The Outline of the Course Complexity Theory

1. Introduction to (time and space) Complexity through some well-known Problems
   - 1.1. Graph Reachability Problem
     - 1.1.1 How to present the problem?
     - 1.1.2 Non-deterministic and deterministic algorithm for the problem
     - 1.1.3 How many (?) time and space do we need to solve the problem?
   - 1.2 SAT Problem
     - 1.2.1 How many time and space do we need?
     - 1.2.2 Circuits

2. The Computational Model
   - 2.1 Coding
   - 2.1 Turning Machines (Deterministic and Non-deterministic)
   - 2.2 Universal Turing Machine
   - 2.3 Complexity Classes DTIME(f), NTIME(f), DSPACE(f), NSPACE(f)
   - 2.4 Complexity Classes and the number of the tapes of Turing Machines
   - 2.5 Universal Turing Machine and time (space) complexity

3. Basic Complexity Theorems
   - 3.1 Time (Space) Constructible Functions (WHY: see 3.7)
   - 3.2 Time Linear Speedup
   - 3.3 Space Compression
   - 3.4 Nondeterministic Time and Deterministic Space NTIME(f)$\subset$DSPACE(f)
   - 3.5 Nondeterministic Space and Deterministic Time NSPACE(f)$\subset$DTIME(2^{cf})
   - 3.6 Hierarchy Theorem (ALWAYS DIAGONALIZATION)
     - 3.6.1 Bounded Halting Theorem
     - 3.6.2 Time Hierarchy Theorem
     - 3.6.3 Space Hierarchy Theorem
   - 3.7 Gap Theorem and why constructible functions

4. QUESTION: Which One IS Proper?
   - 4.1 Complexity Classes: L, NL, P, NP, PSPACE, NPSPACE
   - 4.2 From 3.4 and 3.5, we know $L \subseteq NL \subseteq P \subseteq NP \subseteq PSPACE$
   - 4.3 From 3.6.3, we know $L \neq PSPACE$
   - 4.4 From 4.2 and 4.3, at least one of the subset relations must be proper.

5. $P =^{?} NP$: Part 1: The Complexity Class NP
   - 5.1 The Characterization of the class NP in terms of the class P
   - 5.2 Some NP Problems
     - 5.2.1 Connectivity versus Hamilton Path,

5.2.1 SAT and 3-SAT
5.3 Reducibility
5.3.1 Polynomial time Reducibility
5.3.2 Logarithm space Reducibility
5.3.3 Reducing Hamilton Path problem to SAT problem
5.4 NP-Completeness
5.5 Bounded Halting Problem is NP-Complete
5.6 Cook's Theorem: SAT is NP-Complete
5.7 3-SAT is NP-Complete
5.8 Hamilton Path is NP-Compete
5.9 Turing Reducibility
5.10 Oracle Machines
5.11 Is Co-NP Equal to NP?

6. PSPACE= NPSPACE (We Should Celebrate)
6.1 Reachability Problem (A good algorithm)
6.2 Immerman Theorem
6.3 Savitch's Theorem
6.4 PSPACE Complete Problems
6.5 NL Complete Problems
6.6 NL=CoNL (We Should Celebrate)

7. $P =^{?} NP$: Part 2: Polynomial Time Hierarchy
7.1 Oracle Nondeterminitic Machines
7.2 The Characterization of the hierarchy classes in terms of the class P
7.3 PH Complete Problems
7.4 Alternating Turing Machines
8. $P =^{?} NP$: Part 3: Inside NP
8.1 Incomplete Problems in NP
8.2 Relativization
8.3 One Way Functions
8.4 Polynomial Time Isomorphism
8.5 Berman-Hartmanis Conjecture

9. Introduction to Randomized Computation
9.1 Probabilistic Turing Machines
9.2 The Complexity Class BPP
9.3 BPP is in HP

Your Grade:
1. 20% exercises
2. 20% final project
3. 60% final exam