Towers of hanoi (with TC)

```
#include<stdio.h>
#include<time.h>
int TOH(int, char, char, char);
int main()
{
int n;
clock_t t;
t = clock();
printf("Enter no of plates : ");
scanf("%d", &n);
int c = TOH(n, 'A', 'C', 'B');
printf("\n");
printf("Total no of moves : %d\n", c);
t = clock()-t;
double timetaken = ((double)t)/CLOCKS_PER_SEC;
printf("Algo took %f seconds to execute"
       ,timetaken);
return 0;}
```

```
int TOH( int n, char first, char third,
         char second)
{ int count;
if(n>0) {
count = TOH( n-1, first, second, third);
printf("Move disk %d from peg %c to
        peg %c \n", n, first, third);
    count++;
count = count + TOH( n-1, second, third, first);
}
return count;
}
```

DFS with TC :—

```
#include<stdio.h>
#include<conio.h>
#include<time.h>

void dfs(int);
int G[10][10], visited[10], n;
void main() {
int i,j;
clock_t t;
t = clock();
printf("Enter no of vehicles");
scanf("%d", &n);
printf("\nEnter adjacency matrix");
for(i=0; i<n; i++)
{for(j=0; j<n; j++)
{ visited[i]=0;
scanf("%d", &G[i][j]);}}
```

```
for(i=0;i<n;i++)
visited[i]=0;
DFS(0);
t = clock() - t;
double timetaken = ((double)t)/CLOCKS_PER_SEC;

printf("ALGO took %f sec to execute\n",timetaken);
}

void DFS(int i)
{ int j;
printf("\n%d",j);
visited[i]=1;
for(j=0;j<n;j++)
if(!visited[j] && G[i][j]==1)
DFS(j);
}
```