#Task 1

f=open("input1.txt","r")

lst1=[]

for line in f:

l=line

lst1.append(l.split())

lst=[]

for i in lst1:

x=len(i)-1

for j in i:

lst.append(j)

visited=[0]\*100

def DFS(S):

count=0

Q=[]

Q.append(S)

while Q!=[]:

n=Q.pop()

count+=1

visited[n]=1

child=[]

if n==len(lst)-1: # for last row last index

child=[]

elif n>=len(lst)-x-1: # for last row

if lst[n+1]=="Y":

child.append(n+1)

elif (n-x)%(x+1)==0: # for last column

if lst[n+x]=="Y":

child.append(n+x)

if lst[n+x+1]=="Y":

child.append(n+x+1)

elif n%(x+1)==0: # for first column

if lst[n+1]=="Y":

child.append(n+1)

if lst[n+x+1]=="Y":

child.append(n+x+1)

if lst[n+x+2]=="Y":

child.append(n+x+2)

else:

if lst[n+1]=="Y":

child.append(n+1)

if lst[n+x]=="Y":

child.append(n+x)

if lst[n+x+1]=="Y":

child.append(n+x+1)

if lst[n+x+2]=="Y":

child.append(n+x+2)

for i in range(len(child)):

if visited[child[i]]==0:

Q.append(child[i])

return(count)

totalcount=[]

for i in range(len(lst)):

if lst[i]=="Y" and visited[i]==0:

totalcount.append(DFS(i))

print(max(totalcount))

#------------------------------------------------------------------------------------------------------------------#

#Task 2

f=open("input2.txt","r")

lst1=[]

row=f.readline()

column=f.readline()

for line in f:

l=line

lst1.append(l.split())

lst=[]

for i in lst1:

x=len(i)-1

for j in i:

lst.append(j)

x=int(column)-1

visited=[0]\*100

def BFS(S):

time=1

Q=[]

Q.append(S)

while Q!=[]:

n=Q.pop(0)

visited[n]=1

child=[]

if n==0: # for first row first index

if lst[n+1]=="H":

child.append(n+1)

if lst[n+x+1]=="H":

child.append(n+x+1)

elif n==x: # for first row last index

if lst[n+x+1]=="H":

child.append(n+x+1)

if lst[n-1]=="H":

child.append(n-1)

elif n<=x: # for first row

if lst[n+1]=="H":

child.append(n+1)

if lst[n+x+1]=="H":

child.append(n+x+1)

if lst[n-1]=="H":

child.append(n-1)

elif n==len(lst)-x-1: # for last row first index

if lst[n-x-1]=="H":

child.append(n-x-1)

if lst[n+1]=="H":

child.append(n+1)

elif n==len(lst)-1: # for last row last index

if lst[n-x-1]=="H":

child.append(n-x-1)

if lst[n-1]=="H":

child.append(n-1)

elif n>=len(lst)-x-1: # for last row

if lst[n+1]=="H":

child.append(n+1)

if lst[n-1]=="H":

child.append(n-1)

if lst[n-x-1]=="H":

child.append(n-x-1)

elif n%(x+1)==0: # for first column

if lst[n+1]=="H":

child.append(n+1)

if lst[n-x-1]=="H":

child.append(n-x-1)

if lst[n+x+1]=="H":

child.append(n+x+1)

elif (n-x)%(x+1)==0: # for last column

if lst[n-1]=="H":

child.append(n-1)

if lst[n-x-1]=="H":

child.append(n-x-1)

if lst[n+x+1]=="H":

child.append(n+x+1)

else:

if lst[n-x-1]=="H":

child.append(n-x-1)

if lst[n+1]=="H":

child.append(n+1)

if lst[n-1]=="H":

child.append(n-1)

if lst[n+x+1]=="H":

child.append(n+x+1)

if child!=[]:

time+=1

for i in range(len(child)):

if visited[child[i]]==0:

lst[child[i]]="A"

Q.append(child[i])

totaltime.append(time)

aliens=[]

for i in range(len(lst)):

if lst[i]=="A":

aliens.append(i)

totaltime=[]

for i in aliens:

BFS(i)

print(f"{max(totaltime)-1} minutes")

humancount=0

for i in range(len(lst)):

if lst[i]=="H":

humancount+=1

if humancount==0:

print("No one survived")

else:

print(f"{humancount} survived")