CS 173, Lecture B
September 3, 2015
Tandy Warnow

1 Review of Sept 1 material

The examlet on September 3 will have four problems:

- One induction proof
- One problem on simplifying a logical expression
- One problem where you answer whether a logical expression is a tautology, satisfiable but not a tautology, or not satisfiable
- One problem where you take a 2CNF formula, and construct the graph that represents the formula

For A and B logical expressions (or just variables), and for T and F the logical constants **true** and **false**, respectively,

- $A \rightarrow B \equiv \neg A \vee B$
- $T \wedge A \equiv A$
- $T \vee A \equiv T$
- $F \wedge A \equiv F$
- $F \vee A \equiv A$
- De Morgan's laws
- Distribution properties
- Contrapositive: $(A \rightarrow B) \equiv (\neg B \rightarrow \neg A)$

The graph algorithm for 2SAT. 2SAT is the problem where the input is a 2CNF formula, and you wish to know if the formula is satisfiable.

The graph algorithm constructed a graph with

- One vertex for each literal (X and $\neg X$)
- Two directed edges for each clause

So:

$$A \vee B$$

is represented by two edges:

$$\neg A \rightarrow B$$

and

$$\neg B \rightarrow A$$

Why?

Suppose our logical expression was written as:

$$(A_{1,1} \rightarrow A_{1,2}) \wedge (A_{2,1} \rightarrow A_{2,2}) \wedge \ldots \wedge (A_{k,1} \rightarrow A_{k,2})$$

Now remember the contrapositive:

$$(A \rightarrow B) \equiv (\neg B \rightarrow \neg A)$$

So the i^{th} clause $A_{i,1} \rightarrow A_{i,2}$ gives you *two* directed edges:

- $A_{i,1} \rightarrow A_{i,2}$
- $\neg A_{i,2} \rightarrow \neg A_{i,1}$

So you can make a directed graph out of this kind of logical expression – the *AND* of a bunch of clauses, each of the form $A \rightarrow B$ where A and B are literals.

You would have one vertex for each literal, and two directed edges for each clause.

Easy, right?

Remember what an instance to 2SAT looks like: a logical formula in 2CNF. Hence it is the *AND* of a bunch of clauses, each clause of the form $X \vee Y$, where X and Y are literals.

Can we rewrite each of these clauses in the form $A \rightarrow B$, where A and B are literals?

Remember that

$$(A \rightarrow B) \equiv (\neg A \vee B)$$

Hence

$$(\neg A \rightarrow B) \equiv (A \vee B)$$

Now combine that with the contrapositive, and we get:

$$(X \vee Y) \equiv (\neg X \rightarrow Y) \equiv (\neg Y \rightarrow X)$$

So we could rewrite a CNF formula to be a conjunction of a bunch of clauses of the form $A \rightarrow B$.

Therefore: given an instance of 2CNF, each clause $(X \vee Y)$ is really the same thing as $(\neg X \rightarrow Y)$, and also the same thing as $(\neg Y \rightarrow X)$.

So

- Each clause gives you two directed edges.
- Each literal gives you a vertex
- The graph encodes all the information in the 2CNF expression

- The 2CNF expression is satisfiable **iff** the graph does not have a directed cycle that includes both X and $\neg X$ for some logical variable X.
- The algorithm described in the PPT from September 1 shows how to construct a satisfying assignment, if it exists.

An instance of 2SAT to work on. Consider

$$(A \vee \neg B) \wedge (B \vee A)$$

Now consider the directed graph we would get for this expression.

- How many vertices in the graph?
- How many directed edges in the graph?
- Is there any directed path from X to $\neg X$, for some literal X?
- Is there any directed cycle containing both X and $\neg X$, for some literal X?
- Is this 2CNF expression satisfiable? If so, provide the satisfying assignment, and explain how you got it (using the graph algorithm)

2 Introduction to Induction Proofs

The *idea* behind induction is simple. If I have a very contagious cold, and I sneeze then the person to my right will catch the cold; then she will sneeze, and the person to her right will catch it; and then he will sneeze, etc. Eventually everyone (to my right) will catch the cold.

An easy induction proof. **Theorem**: For all $n \in Z^+$, $\sum_{i=1}^{n} i = n(n+1)/2$

Proof. We prove this theorem by induction.

Base case: The base case is $n = 1$. Substituting $n = 1$ we get both LHS and RHS equal to 1, so the base case is true.

Our **inductive hypothesis** is that for some positive integer K, and for all integers k with $1 \leq k \leq K$, $\sum_{i=1}^{k} i = k(k+1)/2$.

We now wish to show that this statement is also true for $K + 1$.

Consider $\sum_{i=1}^{K+1} i$.

By definition this is

$$\sum_{i=1}^{K+1} i = \sum_{i=1}^{K} i + (K + 1)$$

By the inductive hypothesis,

$$\sum_{i=1}^{K} i = K(K + 1)/2$$

Hence,

$$\begin{aligned}\sum_{i=1}^{K+1} i &= \sum_{i=1}^{K} i + (K+1) \\ &= K(K+1)/2 + (K+1) \\ &= (K+1)(K/2+1) \\ &= (K+1)(K+2)/2\end{aligned}$$

Since K was arbitrary, our theorem is proved.
Q.E.D.

