

O presente código utilizou as estruturas de dados estudadas em sala de aula, pilha (o primeiro colocado é o último a sair), fila (o primeiro a entrar é o primeiro a sair), array (conjunto de dados com tamanho definido), lista encadeada simples (conjunto de dados vinculados por um nó com um espaço para armazenar uma referência da localização na memória onde o próximo elemento da lista, ou o anterior, se encontra) e lista encadeada dupla (conjunto de dados vinculados por um nó com um espaço para armazenar a referência da localização de memória onde se encontra o próximo elemento da lista e outro espaço para armazenar a referência da localização de memória onde se encontra o elemento anterior). Para rodar o código de forma mais didática foi implementado um menu com dados pessoais, como nome e cpf. O programa seguiu a lógica de colocar cada classe representando cada estrutura, pilha, fila, array, lista encadeada simples e lista encadeada dupla, com as aplicações de suas respectivas funções. As funções recebem, armazenam, ordenam, mostram e removem os dados fornecidos para as estruturas de forma prática para que o usuário entenda como funciona a estrutura das classes e subclasses. As complexidades de tempo e espaço do código variam entre $O(1)$ e $O(n)$ para todas as estruturas que estão sendo utilizadas. Durante o desenvolvimento do trabalho encontramos dificuldades para integrar de maneira eficiente a utilização de todas estruturas básicas pedidas no trabalho. A utilização das classes pilha e fila foram um desafio especial para a implementação do código pois funcionam de maneira bastante distinta quando se trata da entrada e retirada de dados, o que traz uma dificuldade maior para a funcionalidade fluida do código escrito. Entretanto, mesmo com as dificuldades descritas o código roda de forma eficiente com a entrada de dados do usuário no menu.