suppose their are i points in G,

and for each point i,

we have pointlink(i){M[i,1],M[i,2]....M[i,i-1]}

if( their are more than three M in pointlink(i) = 1 and more than three M in pointlink(i) = 0 ),

add i int to list pointlist[], it takes O(n^2)

for each point in pointlist[], remove point i if i only linked two or less points that are in pointlist[] and also remove point i if i only not linked two or less points that are in pointlist[].

it takes O(n\*6n) = O(n^2)

for every point  i in pointlist[],use greedy to select points, if M(i,j) = 1,add j into i's list ,then add all the point with M(j,k) = 1, remove point j from pointlist[].

it takes  O(n^2)

after that, the largest list is the largest H, and if their isn't 7 points in that list, output "No such H can be found"

finally it takes O(n^2) +O (n^2)+O(n^2) = (n^2)