Lab 8 search

Objective

1. practice various searching algorithms
2. strengthen array and linked list processing

Overview

Write a menu driven search program implementing the following:

```
do {
     cout<<"\nChoose your search type:";
     cout<<"\n1. Arrays: Sequential Search without recursion";
     cout<<"\n2. Arrays: Sequential Search with recursion";
     cout<<"\n3. Ordered Arrays: Binary Search without recursion";
     cout<<"\n4. Ordered Arrays: Binary Search with recursion";
     cout<<"\n5. Linked List: Search without recursion";
     cout<<"\n6. Linked List: Search with recursion";
     cout<<"\nEnter 0 to exit.\nYour choice: ";
     ...
     cout<<"\nSpecify the element to be searched for: ";
     ...
} while(i!=0);
```

Print the list before selecting the element to search for.
The search should return the position of the element (starting at zero). If the element is not in the list, the function should return a negative one (-1).
You may find it useful to create helper functions to create filled linked lists, filled arrays, and sorted arrays.

Sample output:

```
Choose your search type:
1. Arrays: Sequential Search without recursion
2. Arrays: Sequential Search with recursion
3. Ordered Arrays: Binary Search without recursion
4. Ordered Arrays: Binary Search with recursion
5. Linked List: Search without recursion
6. Linked List: Search with recursion
Enter 0 to exit.
Your choice: 1

Specify the number of elements to be searched: 10

Elements added to the array: 41 67 34 0 69 24 78 58 62 64

Specify the element to be searched for: 62
Element index: 8
```

Discussions

- if you're not comfortable with the exercise, you can use integer as your list elements.
- I would prefer that you use stock objects as your list elements if you're up to it.

Comments

You do not have permission to add comments.