Exam 1

October 18, 2000

Problem 1 [30 points] **Predicates and Quantifiers**

Use the following propositions where the universe of discourse is the set of all integers:

$$\begin{aligned} P(x): &\quad x \text{ is a perfect square.} \\ Q(x): &\quad x \text{ is prime.} \\ R(x,y): &\quad x < y. \\ S(x,y): &\quad x \text{ divides } y. \\ T(x,y): &\quad x = y. \end{aligned}$$

Express each of the following statements using quantifiers, logical connectives, and the above propositions, along with variables and integers.

(a) **(5 points)** 4 is a perfect square.

(b) **(5 points)** There is an integer which divides all integers.

(c) **(5 points)** Two numbers are equal exactly when neither is less than the other.

(d) **(5 points)** An integer p is prime exactly when the only positive number less than p that divides p is 1.

(e) **(5 points)** An integer r is a perfect square only if there is a positive number between 1 and r that divides r.

(f) **(5 points)** Given ONLY the above statements, can you conclude that 4 is not prime? If not, list a true (but general) fact you need. If so, list which of the above statements are necessary.

Problem 2 [25 points] **Relations**

For each relation listed in parts (a) through (e) circle all terms listed which apply to it.

(a) **(5 points)** $A = \{a, b\}, R = \{(a, a), (a, b)\}$.

Reflexive	Symmetric
Antisymmetric	Transitive
None of these	

(b) **(5 points)** aRb whenever a and b are students and either $a = b$ or a has a GPA greater than that of b.

Partial Order	Total Order
Equivalence Relation	None of These

(c) **(5 points)** aRb whenever a and b are real numbers and $a = b^x$ for some rational number x.

Partial Order	Total Order
Equivalence Relation	None of These

(d) **(5 points)** aRb whenever a and b are rational numbers and ab is an integer.

Partial Order	Total Order
Equivalence Relation	None of These

(e) **(5 points)** aRb whenever a and b are integers such that there is a positive integer k such that $3^k | a$ and $3^k | b$ and $3^{k+1} \not| a$ and $3^{k+1} \not| b$.

Partial Order	Total Order
Equivalence Relation	None of These

Problem 3 [15 points] **Directed Graphs**

Recall that a *tournament graph* is a directed graph in which between any two different nodes a and b there is an edge either from a to b or from b to a but not both, and there are no other edges in the graph.

For each of the following, answer true or false and give a brief justification of your answer.

(a) **(5 points)** There is no tournament graph on 4 nodes with an Euler circuit.

(b) **(5 points)** There is no tournament graph on 5 nodes with an Euler circuit.

(c) **(5 points)** There is no tournament graph on 225 nodes with an Euler circuit.

Problem 4 [40 points] **Proof Comprehension**

For this proof, fill in reasons for each step denoted with a '*' and say which statements they are based on.

(a) **(8 points)** Given: $\forall x \exists y (Q(x,y) \vee R(x,y))$,

Prove: $\forall x [\exists y Q(x,y)] \vee [\exists y R(x,y)]$

Proof:

1. Let x_0 be arbitrary.
2. $\exists y (Q(x_0, y) \vee R(x_0, y))$
3. $Q(x_0, y_0) \vee R(x_0, y_0)$

   $(\exists y Q(x_0, y) \vee (\exists y R(x_0, y))$:

   1. case i: Assume $Q(x_0, y_0)$.

      *1.1. $\exists y Q(x_0, y)$

      *1.2. $(\exists y Q(x_0, y) \vee (\exists y R(x_0, y))$

   2. case ii: Assume $R(x_0, y_0)$.

      2.1. $\exists y R(x_0, y)$

      2.2. $(\exists y Q(x_0, y) \vee (\exists y R(x_0, y))$

   *3. Therefore, $(\exists y Q(x_0, y) \vee (\exists y R(x_0, y))$

*4.$\forall x [\exists y Q(x,y)] \vee [\exists y R(x,y)]$

For each of the following proofs, either say it is valid, or say it is invalid and what about the proof is incorrect.

(b) **(8 points)** Theorem: $2 = 1$. Proof:

1. Assume $2 = 1$.
2. $x^2 + 2 = x^2 + 1$.
3. $2xdx = 2xdx$. (differentiating both sides)
4. $2 = 2$ (dividing both sides by xdx)
5. Therefore, $2 = 1$ (QED)

(c) **(8 points)** Theorem: All rational numbers x greater than or equal to 1 are such that $x \leq 2^x$.

Proof: Let $P(n)$ be the statement "$n \leq 2^n$."

1. We know that $1 \leq 2^1 = 2$. Therefore, $P(1)$.
2. Assume $P(n)$ where $n \geq 1$.
3. Then, $n \leq 2^n$
4. Adding 1 to both sides, we get $n + 1 \leq 2^n + 1$
5. Substituting that $1 \leq n$, we get $n + 1 \leq 2^n + n$.
6. Substituting from step 3 into the RHS of step 5, we get $n + 1 \leq 2^n + 2^n$
7. Simplifying, we get $n + 1 \leq 2^{n+1}$.
8. Thus, $P(n + 1)$. Therefore, by induction, $\forall n P(n)$.