3 Necessary parts of induction proofs

- Inductive Hypothesis
- Base case
- Use the inductive hypothesis (that a statement is true for some value K) to prove it true for the next value $(K+1)$.
- Point out that K was arbitrary so the result holds for all K.
- Optional: say "Q.E.D."

The Inductive Hypothesis. The inductive hypothesis must be a statement that that depends on a parameter K and that is either true or false.

Your inductive hypothesis must be that the statement is true for all values k between some base case n_0 and some specific (but arbitrary) value K.

Example of bad inductive hypotheses:

- The inductive hypothesis is that $f(n)$ is even
- The inductive hypothesis is that $g(n) > n$ for all n
- The inductive hypothesis is $n^2 + 3$
- The inductive hypothesis is that $f(3) = 17$

The base case. The base case is the first value(s) for which you want to prove the statement true.

Often $n_0 = 1$, but not always. Be careful to check. Sometimes you need to establish several base cases.

Typically the base case is done properly. But sometimes it's missing - and that's a bit mistake.

4 Recurrence relations

Recurrence relations are generally functions defined recursively, as in

1. $g(1) = 3$ and $g(n) = 3 + g(n-1)$ for $n \geq 2$
2. $h(3) = 5$ and $h(n) = 2h(n-1)$ for $n \geq 4$
3. $f(1) = f(2) = 1$ and $f(n) = f(n-1) + f(n-2)$ for $n \geq 3$.
4. $k(3) = 7$ and $k(n) = 2k(n-1) - 1$ for $n \geq 4$

Sometimes it is easy to find closed form solutions for recursively defined functions, but not always. For example, can you find a closed form solution for all the functions above? Or perhaps just a lower (or upper) bound?

Induction gives you a way of proving closed form solutions, or upper and lower bounds, on recursively defined functions (i.e., recurrence relations).

Another induction proof. Let $f(n)$ be a function that maps the set of integers greater than 3 to the set of integers, defined by:

$$f(4) = f(5) = 1$$

$$f(n) = 2f(n-1) + f(n-2) \text{ if } n > 5$$

Prove that $f(n) \geq 2^{n-5}$ for $n > 5$.

Proof: by induction on n.

We will need two base cases: $n = 6$ **and** $n = 7$**.** By definition, $f(6) = 2f(5) + f(4) = 3$. Note that $2^{6-5} = 2^1 = 2$, and so $f(n) > 2^{n-5}$ for $n = 6$.

For $n = 7$, note that $f(7) = 2 \times f(6) + f(5) = 6 + 1 = 7$. Note also that $2^{7-5} = 2^2 = 4$, so that $f(n) \geq 2^{n-5}$ when $n = 7$. Hence, both base cases hold.

The **inductive hypothesis** is that for some arbitrary $K \geq 7$, $f(n) \geq 2^{n-5}$ for all $n = 5, 6, \ldots, K$. We wish to show that this statement holds for $K + 1$.

By definition,

$$f(K+1) = 2f(K) + f(K-1)$$

Therefore, by the inductive hypothesis,

$$f(K+1) \geq 2 \times 2^{K-5} + 2^{K-6}$$

$$\geq 2^{K-4} = 2^{(K+1)-5}$$

Since K was arbitrary, the result holds for all K. Q.E.D.

Another induction proof. Let $f : Z^+ x Z^+ \rightarrow Z^+$ be defined by

- $f(n, m) = n + m$ if $n = 1$ or $m = 1$,
- $f(n, m) = f(n-1, m) + f(n, m-1)$, otherwise

We would like to prove that $f(n, m) \geq n + m$ for all $n \geq 1$ and $m \geq 1$.

Base case: $n = 1$ or $m = 1$ follows immediately. So we prove the rest by induction.

Our inductive hyopthesis will be:

> $\exists K \geq 2$ such that $f(n, m) \geq n + m$ for all positive integers n, m with $n + m \leq K$

The base case is $K = 2$; hence $n = m = 1$ and the statement holds. Now assume that for some arbitrary K, the statement holds. We wish to show it holds also for $K + 1$.

Let n, m be given so that $n + m = K + 1$. If $n = 1$ or $m = 1$, then $f(n, m) = n + m$, and the statement holds.

So assume $n \geq 2$ and $m \geq 2$, so that $f(n, m) = f(n-1, m) + f(n, m-1)$. Then, by the inductive hypothesis,

$$f(n-1, m) \geq n + m - 1$$

and

$$f(n, m-1) \geq n + m - 1$$

Hence

$$f(n, m) \geq 2(n + m - 1) = 2n + 2m - 2 > n + m - 1$$

Since K was arbitrary, the statement holds for all K. Q.E.D.

5 What we learned

We learned:

- The base case is sometimes more than one value.
- The inductive hypothesis is sometimes not on an obvious parameter, but on something built using obvious parameters (like the sum)
- Induction can be used to prove upper or lower bounds on recursively defined functions.
- Induction proofs are not difficult!

6 Two-person games

Remember the original two person game?

Who wins?

Prove your assertion true by induction on the total number of rocks in the game.