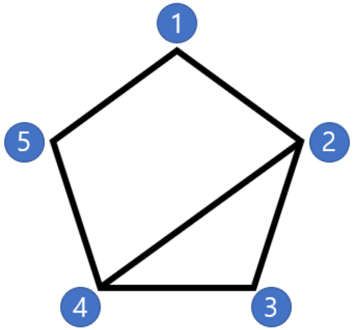
P10. Tumor

[홍원표 21701065]



문제 분석

n number of tumor cells (vertex), **b** number of blood vessels (edge), each vertex has its own weight, $\mathbf{w_i}$, and each edge is specified by linked vertices. Therefore, the whole Petri dish can be described as a **graph**. The goal of this problem is to find the **maximum weight** of tumor cluster, which is defined as a **complete subgraph**.($2 \le n \le 450$), ($2 \le b \le 900$)

문제 풀이

Since it is possible for **two-cell-cluster to be heavier than hundred-cell-cluster**, (ex, 100+100 > 1+1+…+1) there is no other way but to **find all sorts of complete subgraphs (clique)** in a given graph. A simple way to find all cliques is to **look for all combinations** of vertices and see if each vertex is connected to each other. This takes **2^n - 1 times**. However, we can upgrade this method if we **neglect the combinations that include an already known non-clique**. For example, if we know that (1-2-3) is not a clique, (1-2-3-4) can't be a clique either. Therefore, an improved method uses **recursions** to **add vertex one by one from previously made clique.** If (1-2) is a clique, it checks for (1-2-3) next, and so forth. If (1-2) is not a clique, it moves on to (1-3). Also, we could **neglect all decreasing order** such as (1-3-2), because the same combination, (1-2-3), would already have been checked before.

문제 풀이 분석 **[Time Complexity: ??? Space Complexity: $O(n^2)$]**

Using the method explained above, the highest time complexity for n vertices, 2^n-1, would be having n(n-1)/2 edges, which means the graph itself is complete subgraph. However, since the number of edges, b is given to be 2n at most, it will definitely be smaller than 2^n-1. However, accurate time complexity can't be found because even if n and b are not changed, if the edge links are different, the time complexity will be different. However, nC1, nC2 are definitely taken no matter how graph looks like. If the biggest clique has size x, it will have the time complexity of nC1 + nC2 + … + nCx-1, and then reduced number of nCx + … + nCn. The graph link availability, weight of each vertex, and etc must be saved. The biggest space needed is the graph link availability which is n^2,

Discussion

The problem stated that 'a blood vessel never crosses another blood vessel', but this did not make sense to me. I had to solve the problem without thinking about this line, but I still got the 'correct' mark. If anyone has an opinion on this statement, please let me know!

`

1	
1	2
1	2
	3
1	2
	4
1	2
	5
1	3
1	4
1	5
2	
2	3
2	3
	4
2	3
	4
	5
2	3
	5
2	4
2	4
	5
2	5