# Things to remember for 6.042 Midterm

David Chen

October 26, 2010

**Abstract**

The following is a short enumeration of topics that are "good to know" for the upcoming 6.042 exam. Readers should take note that this document is in no way meant to be a comprehensive treatment of the topics on the exam. Nor is this document meant to be a representative sample of the question subjects that will be on the midterm. For any lingering doubts, please consult Prof. Leighton's text or email 6042-staff@

## 1 Propositions

The things to know from this chapter are mostly definitions and how to think about things.

- **Proposition**: A statement that is true or false. Example: "P='Penguins are Birds"' is a true proposition.
- **Predicate**: A predicate is a proposition whose true/false value depends on one or more variables. The hypotheses that we use in induction are predicates. For example, $P(n)$ : There are no odd cycles in an $n$-node bipartite graph is a predicate.
- **Implication**. The FTL book describes it with a certain truth table, but the easiest way to think about this is if-then. $P \Rightarrow Q$ is best thought of as "if $P$ is true, then $Q$ is true as well." Note that this does not mean that if $Q$ is true, then $P$ is true.
- Two statements can be shown equivalent if their truth tables are identical. As an exercise, show that $P \Rightarrow Q$ is equivalent to $\neg Q \Rightarrow \neg P$.
- The statement in the point above, $\neg Q \Rightarrow \neg P$, is the **contrapositive** of the statement $P \Rightarrow Q$. Many times, proving the contrapositive of a statement is easier than directly proving the actual statement. Because they are equivalent, this is just as valid form of proof as any!
- Recall that switching the order of quantifiers does not generate equivalent predicates! For example, $\forall x \in X \exists y \in Y.P(x, y)$ is not equivalent to $\exists y \in Y \forall x \in X.P(x, y)$. You can see this for $X$ is the set of all students, $Y$ is the set of all classes, and $P(x, y)$ is the statement that $x$ falls asleep in $y$.

## 2 Patterns of proof

- **proof by cases**. When proving something, break it up into sections. For example, when proving something true for a set of integers, one can break them up into odd and even numbers, and show the statement to be true for both cases. *In proof by cases, you must show that the way you broke up the cases is exhaustive: that is, that there can be no situation that does not fall into one of your cases.*
- **Assume P true**. To prove $P \Rightarrow Q$, start with $P$ and assume it is true, and show that $Q$ follows from a natural argument. Simple enough.
- **Contrapositive**. Because the contrapositive is equivalent to the statement itself, proving the contrapositive is just as effective!
- **Proof by contradiction**. Say you are trying to prove proposition $P$. Assume that $P$ is false, and then show through a sequence of logical steps that some mathematical contradiction happens. Because everything else was logical, it must have been your initial assumption that was false.

In page 40 of the text, Leighton mentions some good proof practices. These will help you not only in your midterm and subsequent problem sets, but all your future academic and worldly endeavors.

- **State your game plan**. It makes it much easier for the graders to understand your solution if you tell them what you want to do first, before you go ahead and do it. Makes sense?
- **Keep a linear flow** Same vein as above. Jumping around proofs is bad - like monkeys jumping on the bed.
- **A proof is an essay, not a calculation**. Proofs are essentially arguments. When you make an argument, it's easiest to understand you if you use adequate words. Notation (equations, etc.) are there to supplement an argument, not replace it.
- **Avoid excessive symbolism**. Same as above. $\forall a \in A$ such that $N(a)$, $\exists b \in A$ s.t. $EQ(a, b) \wedge \neg N(b)$, where $A$ is the set of all arguments, $N(x)$ is true if $x$ has too many symbols, and $EQ(x, y)$ is true if $x$ and $y$ are arguments for the same idea.

Many students have tended to also overthink problems, thinking that they were too "easy." In the end, it should not be how easy the solution is that determines its correctness, but whether the argument and logical deductions were, in fact, logical. When you write down a solution, go through each line of it and ask yourself "does this seem reasonable?" When you have a deep conviction that it is correct, it usually is. Trust yoursef.

## 3 Induction/WOP/Strong Induction

Induction, Well-ordering, and Strong Induction are three equivalent ways to prove predicates. They all sound somewhat similar - it is up to you to decide what is the most appropriate tool for the job. However, it is important to bear in mind that all three methods exist. If a problem seems too hard to do with regular induction, make those extra assumptions and use strong induction! If building up from $n$ to $n + 1$ seems too tough, go down from $n + 1$ to $n$ and use WOP!

### 3.1 Induction

We need the following for an induction proof:

- An induction hypothesis $P(n)$. This is a statement that depends on $n$: for example, if we
- Show the base case. Often times, this means to show $P(1)$ is true.
- Show the inductive step. Assume that $P(n)$ is true, then show that this implies that $P(n+1)$ is true.

