Discrete Math in CS	**Problem Set 1**
CS 280 Fall 2005	**Due September 9, 2005**

Reading: Rosen, Sections 1.1 – 1.3, 1.5, and Section 3.1. For examples of the style in which we want you to write proofs, see the proofs in the Lecture Overviews and in the book (especially pages 63-68 and 214-216).

Note: Some of the questions present "claims," and ask you to decide whether they are true or false. For such a question, you should do one of the following two things: (i) give a proof of the claim; or (ii) give a counter-example that demonstrates the claim is false, together with an explanation of your counter-example.

In addition to correctness, solutions involving proofs or explanations will be graded on style and clarity.

---

(1) Recall the definition of the Borda Count method for voting, which came up in the first lecture. Each voter gives k points to their first-place option, $k-1$ to their second-place option, and in general $k-j+1$ points to their j^{th} favorite option. For each option, the total number of points given by all voters is then added up, and the options are ranked in order of their point totals. (If two options gets the same point total, we will break the tie alphabetically. Granted, we'd need a better tie-breaking rule if these were really candidates in an election, but alphabetical tie-breaking is at least easy to think about for our purposes here.)

Now, the Borda Count satisfies Arrow's Unanimity Property, so it must be the case that it fails to satisfy Arrow's Independence of Irrelevant Alternatives Property. (Arrow's Theorem says that it can't satisfy both.)

Give a specific example on which the Borda Count fails to satisfy Independence of Irrelevant Alternatives. That is, give an example of a set of rankings, including options a, b, c, and possibly others as well, so that the relative order of a and b in the consensus changes when we slide only c forward and/or backward in the rankings of some voters.

---

(2) Construct truth tables for the following Boolean expressions, as in Section 1.1 of Rosen's book. (Note that Rosen uses $\neg p$ to denote the negation of a proposition p, while we've been using $\overline{p}$. Both of these are standard notations, and we'll treat them as equivalent pieces of notation in this course.)

(a) $(p \longrightarrow (p \wedge q)) \longrightarrow q$.

(b) $(p \vee q) \wedge (p \vee \overline{q}) \wedge (\overline{p} \vee q) \wedge (\overline{p} \vee \overline{q})$.

(3) For each of the following English sentences, write an English sentence that is equivalent to its logical negation. While it will help to convert the sentence into quantifiers and

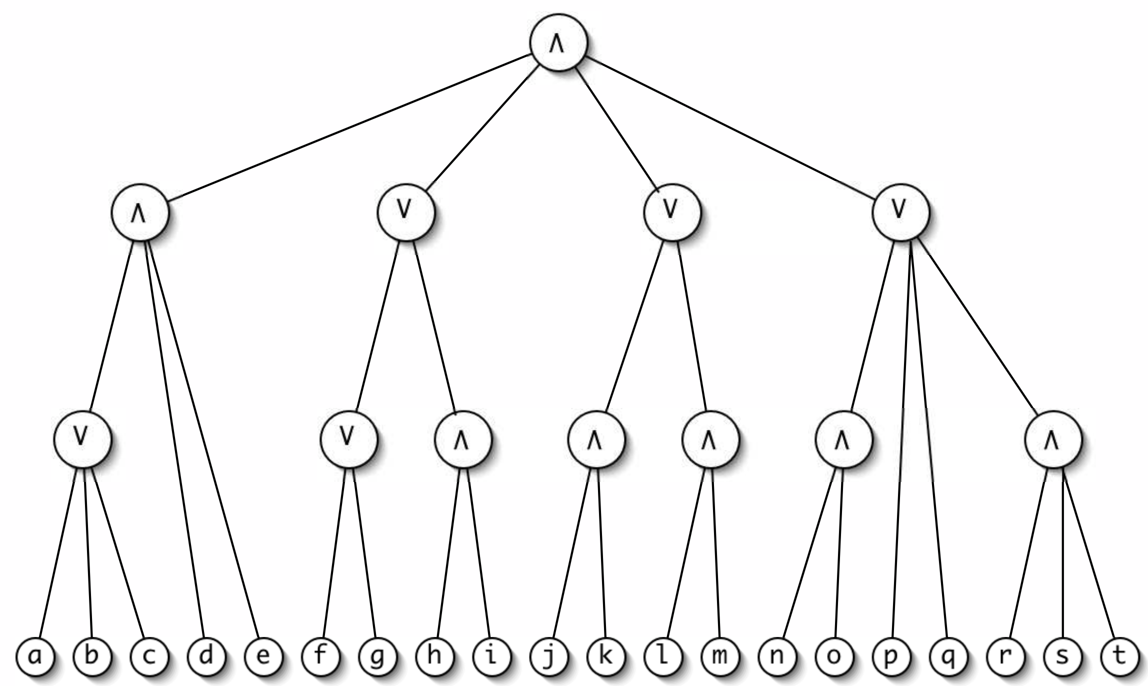

Figure 1: A Boolean formula with 20 variables.

