CMPUT 272 Winter 2001: Section B3
Final Exam
Tuesday, April 17
Time: 120 minutes
Weight: 50%
Total Points: 75

Last Name:
First Name:
Unix ID:

- This exam is **open book**
- This exam should have 9 pages and 8 questions. You are responsible for checking that your exam is complete.
- Unless otherwise specified, you may use whichever of HR or GT notation you prefer.
- If you need additional paper it is available from a proctor.

Question	Mark	
1		10
2		10
3		13
4		10
5		4
6		8
7		10
8		10
Σ		75

Question 1 [10 points]
This question requires you to do a `mizar` derivation.

- Use only the basic rules of inference, namely NE, NI, CE, CI, DE, DI, IE, II, EqE, EqI, RE, ContrI.
- Each step requires a justification (i.e. a **reference** to the formulas it is inferred from) and an annotation (i.e. the **name** of the inference rule used).
- You may use GT connectives, and omit writing the "[]" after propositions,

```
environ
  pr1: (B[] implies A[]) implies (C[] or D[]);
  pr2: not C[] & not D[];

begin
  claim: not A[]
  proof
```

Question 2 [10 points]

Let $T(n) = \begin{cases} 2 \cdot T(\lfloor \frac{n}{2} \rfloor) + n & n > 1 \\ 0 & n = 1 \end{cases}$

Use induction to prove that $T(n) \leq n \log_2 n$ for all natural numbers $n \geq 1$. Clearly indicate the base case, induction step, and induction hypothesis. Also indicate where the induction hypothesis is used.

Question 3 [13 points]

The following procedure takes an array A as input. The output is a value res together with the boolean flag *valid*. When the program exits, res is intended to hold the sum of the inverses of A's elements, unless a 0 entry is encountered, in which case *valid* is false.

Sum of Inverses

Nat n;
Rat A[1..n], res;
Boolean valid;
Preconditions: (none)
Nat k;
$k, res, valid := 1, 0, true;$
do
    Variant: ?
    Invariant: $\left(res = \sum_{i=1}^{k-1} \frac{1}{A[i]}\right) \wedge \left(\neg valid \Rightarrow k \leq n\right)$
  $[]\ k = n+1 \overset{exit}{\longrightarrow}$
  $[]$ **not** $valid \overset{exit}{\longrightarrow}$
  $[]\ k \leq n\ \&\ valid \longrightarrow$
    if
      $[]\ A[k] = 0 \longrightarrow$
        $valid := false;$
      $[]\ A[k] \neq 0 \longrightarrow$
        $res := res + (1/A[k]);$
        $k := k + 1;$
    fi
od
Postconditions: $valid \Rightarrow res = \sum_{i=1}^{n} \frac{1}{A[i]}$

3.a [2 pts]: State a variant for the loop. (without proof)

In the following you may assume that the loop terminates and that one of the guards always evaluates to true.

3.b [9 pts]: Prove that the invariant is true at the beginning of <u>each</u> iteration of the loop.

3.c [2 pts]: Prove that the postcondition is true when the loop exits.

Question 4 [10 points]
Using the predicates

H(x): x is hero.
Brv(x): x is brave.
Bfl(x): x is beautiful.
A(x,y): x admires y.

translate the following sentences into predicate logic.

The universe of discourse consists of people, including the constant **Xena**.

1. Xena is a beautiful hero.

2. Not all brave people are heros.

3. Every hero has a beautiful admirer.

4. Xena admires everybody who's brave.

5. Only heros and beautiful people are admired by everybody.

Question 5 [4 points]
The truth table for the logical connective xor is:

π	ψ	π xor ψ
T	T	F
T	F	T
F	T	T
F	F	F

5.a [2 pts]: Show that the following elimination rule for xor sound: $\dfrac{\pi \texttt{ xor } \psi}{\pi \texttt{ or } \psi}$

5.b [2 pts]: Propose a sound introduction rule for xor.

Question 6 [8 points]
For functions f, g, determine whether the following statements are true or false.

Provide a counterexample for each false claim. (No justification is required for true claims.)

Recall that the order of composition is: $(f;g)(x) = g(f(x))$.

1. $\forall f, g$ if $f;g$ is 1-1 then f is 1-1.
2. $\forall f, g$ if $f;g$ is onto then f is onto.
3. $\forall f, g$ if f is total then $f;g$ is total.
4. $\forall f, g$ if f is not 1-1 then $f^{\sim}$ is not a function.

Question 7 [10 points]
Let R be a relation on the set A. Provide a derivation showing that if $R \circ R \subseteq R$ then R is transitive.

To save time writing:

- you may use GT notation
- you may abbreviate the composition of R with itself by $R \circ R$
- you may abbreviate `In[x,y,R]` as either $\langle x, y \rangle \in R$ or xRy
- you may abbreviate `Subset[S,T]` as $S \subseteq T$

```
environ
  reserve r,s,t for RELATION;
  reserve x,y,z for ELEMENT;
  CompositionDef:  for r,s,t holds Composition[r,s,t] iff
    (for x,z holds In[x,z,t] iff (ex y st In[x,y,r] & In[y,z,s]));
  TransitivityDef:  for r holds Transitive[r] iff
    (for x,y,z st In[x,y,r] & In[y,z,r] holds In[x,z,r]);
  SubsetDef:  for r,s holds Subset[r,s] iff
    (for x,y st In[x,y,r] holds In[x,y,s]);
  Given R being RELATION;

begin
  claim:  Subset[R∘R,R] implies Transitive[R] == where Composition[R,R,R∘R] holds
  proof
```

Question 8 [8 points]
Prove that $a =_{mod\ 7} b$ and $c =_{mod\ 7} d$ implies that $a + c =_{mod\ 7} b + d$.

For example: if $1 =_{mod\ 7} 8$ and $2 =_{mod\ 7} 16$ can we conclude $3 =_{mod\ 7} 24$?

Structure your argument like a `mizar` derivation. You may use facts about integer arithmetic.

`environ:`

`Mod7Def:` $\forall x, y \in \mathbb{Z}\ \left(x =_{mod\ 7} y \Leftrightarrow \exists n \in \mathbb{Z}\ (x - y = 7n)\right)$;

`begin`

`claim:` $\forall a, b, c, d \in \mathbb{Z}\ \left((a =_{mod\ 7} b) \wedge (c =_{mod\ 7} d) \Rightarrow (a + c =_{mod\ 7} b + d)\right)$

`proof`