

# Ronaldo marques



**ASSIGN  
BUSTER**

Ronaldo Marques COSC-2436-73426 15/03/2013 Project 2 5. 2-2 A) A :

Array[1.. n] I: Integer // Temp variable X: integer // Random integer between

1 and ' n' N: Integer // Size of array A Count: Integer // counts how many

elements has been searched CheckedA: Array[1.. n] // it will keep track of

index that was already checked Function int Random-Search(A, x) //Initialize

variables For i := 1 to n CheckedA[i] = false N = A. Length // gets A length

Count := 0 While (count < n) I := Random(1, n) // uses the library function to

get a a random integer between 1 and n //assigns it to I If (not

CheckedA[I]) // check if the record has been searched before count++

//increment count by 1 CheckedA[ I ] := true // set the flag to show this record

has been searched if (A[I] = x) return I // if x is found in A[I] the function

returns I and terminates return -1 // if the while loop exit without returning a

value it means the value //was not found on A[1.. n], so we return -1 to show

that no record was found. B) Expected number of indices into A is a

geometric random variable with expectation of  $1/p$  and  $p$  equal to  $1/n$  to be

right. So,  $E[X] = n$ . C) Just like the last problem but not with probability equal

$k/n$   $E[X] = 1/p = n/k$  D) In order to the function exit with -1 value in case if no

result was found, all the items of CheckedA array must be set to true and the

count variable must be equal to the number of elements on A (n). Since I is

picked randomly and it can be picked multiple times it would have to be  $E(X)$

$= n \ln n + O(n)$ .