Linked List Implementation:

1) Given a definition of the list implemented using linked list:

```
typedef struct node {
  char elem;
  struct node *link;
}*charList;   //List datatype
```

Activity:

A) Declare a variable of type charList. How many bytes is allocated to variable of charList?
Assume that after the declaration, there are statements that will populate the list with 3 elements.
Draw the list with 3 elements. For each variable, draw a box and label the box with a name, value, and address.

B) Given a function specification:
Given a list and an element X, function findElem() will return true if element X is in the list; otherwise return false.

Note: The header file stdbool.h has a bool data type with values: true and false

Constraints: Only 1 return statement
No break and continue statements

Do the following steps to better understand functions and how to create them:

1) Write an appropriate function header of function findElem().
2) Write an appropriate function call. Before the call, declare the variable/s used in the call and initialize the variable/s if necessary. Note: Do not pass garbage values to the called function.
3) Assume that the function call in #2 is in main() function, draw the execution stack (call stack) representing the call. For each variable, draw a box and label it with name, value, and address. Note: You can use arbitrary addresses such as A100, B100, etc
4) Write the code of the function findElem().
5) Simulate the function using the following test cases:
    a) the list is empty
    b) the list is not empty and element X is in the list
    c) the list is not empty and element X is not in the list