(d) **(8 points)** Theorem: If G is a finite graph with positive integer-weighted edges that has a spanning tree, it has a minimal spanning tree (a spanning tree T such that the sum of the weights of the edges in T is as small as it could be for any spanning tree).

Proof: Every spanning tree of G has a total weight w which is non-negative integer. Therefore, the set of possible total weights of spanning trees is a subset of the non-negative integers. By the well-ordering property, since there is a spanning tree, there must be a least weight of a spanning tree, and therefore there must be a spanning tree that has that weight. Thus, there is a minimal spanning tree.

(e) **(8 points)** Theorem: There are infinitely many prime numbers. Proof:

Let S be the set of all prime numbers. Now let $k = 1 + \prod_{p \in S} p$. Note that k is not divisible by any p in S since the remainder when dividing by p would be 1. Furthermore, notice that if p is any element of S then $k = 1 + \prod_{q \in S} q \geq 1 + p > p$. But this is a contradiction. Therefore, there must be infinitely many prime numbers.

Problem 5 [30 points] **Hamiltonian Circuits**

In this problem you will prove that for any positive integer n, any $2n$ by $2n$ grid graph has a Hamiltonian circuit.

An n by m grid graph is made up of vertices described as pairs of integers (x, y) where $0 \leq x < n$ and $0 \leq y < m$. There is an edge between two vertices if their x-coordinates differ by 1 or if their y-coordinates differ by 1 but NOT if both differ by 1.

So, for example, there would be edges between the vertices $(2, 3)$ and $(3, 3)$ but not between $(2, 3)$ and $(3, 2)$.

(a) **(8 points)** Draw a 4 by 4 grid graph. Label vertices $(1, 3)$, $(2, 2)$, and $(2, 3)$.

(b) **(22 points)** Prove by induction that any $2n$ by $2n$ grid graph has a Hamiltonian circuit for $n \geq 1$. (Other proof methods will not get credit)

Problem 6 [40 points] **State Machines and Invariants**

Consider the following algorithm, that, given a number, produces the prime factorization of that number. The prime factorization of a number is a list of all primes that divide that number, where primes may be listed multiple times if they divide the number multiple times. The product of all the numbers in the list should be equal to the original number.

Assume that Append $(List, i)$ takes a list and adds the element i to the beginning of it. Assume that () refers to the list with no elements.

FactorPrime (n):
1. $i := 2$.
2. $List := ()$.
3. $m := n$.
4. IF $m = 1$ THEN
   FactorPrime(n): $= List$.
5. IF $i|m$ THEN
   Append $(List, i)$.
   $m := m/i$.
   ELSE
   $i := i + 1$.
6. Go to step 4.

(a) **(10 points)** Represent this algorithm as a state machine. In what states will there be no transitions?

(b) **(10 points)** Using an invariant, prove that if the algorithm ever gets to the point where $m = 1$, then the list is a factorization of its input n.

(c) **(10 points)** Prove that *List* never contains any numbers that are not prime.

(d) (10 points) Prove that this algorithm always terminates. You may need to use your invariant from the previous parts.

Problem 7 [20 points]

Consider the following description of a state machine A.

Q: The set of tuples consisting of:
  M, a multiset of integers
  $request$, either $insert$(x) or get
  $response$, either $null$ or OK or x or $error$
Q_0: $M = \emptyset$, $request = null, response = null$
L:
  Input transitions: $Insert(x)$, Get
  Internal transitions: $Perform\text{-}insert$, $Perform\text{-}get$
  Output transitions: $Respond(r)$
δ:
  Insert(x):
    Can occur if: $request = null$
    Effect: $request := insert(x)$
  Get:
    Can occur if: $request = null$
    Effect: $request := get$
  Perform-insert:
    Can occur if: $request = insert(x)$
    Effect: add x to the multiset M, $response := OK$
  Perform-get:
    Can occur if: $request = get$
    Effect: if M is not empty
        remove any element x nondeterministically from M
        $response := x$
      else
        $response := error$
  Respond(r):
    Can occur if: $response = r, r \neq null$
    Effect: $response := null$, $request := null$

Now consider another machine B which differs from A only in its state and the *Perform* transitions.

Q: The set of tuples consisting of:

S, a sequence of integers

request, either *insert*(x) or *get*

response, either *null* or *OK* or x or *error*

δ:

Perform-insert:

Can occur if: $request = insert(x)$

Effect: $S := xS$ where xS denotes the concatenation of x and S

response := *OK*

Perform-get:

Can occur if: $request = get$

Effect: if M is not empty

Can occur if: $request = get$

Effect: if $S = xS'$

$S := S'$

$response := x$

else

$response := error$

Answer the following questions based on the above descriptions of A and B. If your answer is yes, give an abstraction relation. If no, give a sample trace to justify your answer.

(a) (10 points)

Does A implement B ?

(b) (10 points)

Does B implement A ?