

# CSN-261

## Assignment 2

Name: Deepanshu  
Matia

Enrolment: 18114018

## Problem 1:-

Write a program transpose.c that takes n, a, b, inputfile.txt in argv[1], argv[2], argv[3], and argv[4], respectively, applies the above encryption; and writes the result to outputfile.txt. Further, write a program inverseTranspose.c that decrypt the outputfile.txt and result in a new file named decryptedOutputfile.txt. Finally, write a program compareFiles.c to find the equivalence between the inputfile.txt and decryptedOutputfile.txt files.

```
[Deepanshus-MacBook-Air:P1 deepanshumatia$ gcc -o transpose transpose.c
[Deepanshus-MacBook-Air:P1 deepanshumatia$ time ./transpose 5 3 2 Sample_testcase_2.txt

real    0m0.010s
user    0m0.002s
sys     0m0.003s
[Deepanshus-MacBook-Air:P1 deepanshumatia$ gcc -o inversetranspose inversetranspose.c
[Deepanshus-MacBook-Air:P1 deepanshumatia$ time ./inversetranspose 5 3 2 outputfile.txt

real    0m0.011s
user    0m0.003s
sys     0m0.006s
[Deepanshus-MacBook-Air:P1 deepanshumatia$ gcc -o comparefiles comparefiles.c
[Deepanshus-MacBook-Air:P1 deepanshumatia$ time ./comparefiles Sample_testcase_2.txt decryptedoutputfile.txt
The files match successfully!

real    0m0.009s
user    0m0.002s
sys     0m0.004s
Deepanshus-MacBook-Air:P1 deepanshumatia$
```

## Algorithms used :-

Arrays have been traversed linearly to access data.  
Files have also been accessed to take input.

## Data Structures used :-

1D array has been used.

## Problem 2:-

Write a C program, MAT.c to represent any region (in image array representation), into its quadtree form.

```
Last login: Wed Aug  7 12:43:43 on ttys000
[Deepanshus-MacBook-Air:~ deepanshumatia$ cd /Users/deepanshumatia/Desktop/CSN-261\ L2/P1
Deepanshus-MacBook-Air:P1 deepanshumatia$ gcc -o MAT MAT.c -lm
[Deepanshus-MacBook-Air:P1 deepanshumatia$ time ./MAT
1 1 1 1 2 2 3 3
1 1 1 1 2 2 3 3
1 1 1 1 4 4 5 5
1 1 1 1 4 4 5 5
6 6 7 8 13 13 14 14
6 6 9 10 13 13 14 14
11 11 12 12 15 16 19 19
11 11 12 12 17 18 19 19
Leafnode      Bitvalue      Level
1              0              1
2              0              2
3              0              2
4              1              2
5              1              2
6              0              2
7              0              3
8              1              3
9              1              3
10             1              3
11             0              2
12             1              2
13             1              2
14             1              2
15             1              3
16             1              3
17             1              3
18             0              3
19             0              2

real    0m0.010s
user    0m0.003s
sys     0m0.004s
Deepanshus-MacBook-Air:P1 deepanshumatia$
```

## Algorithms used:-

Recursion has been used to create the maximal array.

## Data Structures used :-

Quadtree and 2D arrays have been used to implement the program.