Homework 2: Q3

**Name:** Xinkai Lin, Xinkaili

*Don’t forget to input your list of collaborators and sources on* ***AutoLab****.*

**Please submit this file as a PDF.**

# **Algorithm Idea**

Begin Algorithm Idea: set n be the number of student s and engineers e, and m = n2. So that m > n will always be satisfied. In this algorithm, I will set each student meet with all the engineer possible. However, it will be ended up pairing them as (s1, en), (s2, en-1), ……, (sn, e1).

1. **Algorithm Details**

**Algorithm:** Algorithm1

Initially all student and engineer are free

for s ∍ n, 1….n //loop through all the student

for e ∍ n, 1….n //loop through all the engineer

if e is the last engineer

pair (s, e) //pair up each student and engineer as (s1, en), (s2, en-1), ……, (sn, e1)

else

student meet with engineer and leave itself

endif

endfor

endfor

1. **Proof Idea**

Begin Proof Idea: In this algorithm, we will be demonstrating an example of n=2, m=4. Whereas m>n. I’ll let the first student meet with all the engineer, if he is the last engineer they will match. However, student will meet with another engineer and do nothing. So on so forth, when it iterate all the student the result should be (s1, en), (s2, en-1), ……, (sn, e1), which no one gets stood-up.

1. **Proof Details**

Begin Proof Details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CSE Major | Slot 1 | Slot 2 | Slot 3 | Slot 4 |
| S1 | E1 | Free | E2 (truncate here) |  |
| S2 | Free | E1 (truncate here) |  |  |

Based on the schedule we can see that there is n (which is 2) number of students and engineers, and m (n2=4) slots. While we iterate through all the students, we check if he meets the last engineer which is E2 in this case. Before he has met E2, he will first meet with E1. However, he will just go talk to the E1 and not do anything until he met E2 and assigned with E1. For S2, since our n=2, we will pair S2 with En-1 which is E2 and not care about the rest.