Submitted by Hardik Bishnoi

NUID : 002807991

Question 1.2

Show the Correctness proof for linear search algorithm determining the correctness of the loop invariant at initialization (before the execution), maintenance (during the execution) and termination (after the execution).

```python
def linearSearch(myList, item):
    for items in myList:
        if (item == items):
            return True;
    return False
```

DISCLAIMER

Below, I try an approach which may be uncommon. Usually proofs by loop invariant are made through assumption of the element existing in the array, however - I want to probe if the converse holds true. My answer is based around the assumption that instead of finding `item`, the algorithm does **not** find it (in a contradictory sense). I hope you find my answer below convincing.

Defining Variables

Let the current index of 'items' when searching in myList be i.
Let the number of elements in myList be n.

Loop Invariant

If at the start of next iteration, if `item` is not present in `myList` then it is also not present in the subarray `myList[0:i]` (in line 2, created for searching for item through comparison)

Initialization:

When we initialize `items` in the loop, the subarray `myList[0:i]` is empty since no comparisons have been made yet. Hence it follows that the element `item` is not present in the subarray at this point, so the case of initialization is proved.

Maintenance:

At the start of the next iteration, if `item` is not present in `myList`, then it is not present in `myList[0:i]`. If by checking through comparison in line 3 (`item == items`) we do find a match, the algorithm will terminate with a return of `True`. If `item != items`, the loop invariant is upheld since at this point $item \in myList[0:i]$.

Termination:

Upon termination condition `myList[i] > n` where n is the length of the array `myList`. Since the algorithm will run from index 0 to $n-1$, and on not finding `item == items` to be True at $n-1$, the loop will increment the value of i by 1 hence it will become $(n-1) + 1 = n$. So `item` is not present in the array `myList`, and the last line returns `False`. The loop invariant holds true for termination because `item` is not present in `myList[0:i]`.

Thank you.