Week11-the second problem

We have learned about the category of Graph Theory’s problem. We're going to study a lot of different types of problems. Learning different methods is very useful for solving different problems.

In this week’s practical, I pick the second practice question. The problem name is AlmostBipartiteMatching. And the category of this question is Graph Theory.

For this question, I need to understand the meaning firstly. I will be given a graph like bipartite and calculate the maximum matching of that graph. Because the maximum matching of a graph is the largest possible set of edges in which no two edges share a common vertex. So, I use the if statement like this: if((nA%2==0 && nB%2==0)||(nA%2!=0 && nB%2==0) || (nA%2==0 && nB%2!=0)). Then, int l=edgesA.length. Next, I use the for loop and to set a variable like int i = 0, to define the condition for the loop to run (i must be less than the length of l and i++). The most important step is that I should use the if statement and the condition is (edgesA[i]%2==0 && edgesB[i]%2==0).

Class: AlmostBipartiteMatching
total_marks: 106
out_of: 106

There are some challenges in thinking how to use a better method to solve this type of problem. The benefit of practicing is to practice more problems and accumulate experience and master the method.