Boolean connectives in order to take the negation, you do not need to hand this in with your answer. Your final answer should consist just of an English sentence.

Also, your answer should not simply begin, "It is not the case that ..." (or some equivalent phrasing) and then just repeat the original sentence.

Note: Inevitably, sentences written in English can be ambiguous. For sentences you think are ambiguous, explain what you see as its two (or more) possible meanings, and explain which one you are using in forming the negation. Also, note that the question does not require you to actually decide whether you think the sentence is true or false.

Example: Suppose the sentence is, "In every month this year, the market value of Google has exceeded the market value of General Motors." Then a correct answer would be, "In some month this year, the market value of General Motors was at least as large as the market value of Google." Personally, I don't see any ambiguity — i.e. alternate interpretations — for this sentence.

(You might arrive at such an answer by first converting the sentence to pseudo-mathematical notation like, "$\forall$ months in this year: $value(Google) > value(GM)$." Then, negating the quantified sentence, you'd get "$\exists$ month in this year: $value(GM) \geq value(Google)$." Finally, converting this to an English sentence, you'd get a correct answer. Again, you should only write the final sentence you obtain.)

The sentences:

1. On every major software release this year, at least one of the product managers super-

vising it had a degree in computer science or psychology.

2. Every point in Columbia is within two hundred miles of both the Equator and the ocean.

3. In each of the past five months, there have been at least six days on which the temperature exceeded ninety degrees.

4. In each of the last ten years, there has been a U.S. Senator who voted against every major piece of environmental legislation that year.

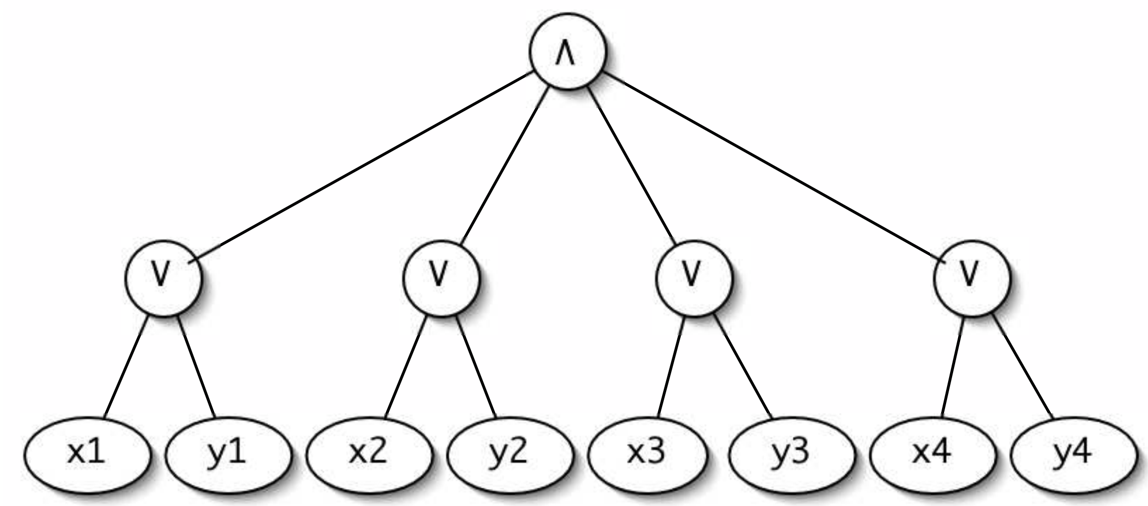

Figure 2: The Boolean formula B_4.

(4) Consider the formula B depicted in Figure 1; it has 20 variables labeled "a," "b," "c," … "t" in the figure. Give a partial setting for B that forces the value T while specifying as few variables as possible. That is, your partial setting should force the value T, and there should be no partial setting that specifies fewer variables and also forces the value T. You do not need to provide any explanation for your answer.

(*Note:* There are far too many possible partial settings to try them all explicitly. Instead, think about how one can force the value T while specifying only a few variables.)

