The first exam will be given in class on Monday, March 4. If you have an unavoidable conflict with the date of an exam, please see me as soon as possible (before the exam date!) to discuss options for rescheduling. Last minute rescheduling/extensions will not be accommodated for something known about in advance.

The exam covers material from the first two chapters of the text, up to and including section 2.5. Section 2.6 (counting) will not be on the exam.

Many of the questions on the exam will be similar to problems on the homeworks. In particular, you can expect some short proofs, including both a formal proof of the validity of an argument and a more informal mathematical proof. There may be a proof by induction, but with sums or other number properties like we've been working with, not involving code (recursion or loop invariants). In addition, the exam may include some short answer questions that ask you to define, discuss, or explain a term or other concept.

Terms and ideas that you should be familiar with:

- translations from logic to English, and from English to logic
- proposition; propositional logic
- the logical operators $\wedge$, $\vee$, and $\neg$
- truth table
- logical equivalence ($\equiv$)
- the conditional or "implies" operator ($\rightarrow$)
- definition of $p \rightarrow q$ as $(\neg p) \vee q$
- negation of a conditional: $\neg(p \rightarrow q) \equiv p \wedge \neg q$
- tautology
- Boolean algebra
- the basic laws of Boolean algebra — you do not need to have the laws themselves memorized, but you should be familiar with what they mean and how to utilize them
- logic circuits and logic gates
- making a circuit to compute the value of a compound proposition
- finding the proposition whose value is computed by a circuit
- converse of an implication ($p \rightarrow q$ has converse $q \rightarrow p$)
- contrapositive of an implication ($p \rightarrow q$ has contrapositive $(\neg q) \rightarrow (\neg p)$)
- an implication is logically equivalent to its contrapositive
- predicates; predicate logic
- one-place predicate, two-place predicate, etc
- domain of discourse
- the quantifiers $\forall$ and $\exists$
- the rules of predicate logic — you do not need to have the laws themselves memorized, but you should be familiar with what they mean and how to utilize them

- arguments, valid arguments, and deduction
- premises and conclusion of an argument
- formal proof of the validity of an argument
- how to show that an argument is invalid
- translating arguments from English into logic
- the rules of deduction — you do not need to have the rules themselves memorized, but you should be familiar with what they mean and how to utilize them
- mathematical proof
- direct proof
- existence proof
- counterexample
- proof by contradiction
- rational number (a real number that can be expressed as a quotient of integers, $\frac{a}{b}$)
- irrational number (a real number that is not rational such as π or $\sqrt{2}$)
- divisibility $m \mid n$ (for integers n and m, $m \mid n$ if there is an integer k such that $n = km$)
- prime number (greater than 1, and cannot be factored into smaller integers)
- proof by mathematical induction
- summation notation, for example: $\sum\limits_{k=1}^{n} a_k$
- sets
- set notations: $\{a, b, c\}$, $\{1, 2, 3, \ldots\}$, $\{x \mid P(x)\}$, $\{x \in A \mid P(x)\}$
- the empty set, $\emptyset$ or $\{\}$
- equality of sets: $A = B$ if and only if they contain the same elements
- element of a set: $a \in A$
- subset: $A \subseteq B$
- $A = B$ if and only if both $A \subseteq B$ and $B \subseteq A$
- union, intersection, and set difference: $A \cup B$, $A \cap B$, $A \setminus B$
- definition of set operations in terms of logical operators
- power set of a set: $\mathcal{P}(A)$ power set is)
- universal set
- complement of a set (in a universal set): $\overline{A}$
- DeMorgan's laws for sets — you do not need to have the laws themselves memorized, but you should be familiar with what they mean and how to utilize them
- bitwise operations in Java: `&`, `|`, $\sim$
- correspondence between n-bit binary numbers and subsets of $\{n-1, n-2, \ldots, 1, 0\}$
- `&`, `|`, $\sim$ as set operations (intersection, union, complement)
- the shift operators `<<`, `>>`, and `>>>`
- hexadecimal numbers, and converting to and from binary — you do not need to memorize the binary equivalents for each hexadecimal digit but you should understand what hexadecimal numbers are and being able to convert given a table
- ordered pair: (a, b)

- cross product of sets: $A \times B$
- function $f : A \to B$
- the domain, range, and image of a function
- one-to-one function
- onto function
- bijective function
- B^A, the set of all functions from the set A to the set B