CS 4820, Spring 2018	**Homework 8, Problem 1**
Name: Rongguang Wang	
NetID: rw564	
Collaborators: Siyuao Liu (sl2928); Yihao Chen (yc2288)	

(1) *(8 points)* At Cornell University, doctoral students in Computer Science can earn their Ph.D. either in Ithaca or at the new Cornell Tech campus in New York City. However all students initiate their study in Ithaca and complete some course requirements before potentially moving to New York City. This inspires the following course scheduling problem.

A department has p professors and $2p$ courses. Each professor teaches two courses, one per semester. Assume that professors are numbered $0, \ldots, p-1$ and courses are numbered $0, \ldots, 2p-1$ such that professor i teaches courses $2i$ and $2i+1$, for each $i = 0, \ldots, p-1$. There are s students (numbered $0, \ldots, s-1$) each of whom is trying to finish their remaining course requirements in the fall semester so that they can move to a different campus in the spring. The remaining course requirements for student j are fulfilled by taking at least $n(j)$ courses from a specified list $L(j)$.

The Impatient Grad Students Problem (IGSP) is the following decision problem: given the numbers $p, s, n(0), \ldots, n(s-1)$ (in binary) and the lists $L(0), \ldots, L(s-1)$, decide whether the courses can partitioned into the fall and spring semesters such that

1. each professor teaches one fall course and one spring course
2. for each list $L(j)$, at least $n(j)$ of the courses in $L(j)$ are scheduled in the fall semester.

Prove that IGSP is NP-complete.

Solution:

The problem is aiming to proof that the Impatient Grad Students Problem (IGSP) is NP-Complete. In IGSP, there are p professors, $2p$ courses which means that each professor i teach one course $2i$ in fall semester and one course $2i+1$ in spring semester. There are s students who need to take required number of courses in fall semester. Each student j should take at least $n(j)$ courses in fall semester from the course list $L(j)$. The numbers p, s, and $n(j)$ are binary numbers. The problem is that whether the courses can be partitioned into fall and spring semesters by enforcing the constraints that each professor teaches one fall course and one spring course, and there are st least $n(j)$ of the courses in $L(j)$ are scheduled in the fall semester for each list $L(j)$.

First, we need to prove that IGSP problem is in NP by describing a polynomial-time verifier. Given an instance of the problem, and a proposed schedule specifying the teaching professor and the teaching semester of the courses. The verifier tests that (a) each professor teaches one fall course and one spring course and (b) there are st least $n(j)$ of the courses in $L(j)$ are scheduled in the fall semester for each list $L(j)$. To test property (a), the professors should be traversed in $O(\log p)$ to check if each professor teaches one odd number course $(2i+1)$ and one even number course $(2i)$. To test property (b), an array should be constructed in $O(\log p)$to store one bit for each fall semester and spring semester specifying whether a course is teaches in that semester, then, in $O(\log s)$ time, one can run through each students course lists and look up those semesters in the array to check if the number of courses teaches in fall semester is at least $n(j)$. Thus, the verification time is in polynomial time which means IGSP problem is in NP.

Next, to prove that IGSP partition problem is NP-Complete, we reduce from 3SAT. Given an instance of 3SAT with n variables and m clauses, we construct an instance of the IGSP problem with n professors and m students. For each clause j containing three literals, the corresponding student need only choose one course from the three courses in the course list. In another word, the course number requirement $n(j) = 1$ for each student j and the size of each student course list $L(j)$ is 3. A literal such as x_i corresponds to the course being teach in fall semester by professor i. A negated literal such as $\bar{x_j}$ corresponds to the course being teach in spring semester by professor j. Generating the set of professors and set of students, and the list of student courses takes $O(m + n)$ time, so this is a polynomial-time reduction. To prove the correctness of the reduction, we must show two things.

1. If the 3SAT instance is satisfiable, then there is a partition of courses that satisfies all of the problem constraints. Given a satisfying truth assignment of the 3SAT formula, partition a course teaching by professor i in fall semester if x_i is true, and in spring semester if x_i is false. Clearly, this partition assigns each professor one course each semester, but not both. Furthermore, by our assumption that the truth assignment satisfies at least one literal in each clause, and by our construction of the students course number requirement constraint, the partition contains a course in at least one of the student js three courses from the course list $L(j)$, for every j.

2. If there is a partition of courses that satisfies all of the problem constraints, then there is a satisfying truth assignment of the 3SAT formula. Given a partition of courses, assign x_i the value true or false according to whether the course teaching by professor i is in fall or spring semester, respectively. By our construction of the course constraints of student j, we know that if the partition includes a course in student js course list $L(j)$, then the corresponding truth assignment satisfies at least one of the literals in clause j. Thus, we have constructed a satisfying truth assignment of the 3SAT formula, as desired.