

ATIVIDADE 1

PARTE 1 - BUSCA LINEAR C/ VETORES

Desenvolva um programa capaz de ler um array de n valores (a ser definido pelo usuário). Após a leitura, o programa deve solicitar ao usuário que digite uma chave que será buscada dentro do vetor. A seguir, o programa deve exibir o vetor inteiro (utilize o procedimento `exibe_vetor`) e, se a chave tiver sido encontrada, seu índice. Se a chave não tiver sido encontrada, exiba apenas uma mensagem dizendo que a chave não foi encontrada.

É obrigatório o desenvolvimento das seguintes subrotinas:

- **le_vetor** - Função responsável pela leitura dos valores de um vetor até que o valor -100 seja lido OU o vetor chegue no tamanho máximo definido pela constante `MAX_ARR`. Deve receber uma referência ao vetor a ser preenchido e retornar a quantidade de valores preenchidos pelo usuário.
- **busca_linear** - Função responsável pela implementação do algoritmo de busca linear. Deve receber uma referência ao vetor, seu tamanho e a chave a ser buscada. Seu retorno deve ser -1 caso não ache nada ou o índice da chave, caso ela seja encontrada.

Exemplos:

```
Digite o vetor: 2 4 100 99 -5 -100
Chave a ser buscada: 99
[2, 4, 100, 99, -5]
Chave 99 encontrada na posição 3.
```

```
Digite o vetor: 3 10 -100
Chave a ser buscada: 50
[3, 10]
Chave 50 não foi encontrada.
```

PARTE 2 - BUSCA COMBINADA C/ MATRIZES

Desenvolva um programa capaz de ler uma matriz de dimensões `[TAM_L][TAM_C]` (são constantes no código. **NÃO MODIFIQUE!**). Após a leitura, o programa deve solicitar ao usuário que digite uma chave que será buscada dentro da matriz. Após isso, a matriz deverá ser exibida (utilize o procedimento `exibe_matriz`).

A busca na matriz deverá ser realizada linha a linha. Se a linha estiver ordenada em ordem crescente, deve-se utilizar o algoritmo de busca binária. Se a linha não estiver ordenada, deve-se utilizar o algoritmo de busca linear. A escolha de algoritmo para cada linha da matriz deve ser exibida durante a execução do programa. A seguir, o programa deve exibir os índices (`[linha, coluna]`) onde a chave foi encontrada, caso ela tenha sido encontrada. Se a chave não tiver sido encontrada, exiba apenas uma mensagem dizendo que a chave não foi encontrada.

É obrigatório o desenvolvimento das seguintes subrotinas:

- **le_matriz** - Procedimento responsável pela leitura dos valores de uma matriz de tamanho `[TAM_L][TAM_C]`. Deve receber uma referência à matriz a ser preenchida.
- **busca_linear** - Função responsável pela implementação do algoritmo de busca linear. Deve receber uma referência ao vetor e a chave a ser buscada. Seu retorno deve ser -1 caso não ache nada ou o índice da chave, caso ela seja encontrada.
- **busca_binaria** - Função responsável pela implementação do algoritmo de busca binária. Deve receber uma referência ao vetor e a chave a ser buscada. Seu retorno deve ser -1 caso não ache nada ou o índice da chave, caso ela seja encontrada.
- **verifica_ordenacao** - Função responsável pela verificação da ordenação de um vetor em ordem crescente. Recebe como parâmetro uma referência ao vetor e deve retornar 0, caso não esteja ordenado e 1, caso esteja.

Exemplos:

```
Digite a linha 0: 1 2 3 4 5 6
Digite a linha 1: 7 8 9 10 11 12
Digite a linha 2: 13 14 15 16 17 18
Valor a ser buscado: 10
[[1, 2, 3, 4, 5, 6],
 [7, 8, 9, 10, 11, 12],
 [13, 14, 15, 16, 17, 18]]
Linha 0 ordenada. Verificando por busca binária.
Linha 1 ordenada. Verificando por busca binária.
Chave 10 encontrada em [1, 3].
```

```
Digite a linha 0: 10 14 11 2 3 4
Digite a linha 1: 5 6 7 8 9 10
Digite a linha 2: 40 3 2 11 20 3
Valor a ser buscado: 11
[[10, 14, 11, 2, 3, 4],
 [5, 6, 7, 8, 9, 10],
 [40, 3, 2, 11, 20, 3]]
Linha 0 desordenada. Verificando por busca linear.
Chave 11 encontrada em [0, 2].
```

```
Digite a linha 0: 1 2 3 4 5 6
Digite a linha 1: 7 8 9 10 11 12
Digite a linha 2: 13 14 15 16 17 18
Valor a ser buscado: 20
[[1, 2, 3, 4, 5, 6],
 [7, 8, 9, 10, 11, 12],
 [13, 14, 15, 16, 17, 18]]
Linha 0 ordenada. Verificando por busca binária.
Linha 1 ordenada. Verificando por busca binária.
Linha 2 ordenada. Verificando por busca binária.
Chave 20 não foi encontrada.
```

