 # means Comment Out.

#The MCL algorithm follows but modifies a little Figure15 that is proposed in Van Dongen's thesis, p.55.

MCL (G,e,r) {

G=G+I;
$T_1=T_G$;

for k=1,...,∞ {

$T_{2k}=Exp_e(T_{2k-1})$; # Expansion
$T_{2k+1}=\Gamma_r(T_{2k})$; # Inflation

# Starting cluster stage.
for i=1,...,n {
T_{2k+1} = = $[t_{ij}]$(i=1,2,…,m; j=1,2,…,m);
C_i={$[t_{ij}]$| for j=1,...,m{$[t_{ij}]$>0.1};};
}

Ending cluster stage.
$ClusterStage_k$={$C_k(1)$, $C_k(2)$, ..., $C_k(d)$};
If(T_{2k+1} is (near-) idempotent) break;
}

Below is the BMCL algorithm to divide large-sized core clusters made by the original MCL.

# Selecting Core Clusters,
if(Size($C_k(p)$) > 2*Standard Deviation(Size $C_k(j)$), then $CoreCluster_n = C_k(p)$;

Selecting the representative node for each cluster C_k .
$Representaive\_ClusterStage_k$ = {Max (Degree ($C_k(j)$)) | j=1,2,….,d};

Removing the representative node of the core cluster from the following two lists which are the arguments of the function Complement,
CC'_n= Complement($CoreCluster_n$, $Representaive\_ClusterStage_k$)
RCS'_k= Complement($Representaive\_ClusterStage_k$, $CoreCluster_n$)

The Function ExtractAdjacency(adjacency_matrix, {row_number,column_number})is to extract rows and columns of an adjacency matrix,
$CC'_n\_RCS'_k$ =ExtractAdjacency(G, {CC'_n, RCS'_k })

Tr means transposition of a matrix.
$PathNumbersMatrix_n$ =$CC'_n\_RCS'_k$ * Tr($CC'_n\_RCS'_k$);

Generating an adjacency matrix by setting all diagonal elements=0 and all non diagonal elements larger than 1 = 1.
$LatentAdjacencyMatrix_n$ =MakeAdjacencyMatrix($PathNumbersMatrix_n$);

#Repeat MCL,
MCL($LatentAdjacencyMatrix_n$);