/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* File : Lab3\_3.3.doc

\* Author : BOGGAVARAPU SAI RAVITEJA

\* Version : 1.0

\* Date : 2 February 2018

\* Description : Regarding answers to Alogorithm

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

3.3

ALGORITHM Find(Matrix[0..n-1,0..n])

//Input: An n by n+1 matrix Matrix[0..n-1,0..n] of real numbers

for index<- 0 to n-2 do

for nextindex<- index+1 to n-1 do

for tempindex<- index to n do

Matrix[nextindex,tempindex]Matrix[nextindex,tempindex]-

Matrix[index,tempindex] \* Matrix[nextindex,index] / Matrix[index,index]

Ans.

Here we see nested loops in the algorithm

so here we can see that innermost loop gets executed n+1 times for the first i=0

then the middle loop gets executed n-1 times for i=0

And if we go to next iteration innermost loop gets excuted n times

The middle loop gets iterated n-2 times

This gets repeated.

so the time efficiency becomes,

O((n+1)\*(n-1) + (n)\*(n-2)) = O( n^2-1 + n^2-2n ) = O(n^2)[This is approximated for larger values of n)]