9.a) write a program to traverse a graph using BFS method.

```
START
READ number of vertices n
READ adjacency matrix adj[n][n]

INITIALIZE visited[] = 0
INITIALIZE queue
front = 0 , rear = -1

READ starting vertex s
queue[++rear] = s
visited[s] = 1.

WHILE front ≤ rear
v = queue[Front++]
 PRINT v
FOR i = 0 to n-1
  IF adj[v][i] = 1 AND visited[i] = 0
      queue[++rear] = i
       visited[i] = 1
    END IF
   END FOR
 END WHILE
 END
```

```
#include <stdio.h>
int graph[20][20], visited[20], n;

void BFS(int start){

  int queue[20], front=0, rear=0
  visited[start] = 1;
  queue[rear++] = start;

  while(front<rear){
    int node = queue[front++];
    printf("%d", node);

    for(int i=0; i<n; i++){
     if(graph[node][i]==1 &&
          !visited[i])
       {
         visited[i] = 1;
         queue[rear++] = i;
       }
    }
  }
}

int main(){
  int start;

printf("Enter number of vertices:");
scanf("%d", &n);

printf("enter adjacency matrix:\n");
```

```
for (int i=0 ; i<n ; i++)
  for (int j=0 ; j<n ; j++)
    scanf ("%d", &graph [i] [j]);

for (int i= 0; i<n; i++)
   visited [i] = 0;

printf (" enter starting vertex :");
scanf ("%d", &start);

printf (" BFS Traversal :");
BFS (start);
return 0;
}
```

output :

```
Enter number of vertices : 4
Enter adjacency matrix :
0 1 1 0
0 0 1 0
0 0 0 1
0 0 0 0

Enter starting vertex : 0
BFS Traversal : 0 1 2 3
```