(5) For a value $k \geq 2$, consider the Boolean formula B_k defined as

$$(x_1 \vee y_1) \wedge (x_2 \vee y_2) \wedge \cdots \wedge (x_{k-1} \vee y_{k-1}) \wedge (x_k \vee y_k).$$

So B_k has $2k$ variables: $x_1, y_1, x_2, y_2, \ldots, x_k, y_k$. Figure 2 provides an example, depicting B_4.

Consider the following claim.

> If P is a partial setting for B_k that forces the value T, then P specifies at least k variables.

Is this true or false? If you believe it is true, give a proof of the claim. If you believe it is false, disprove it by giving an example of a partial setting that forces the value T and specifies fewer than k variables.

---

(6) Some friends of yours have just released a new version of their software MyCrawl, which takes on-line news feeds and runs them as a CNN-style crawl across the bottom of a user's screen. MyCrawl comes with an option to personalize; what you do is register a standing query with it, and from the large collection of news headlines that it receives, it will only display for you the headlines that match your query.

MyCrawl uses a standard Boolean query language, as described in lecture, consisting of terms combined with AND, OR, and NOT.

So for example, if your standing query with MyCrawl were *Brad AND Jen*, then you would get the following headlines

- 'Brad is so over Jen,' says source
- Jen 'hasn't talked to Brad in weeks,' says close friend

but not

- Brad Pitt to appear in Alan Turing biopic

One thing that the User Experience Group at your friends' company is interested in understanding better is the extent to which users with very similar standing queries will end up seeing different headlines. Consider, for example, the following three standing queries

- *Query 1:* (Google OR Yahoo) AND (copyright OR trademark)
- *Query 2:* Google AND (copyright OR patent) AND infringement
- *Query 3:* Google AND Yahoo AND (patent OR trademark)

(a) Give an example of a headline that would satisfy Query 1 but neither of Queries 2 or 3.

(b) Give an example of a headline that would satisfy Query 2 but neither of Queries 1 or 3.

(c) Give an example of a headline that would satisfy Query 3 but neither of Queries 1 or 2.

In answering (a), (b), and (c), you should make your example headlines be comprehensible English sentences.

(7) Your friend Radhika holds the title of Index Maven in the MyCrawl group. She's in charge of making sure that MyCrawl is gathering as many headlines per minute as possible, so that users get large numbers of results to their standing queries.

"I never thought I'd be in the middle of one of these index-size controversies," she tells you over coffee one day. She's referring to the debates currently taking place on blogs and discussion boards concerning MyCrawl and its perennial competitor NewsStream. The question is: which has a larger collection of headlines?

Enterprising bloggers have tried to test this by posing the same queries to MyCrawl and NewsStream, and seeing which of the two returns more results. (They're using the two pieces of software not in "crawl mode," but in a simpler mode where, in reponse to a query, the software simply reports the number of headlines it has at that moment that match the query.)

The on-line discussions have turned a bit nastier after the following discovery. It turns out that someone posed two queries, "Sony" and "Apple," to both MyCrawl and NewsStream. In both cases, MyCrawl reported a larger number of results to the query. They then posed the query "Sony OR Apple" to both pieces of software, and this resulted in NewsStream reporting a larger number of queries. People are now claiming that this is an inconsistency that proves at least one of the pieces of software is lying about the number of matches to its queries. Other people are writing long posts claiming that there's nothing inconsistent about it. The whole situation's getting very confusing, and Radhika and the rest of your friends at MyCrawl are hoping you can weigh in on this issue. So you spend some time reading through the posts, which, not surprisingly, combine a bit of tecnhical content with large amounts of ad hominem bluster on all sides.

First, we can express the controversy a bit more cleanly using the following notation. For a query q and a collection of headlines H, let $size(q, H)$ denote the number of headlines in collection H that match query q. The supposed inconsistency that has all these bloggers so worked up can be viewed as relying on the following claim, which they've implicitly taken to be true:

> *Claim: For any queries q and r, and any collections of headlines H and I, if $size(q, H) > size(q, I)$ and $size(r, H) > size(r, I)$, then $size(q\ OR\ r, H) \geq size(q\ OR\ r, I)$.*

Basically, they've discovered that taking q = Sony, r = Apple, H = MyCrawl, and I = NewsStream violates this claim. So if you believe the claim, then one of the pieces of software must be misreporting its numbers. On the other hand, if the claim is false, then there's really no problem – it could well be that everyone is telling the truth.

Decide whether you think the claim is true or false. If you believe it is true, give a proof of the claim. If you believe it is false, disprove it by giving an example of two collections of headlines H and I, and two queries q and r, such that $size(q, H) > size(q, I)$ and $size(r, H) > size(r, I)$, but $size(q\ OR\ r, H) < size(q\ OR\ r, I)$.