PROGRAMMING EXERCISE 6 (Linear Search)

The following is a pseudocode algorithm for a linear search function, which takes two parameters **SEARCHITEM** and **FRUIT**, and returns the index (if item is found) or -1 (if item is not found).

```
FUNCTION LinearSearch(string SEARCHITEM, array FRUIT)
BEGIN
   set MAXSIZE to be maximum size of FRUIT
   set element_found to FALSE
   set index to 1
   DOWHILE (NOT element_found) AND (index ≤ MAXSIZE)
        IF FRUIT(index) = SEARCHITEM THEN
           set element_found to TRUE
        ELSE
           index = index + 1
        ENDIF
   ENDDO
   IF element_found = TRUE THEN
      return index
   ELSE
      return -1
   ENDIF
ENDFUNCTION
```

Task 1

Write code for the above function using appropriate annotation.

Evidence 1: Your program code for **`LinearSearch`** function. **[5]**

Task 2

Write a **`main`** function that calls the above **`LinearSearch`** function using the following specifications:

- Request for user to enter string to be searched, **`SEARCHITEM`**
- Type in this sample array data:

  ```
  FRUIT = ['banana', 'durian', 'apple', 'lemon', 'papaya',
  'strawberry', 'honeydew', 'strawberry', 'honeydew']
  ```

- Pass user input **`SEARCHITEM`** and **`FRUIT`** array as parameters into function
- Receive return value from function and output on screen **either *position of the item*** (if item is found in array) or message "***`Item is not found in array`***" (if item is **not** in array).

Evidence 2: Your program code for main function. **[8]**

Evidence 3: Screenshots for running the program code by searching for `'papaya'` and `'watermelon'`. **[2]**