CS109A Assignment #1
Due 3:15 PM, 1/19/96

In this and future assignments, we ask that you staple or clip the work into two parts: written homework and programming problems. Please don't forget to put your name on each part.

Written Homework

H1. (15 points) Write the truth table for the following expression. You should discover that it is a tautology. Informally, it says that if x implies that either y or z is true, and z further implies w, then x implies that either y or w is true.

$$\Big((x \to (y + z))(z \to w)\Big) \to (x \to (y + w))$$

H2. (15 points) We would like to prove the theorem "If X is a perfect square and X is even, then X is divisible by 4."

a) Designate propositional variables to stand for the three conditions about X mentioned in the theorem.
b) Write the theorem formally in terms of these propositions.
c) State the contrapositive of your answer to (b), both in terms of your propositional variables and in colloquial terms.
d) Prove the statement from (c). *Hint*: In class, we showed that if X is not divisible by 4, then either X is odd, or X is twice an odd number. You may assume that if X is twice an odd number, then X is not a perfect square (the proof requires another level of detail, but let us just take this statement as true here). Finally, use the tautology of Problem 1. (Exercise 12.9.2, FCS)

H3. (20 points) The statement "if X is not divisible by 4 then either X is odd or there exists integer Y such that $X = 2Y$ and Y is odd" can be expressed in predicate logic, using the following predicates with their informal meanings:

$div4(A)$: "A is divisible by 4."
$odd(A)$: "A is an odd integer."
$double(A, B)$: "$A = 2B$."

a) Translate the statement directly into predicate logic.
b) Write two different expressions of predicate logic that are equivalent to your answer to (a). Try to make the three expressions here and in (a) as different as you can.
c) Express each of your answers to (b) in English prose.

Programming Problems

In this and subsequent assignments, you will need to make typescripts of your interactions with SML. To make a script in file `foo` (choose any name you like in place of `foo`), execute

the UNIX command

```
script foo
```

Then, issue whatever commands you wish, e.g.,

```
sml <test
```

where your ML program is in file `test`. Finally, issue the UNIX command

```
exit
```

At this point, everything that has appeared on your screen since the `script foo` command was executed is in file `foo`. Print `foo` and hand it in. More information about making scripts can be obtained by issuing the UNIX command

```
man script
```

In general, `man` (manual) followed by the name of a command will give you useful information about that command.

P1. (20 points) The quadratic formula involves computation of the expression $\sqrt{b^2 - 4ac}$.

a) Write a sequence of `val` statements that assign a, b, and c the values 2, 1, and -3, respectively, then compute the result of this formula. ML allows you to use the function `sqrt` in expressions; for example `sqrt(4.0)` has the value `2.0 : real`. However, `sqrt` may only be applied to reals, not integers.

b) Then, rerun your sequence of statements with c set to 3 instead of -3. What happens? Can you explain why?

Hand in a script showing each of your programs executed by SML.

P2. (12 points) In a sequence of future programming assignments we are going to work with bit strings represented by a pair consisting of:

1. An integer, the length of the bit string, and
2. A list of integers, the bits string itself. We shall expect that the length of this list is always the value of the first component. We also expect that the elements of the list are all either 0 or 1. However, neither of these expectations have any bearing on this problem.

In this assignment, use a `val` expression to create as the value of variable x the list `[1,0,1,1]`. Then create as the value of variable y the pair `(4,[1,0,1,1])`, using x to help you. Hand in a script showing your program run through SML.

P3. (18 points) The following expressions are erroneous ML. Explain what is wrong with each expression. Modify them simply to be correct ML, and show (in a script) the response that SML/NJ gives for your correct expressions. (a) `not~4<5` (b) `3div2` (c) `3/2` (d) `3<4 and 5<6` (e) `3<4 or 5<6` (f) `if 2<3 then 4 .`