método ImprimirPilhas: Mostra a situação das torres 1, 2 e 3 após cada movimento realizado.
- . Método soluçãoAutomatica: Possui uma variável automática que recebe o método movimentação automática e realiza as movimentações para preencher automaticamente as torres com as pilhas.
- . Método MovimentaçãoAutomática: Exibe opções para o usuário, permitindo que ele digite números para informar se deseja realizar solução automática ou não.

. Classe Main:

- . Apresenta um scanner para o usuário informar o tamanho das pilhas
- . A classe jogo é instanciada e chama os métodos iniciar jogo, menu inicial e solução Automática.

Mauro João Mendes Vidal

https://github.com/Mauro-vidal/TDE_2_EstruturaDeDados

https://youtu.be/H3G_qddVnDI