### 3.2 Well-Ordering Principle

The well-ordering principle states that for any set $S$ of positive integers, thre exists a $c \in S$ that is the minimum. The structure of a WOP proof is:

- A hypothesis $P(n)$.
- Assume for contradiction that $P(n)$ is not true for all $n$. Then, there is some set of positive integers $S$ for which $P$ is not true. By well-ordering, let $c$ be the smallest integer in that set.
- Say we have $c$: then, show that $c-1$ also violates $P$. But this means that $c-1 \in S$, which violates the minimality of $c$. Contradiction!

### 3.3 Strong Induction

Strong induction is the same as induction, except we assume that $P(1), P(2), \ldots, P(n)$ is true before showing $P(n+1)$.

## 4 Number Theory

- **Fundamental theorem of arithmetic**: Every integer $n$ can be factored uniquely into a product of primes.
- **GCD**. The GCD of $a$ and $b$ is the largest number that divides both $a$ and $b$.
- Every linear combination of $a$ and $b$ is a multiple of the GCD.
- $GCD(a, b)$ is a linear combination of $a$ and $b$.
- $GCD(a, b)$ is the smallest positive linear combination of $a$ and $b$.
- Note that if $a$ and $b$ are coprime, $GCD(a, b) = 1$, so there exist $s$ and $t$ such that $as + bt = 1$.
- **The Euclidean Algorithm**. Because $gcd(a, b) = gcd(b, rem(a, b))$, we can find $GCD(a, b)$ by taking successive remainders.
- **The Pulverizer** can be used to find $s, t$ such that $as + bt = gcd(a, b)$. It is done by keeping track of the coefficients of $a$ and $b$ used to express each remainder as a linear combination of $a$ and $b$. Note that the pulverizer can also be used to find multiplicative inverses modulo $n$ (why and how?).

### 4.1 Modular Arithmetic

Recall that $a \equiv b \pmod{n}$ if $n|(a-b)$.

- **Fermat's Little Theorem**. Say $p$ is prime, and $k$ is not a multiple of $p$. Then, $k^{p-1} \equiv 1 \pmod{p}$.
- **Euler's Theorem**. This is a generalization of Fermat's Little Theorem. Define the totient function $\phi(n)$ in the following way:
  - $\phi(p) = p - 1$ for $p$ prime.
  - $\phi(p^k) = p^{k-1}(p-1)$ for $p$ prime, $k$ some integer.
  - $\phi(ab) = \phi(a)\phi(b)$.

  Then, if $a$ and $n$ are coprime, $a^{\phi(n)} \equiv 1 \pmod{n}$. Note that this can also be used to find the inverse of a number modulo $n$.
- **The method of successive squaring**. We have asked you in the past to calculate $x^y \pmod{n}$ for large values of $y$. Note that we can easily calculate $x^{2^k}$ by successively squaring $x$, and we can find some product of those powers of $x$ to find $x^y$.

## 5 Graph Theory

There really isn't too much to know about graph theory, except the many definitions

- A **graph** $G = (V, E)$ is a set of vertices and edges between them.
- The **degree** of a vertex is the number of edges incident to it.
- Two graphs $G_1$ and $G_2$ are **isomorphic** if there exists a bijection $f : V_1 \to V_2$ such that every edge in $E_1$ corresponds to exactly one edge in $E_2$. That is, $G_1$ is isomorphic to $G_2$ if we can relabel $G_2$ with the vertex names of $G_1$, and find that the edges(relationships) between each pair of vertices are the same.
- To disprove isomorphism, look for properties that are not maintained. For example, if one graph has a vertex of degree 6 and the other does not, they cannot be isomorphic.
- $G' = (V', E')$ is a **subgraph** of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$.
- A graph $G$ is **bipartite** if its vertices can be split into sets $V_1, V_2$ such that vertices in $V_1$ only have edges going to $V_2$, and vice versa.
- A graph is **k-colorable** if there is a way to assign $k$ colors to the vertices of the graph such that if a vertex is colored $c$, it is not adjacent to any other vertices of color $c$.
- Bipartite $\Leftrightarrow$ 2-colorable $\Leftrightarrow$ the graph has no odd-length cycles.
- Vertices $v_1$ and $v_2$ are **connected** if there is a path in the graph from $v_1$ to $v_2$.

- A graph $G$ is connected if every vertex has a path to every other vertex.
- A **connected component** is a connected subgraph of a graph.
- A **closed walk** is a set of vertices $v_0 v_1 \dots v_k$ where each $v_i$ is adjacent to $v_{i+1}$ and where $v_0 = v_k$. If each of the $v_i$ are distinct, a closed walk is called a **cycle**.
- An **Euler walk** is a walk on a graph that traverses every edge exactly once. An **Euler tour** is an Euler walk where the start and end nodes are the same.
- A connected graph has an Euler tour $\Leftrightarrow$ every vertex has even degree.
- A **Hamiltonian cycle** is a cycle that visits every node exactly once.

# 6 Partial Orders

Know what partial orders are and how they can relate to directed graphs. Also, what is a topological sort? How does it relate to the partial order?

What are relations? What are equivalence relations? Know that equivalence classes partition the set they are in.

# 7 Sums and Asymptotics

This was this past week. You should know it.