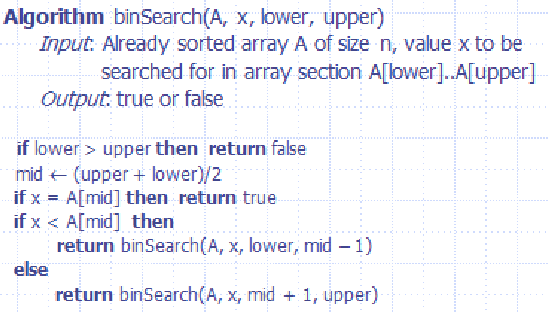
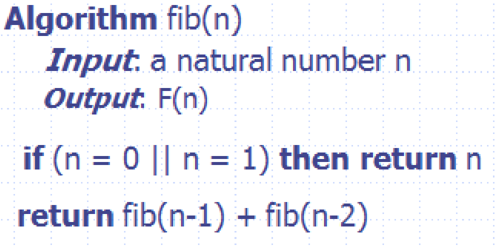
Lab 9

1. Below, the BinarySearch and Recursive Fibonacci algorithms are shown. In each case, what are the subproblems? Why do we say that the subproblems of BinarySearch *do not overlap* and the subproblems of Recursive Fibonacci *overlap?* Explain.



```
Algorithm binSearch(A, x, lower, upper)
    Input: Already sorted array A of size n, value x to be
           searched for in array section A[lower]..A[upper]
    Output: true or false

if lower > upper then return false
mid ← (upper + lower)/2
if x = A[mid] then return true
if x < A[mid] then
    return binSearch(A, x, lower, mid – 1)
else
    return binSearch(A, x, mid + 1, upper)
```



```
Algorithm fib(n)
    Input: a natural number n
    Output: F(n)

  if (n = 0 || n = 1) then return n

  return fib(n-1) + fib(n-2)
```

2. Consider the following instance of the Edit Distance problem: EditDistance(“maple”, “kale”). Taking the iterative dynamic programming approach to solve this problem, fill out the values in the table.

D	“”	“k”	“ka”	“kal”	“kale”
“”					
“m”					
“ma”					
“map”					
“mapl”					
“maple”					

3. https://leetcode.com/problems/climbing-stairs/description/
4. https://leetcode.com/problems/longest-common-subsequence/description/