The fifth exam will be given in registrar-assigned timeslot on Monday, May 11. **Take care of any necessary business before class** *so that you do not need to leave the room during the exam.*

Note that only your dean can reschedule this exam. If you have three exams on the same day, you may work with the instructor and your dean to reschedule one of them. To reschedule this exam, you must be in touch with me **by the end of the day Wednesday 5/6**.

This exam primarily covers the remainder of chapter 4 (section 4.6) and all of chapter 5 (sections 5.1–5.3). It is designed to be similar to the midterm exams in length (approximately an hour), though you will have the full three hour time period to work on it if you wish.

Many of the questions on the exam will be similar to problems on the homeworks. There may be some "short essay" questions that ask you to define something, or discuss something, or explain something, and so on.

In addition, there will be an essay question (3-4 paragraphs) asking you to bring together various things that have been covered in the course and to reflect on what you have learned about the nature of computation. You should be prepared to write about things like logic and logic circuits, different types of automata and languages, and ideas about computability and uncomputability.

Terms and ideas from the last part of the course that you should be familiar with:

- general grammars
- derivations using a general grammar
- how a general grammar generates a language
- Turing machine
- transition diagram for a Turing machine; table of rules for a Turing machine
- how a Turing machine operates
- how a Turing machine computes a function
- evidence that Turing machines are general-purpose computing devices
- Turing-acceptable language, also known as a recursively enumerable language
- Turing-decidable language, also known as a recursive language
- a language L is recursive if and only if both L and L are recursively enumerable
- a language is recursively enumerable if and only if it is generated by a grammar
- the language hierarchy: finite, regular, context-free, recursive, recursively enumerable
- the Church-Turing Thesis
- Turing machines (up to equivalence) can be assigned numbers: $T_0, T_1, T_2, T_3, \ldots$
- the set $K = \{\ n \mid T_n \text{ halts when run with input } n\ \}$
- the set K is recursively enumerable, but it is not recursive
- the set K is not recursively enumerable

- the Halting Problem and its computational unsolvability
- the nature of computation