

```

from collections import deque
directions=[[0,1],[1,0]]
n=int(input())
o=[list(map(int,input().split())) for _ in range(n)]
def bfs(a,b,c,d,p):
    q=deque()
    inq=set()
    q.append((a,b,c,d,p))
    inq.add((a,b,c,d,p))
    step=0
    while q:
        for _ in range(len(q)):
            e,f,g,h,length=q.popleft()
            if e==n-1 and f==n-2 and g==n-1 and h==n-1:
                return step
            else:
                for dx,dy in directions:
                    ne=e+dx;nf=f+dy;ng=g+dx;nh=h+dy
                    if 0<=ne<n and 0<=nf<n and 0<=ng<n and 0<=nh<n and
(ne,nf,ng,nh,length) not in inq and o[ne][nf]==0 and o[ng][nh]==0:
                        q.append((ne,nf,ng,nh,length))
                        inq.add((ne,nf,ng,nh,length))
                    if length==0 and 0<=e+1<n and 0<=g+1<n and o[e+1][f]==0 and o[g+1]
[h]==0 and (e,f,e+1,f,1) not in inq:
                        q.append((e,f,e+1,f,1))
                        inq.add((e,f,e+1,f,1))
                    if length == 1 and 0 <= f + 1 < n and 0 <= h + 1 < n and o[e ]
[f+1] == 0 and o[g][h+1] == 0 and (e, f, e, f+1,1) not in inq:
                        q.append((e, f, e, f+1, 0))
                        inq.add((e, f, e, f+1, 0))
                step+=1
    return -1
result=bfs(0,0,0,1,0)
print(result)

```