

**CSC2401F Instructor: A. Borodin**

**Text: “Introduction to the Theory of Computation: Second Edition” by Michael Sipser**

CSC2401 is an introductory level graduate course which is appropriate for *all* graduate students in Computer Science, as either a breadth course for those not primarily interested in theoretical aspects of computing or as a foundational course for those students who may be planning to focus on theoretical issues. The main part of the course will be an introduction to complexity theory where we discuss uniform and non-uniform models of computation, time and space complexity classes, complexity hierarchies, reductions and completeness, randomization in time and space computations, tradeoff issues and provably hard problems. In the latter part of the course we may discuss (time permitting) one or two relatively new results in complexity theory such as time-space lower bounds, derandomization results, PCP proofs and inapproximation results. Students can access previous versions of this course as perhaps the best indicator of what we intend to be doing. See [www.cs.toronto.edu/~bor/2401f01](http://www.cs.toronto.edu/~bor/2401f01), [2401f02](http://www.cs.toronto.edu/~bor/2401f02), [2401f03](http://www.cs.toronto.edu/~bor/2401f03), [2401f07](http://www.cs.toronto.edu/~bor/2401f07) and [www.cs.toronto.edu/~rackoff/2401f00](http://www.cs.toronto.edu/~rackoff/2401f00).

Here follows a very tentative schedule of topics.

- The Basic Topics

1. The multitape Turing machine model; time and space complexity measures and complexity classes. Why the computational model may not matter. (Sipser Chapter 3 and Section 7.1)
2. The polynomial time complexity thesis. (Sipser Section 7.2)
3. The basic diagonalization technique; time (space) complexity class hierarchies. (Theorems 9.3 and 9.10 in Sipser)
4. Non deterministic computation. (Theorem 3.16 and Section 7.3 of Sipser)
5. The basic inclusions :

$$NC^1 \subseteq L \subseteq NL \subseteq P \subseteq NP \subseteq PSPACE \subseteq \bigcup_k DTIME(2^{n^k}) = EXP$$

6. Logspace reducibility; *STCON* (directed graph s-t connectivity) is complete for *NL* wrt logspace reducibility.  
 $NSPACE(s(n)) \subseteq DSPACE(s(n)^2)$ . The relation between space and parallel time. (Sipser Chapter 8)
7.  $\overline{STCON} \in NL$  and hence  $NL = coNL$ . (Sipser Section 8.6)
8. *BHP* (bounded halting problem), *CIRCUIT – SAT*, *SAT*, etc. etc. are complete for *NP*. (Section 7.4 of Sipser and Kozen Lecture 6)
9. Fixed Parameter Tractable Problems
10. Boolean circuit complexity and non-uniform complexity measures of size and depth. Why are circuit lower bounds so difficult. Algebraic circuit complexity. (Sipser Section 9.3)

11. Provably hard natural languages.  
 $TRE$  = totality of regular expressions (resp.  $TERE$  = totality of extended regular expressions) is complete for polynomial (resp, exponential) space.
12. Randomized computation; symbolic determinant; primality testing in  $P$ .  
 $BPP \subseteq POLYSIZE$ . (See Sipser Section 10.2 and Kozen Supplementary Lecture C)

- Some Possible Additional Topics

1.  $\#P$  (counting problems) and  $IP$  (interactive proofs).  $\#P$  in  $IP$ ;  $IP = PSPACE$ . (Sipser Section 10.4 and Kozen Lectures 15,16,17). Probabilistically checkable proofs, (Kozen Lecture 18)
2. Indirect diagonalization and time-space bounds.  
 $SAT \notin DTISP(n^{\sqrt{3}}, n^{o(1)})$ .
3. The polynomial hierarchy; alternative characterizations of  $NP$  and  $\Sigma_k^P$ .

Alternating Turing machines

$ATIME(\log n) = uniformNC^1$  (log parallel time);  $ASPACE(\log n) = P$ .  
(Sipser Section 10.3 and Kozen Lectures 9,10)

4. Pseudo random generators.  $P = BPP$  if there exists  $L \in E$  such that  $L$  has circuit complexity  $2^{\Omega(n)}$ .
5.  $USTCON$  (undirected graph s-t connectivity) in  $L$ .