1 Problem 1

Algorithm 1: Job Finder Algorithm

Data: m: a list of jobs with associated payment values
Data: k: the maximum number of jobs to work
Result: A list of k jobs with the highest payment values

```
Function job_finder(m, k)
    // Sort the list of jobs in descending order
    m.sort(reverse = True) ;
    return m[0 : k] ;
end
```

1.1 Runtime Analysis

The most time-consuming operation in this algorithm is the sorting step, which takes $\Theta(n \log n)$ time, where n is the number of jobs.

1.2 Correctness Proof

Easy to read proof:

Suppose the optimal solution differs from ours, then at at least one point, they have different jobs in their "job basket".

But since our "job basket" has all the highest paying jobs, the different job(s) they have in their basket that arent in ours are going to be less than ours, so their total amount of money will be less than ours. Contradiction, this is not optimal! So our solution has to be optimal!

Formal math proof (not sure if its correct):

Proof by contradiction:

Let my solution be the set $S = m_{i_1}, m_{i_2}, ..., m_{i_n}$
Say $\exists$ some solution to the problem, the set σ that maximizes the amount of money you will receive and is distinct from S. Then at some point
But since $m_{i_1}, m_{i_2}, ..., m_{i_n} > s, \forall s \in S^c$, then $\sum_{j=1}^{n} m_{i_j} > \sum_{j=1}^{n} m_{f_j}$, where $m_{f_j} \in S^c$
Contradiction! Thus our solution is optimal.

2 Problem 2

I wasn't sure how to formalize this, but here it goes.

1. Find how many connections to other students each student has. If there are no students left, return the counter.

2. If there is no tie, remove the greatest connected student, and increase our counter by one. And go to step one.

a. If there is a tie, break into cases where you remove each of them, and go to step one for all cases. Remove the one whose (local) counter is the minimum, then increase the counter by one.

2.1 Time Complexity

$O(n^n)$, where n is the number of students (this is because the worst case is where none of the students are overlapping)

2.2 Correctness Proof

Not sure if this is correct:

Assume there is an optimal solution to the problem in which they pick a different person to testify at some point. Then they are removing someone with fewer connections, or someone (tied for connections) which results in a more disjoint group of people.

Case 1: (not sure if correct) They are getting rid of fewer people, so there are more people remaining, and at best they can now only tie our solution.

Case 2: If it results in a more disjointed group of people, then obviously its going to take more steps to get everyone testified.

Contradiction!