

(TAD_opac_12) Problema: Neste exercício você deverá implementar um algoritmo de busca chamado busca binária. A especificação completa das funções a serem utilizadas pode ser encontrada no arquivo "array.h". É importante observar que o arquivo "array.h" não deve ser modificado, uma vez que ele define a especificação precisa do problema que deve ser resolvido. Além disso, é crucial enfatizar que você será responsável pela manipulação correta da memória durante a execução do programa. Isso inclui a alocação dinâmica e a liberação de memória conforme necessário, garantindo que não haja vazamentos de memória.

A estrutura Array deverá possuir os seguintes atributos:

- Um vetor alocado dinamicamente para armazenar os números;
- Uma variável para armazenar o tamanho do vetor.

A busca binária é um algoritmo eficiente para encontrar um elemento em um conjunto de dados ordenados. O processo funciona da seguinte maneira:

1. Inicialmente, divida o conjunto de dados ao meio e compare o elemento desejado com o elemento do meio.
2. Se o elemento desejado for igual ao elemento do meio, a busca termina, e o elemento é encontrado.
3. Se o elemento desejado for maior do que o elemento do meio, descarte a metade esquerda do conjunto de dados e repita o processo na metade direita restante.
4. Se o elemento desejado for menor do que o elemento do meio, descarte a metade direita do conjunto de dados e repita o processo na metade esquerda restante.
5. Repita os passos 1 a 4 até encontrar o elemento desejado ou concluir que o elemento não está presente no conjunto de dados.

A busca binária é altamente eficiente, pois reduz o espaço de busca pela metade a cada iteração, tornando-o adequado para conjuntos de dados ordenados. No entanto, é importante observar que o algoritmo só funciona corretamente se o array estiver ordenado. Portanto, antes de iniciar a busca, você deve ordenar o array.

Definição dos formatos de entrada e saída:

Entrada: Uma sequência de números inteiros, terminada por um número seguido de um ponto ".". Ex: "1 2 3." Em seguida, será lido um número que será o elemento a ser buscado no vetor usando busca binária.

Saída: O índice do elemento a ser buscado. Note que ao ordenar o array, é provável que o índice dos elementos sejam alterados. Caso o elemento não seja encontrado, deverá ser imprimido na tela o índice -1.

Ver exemplos de formato de entrada e saída nos arquivos fornecidos com a questão.