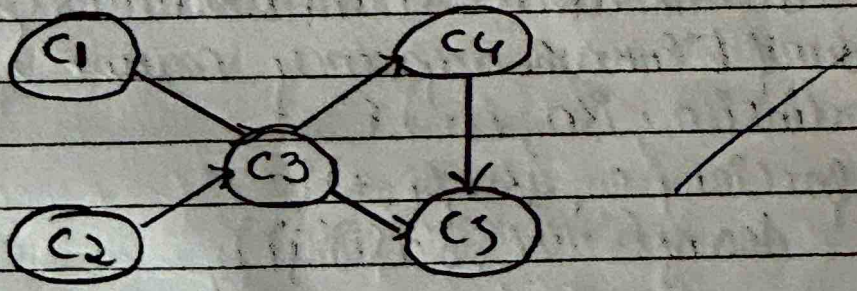
23/05/24 Topological sort using source Removal method

```
#include<stdio.h>
void topologicalSort(int a[][100], int n, int s[], int T[]){
  int indegree[100]={0};
  for(int y=0; y<n; y++){
    int sum=0;
  for(int i=0; i<n; i++)
      sum= a[i][j];
  indegree[j]=sum; }
  int top=-1;
  for(int i=0; i<n; i++){
   if (indegree[i]==0){
      top++;
      s[top]=i;
   }
  }
int T_index=0;
while(top!=-1){
  int u=s[top];
  top--;
  T[T_index++]=u;
  for(int v=0; v<n; v++){
     if(a[u][v]==1){
       indegree[v]--;
     if (indegree[v]==0){
        top++;
        s[top]=v;
     }
     }
  }
}
```

```
printf("topological order");
for(int i=0; i<n; i++){
    printf("%d", T[i]);
}
}
int main(){
int a[100][100], n, s[100], T[100];
printf("Enter the number of vertices");
scanf("%d", &n);
printf("Enter the adjacency matrix:\n");
for(int i=0; i<n; i++)
    for(int j=0; j<n; j++)
        scanf("%d", &a[i][j]);
topological sort(a, n, s, T);
return 0;
}
```

Output



Enter the number of vertices: 5
Enter the adjacency matrix
0 0 1 0 0
0 0 1 0 0
0 0 0 1 1
0 0 0 0 1
0 0 0 0 0
Topological order: C1, C2, C3, C4, C5
1 0 2 3 4

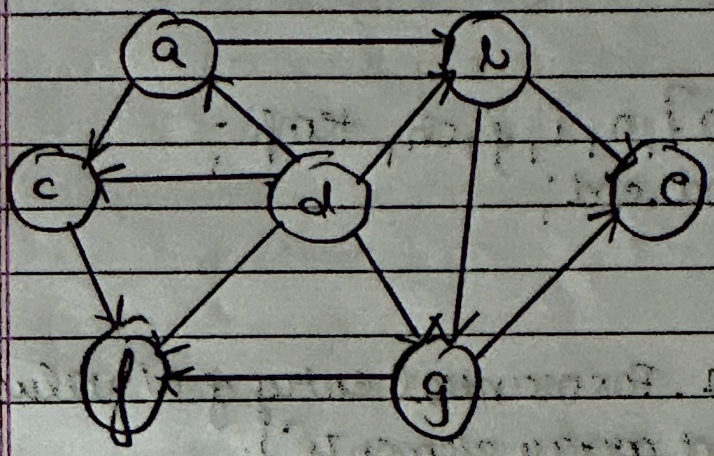
Topological sort using DFS

```
#include<stdio.h>
#include<stdlib.h>
#define MAX_VERTICES 100
int s[MAX_VERTICES] = {0};
int res[MAX_VERTICES];
int j = 0;
void DFS(int u, int n, int a[MAX_VERTICES][MAX_VERTICES])
{
  s[u] = 1;
  for(int v = 0; v < n; v++) {
  if (a[u][v] == 1 && s[v] == 0) {
    DFS(v, n, a);
  }
}

int main() {
  int n;
  printf("Enter the number of vertices");
  scanf("%d", &n);
  int a[MAX_VERTICES][MAX_VERTICES];
  printf("Enter the adjacency matrix");
  for(int i = 0; i < n; i++) {
    for(int j = 0; j < n; j++) {
      scanf("%d", &a[i][j]);
    }
  }
for(int u = 0; u < n; u++) {
  if (s[u] == 0) {
    DFS(u, n, a);
  }
}
printf("Topological order: ");
for(int i = j - 1; i >= 0; i--) {
  printf("%d", res[i]);
```

printf("\n");
return 0;
}

Output



Enter the number of vertices: 7
Enter the adjacency matrix:

	a	b	c	d	e	f	g
a	0	1	1	0	0	0	0
b	0	0	1	0	1	0	1
c	0	0	0	0	0	1	0
d	1	1	1	0	0	1	1
e	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0
g	0	0	0	0	1	1	0

Topological order: d a c b g f e
3 0 1 6 4 2 5

23/8/24