There are a total of 100 points on this problem set. The paper part is due at 5:00 PM (*i.e.*, one and a half hours before class) on April 8$^{\text{th}}$. You can turn it in to Prof. Henz at COM1 #03-28; just put it under the door if he is not there. Late hand-ins are penalized by 5 points / minute, so it would be wise to hand it in somewhat before the last minute.

Problem 1: Natural Deduction for Predicate Logic, 35 points

Please use natural deduction to prove the following sequents; your solutions should look like tables (three columns: number, formula, justification; include boxes when required). Assume that S, P and Q are unary predicates (have arity one):

1) $\vdash (\exists x\ (P(x) \wedge Q(x))) \rightarrow (\exists x\ P(x)) \wedge (\exists x\ Q(x))$ (5 points)
2) $\vdash (\forall x\ (P(x) \wedge Q(x))) \rightarrow (\forall x\ P(x)) \wedge (\forall x\ Q(x))$ (5 points)
3) $\vdash \forall x((\exists y\ (S(x) \rightarrow Q(y))) \rightarrow (S(x) \rightarrow \exists y\ Q(y)))$ (5 points)
4) $\vdash (\forall x\ \neg P(x)) \rightarrow (\neg \exists x\ P(x))$ (10 points)
5) $\vdash (\neg \exists x\ P(x)) \rightarrow (\forall x\ \neg P(x))$ (10 points)

Problem 2: Semantics of Predicate Logic, 15 points

Consider the sentences

$$\begin{array}{lcl} \phi_1 & \equiv & \forall x.\ P(x,x) \\ \phi_2 & \equiv & \forall x.\ \forall y.\ \big(P(x,y) \rightarrow P(y,x)\big) \\ \phi_3 & \equiv & \forall x.\ \forall y.\ \forall z.\ \big(P(x,y) \wedge P(y,z) \quad \rightarrow \quad P(x,z)\big) \end{array}$$

Show the following disentailment relations:

1. $\phi_1,\ \phi_2 \not\models \phi_3$
2. $\phi_1,\ \phi_3 \not\models \phi_2$
3. $\phi_2,\ \phi_3 \not\models \phi_1$

Problem 3: Inductive Definitions and Proofs, 20 points

Consider the following representation of binary trees using the constructors Leaf and $\mathsf{Node}(\cdot,\cdot)$; also written using a notation we covered in class:

$$\mathsf{Tree} \quad \equiv \quad \mathsf{Leaf} \mid \mathsf{Node}\,(\mathsf{Tree}, \mathsf{Tree})$$

Define the size of a Tree τ, written $s(\tau)$, as follows:

$$\begin{array}{rcc} s(\mathsf{Leaf}) & \equiv & 1 \\ s\big(\mathsf{Node}\,(\tau_1,\tau_2)\big) & \equiv & 1 + s(\tau_1) + s(\tau_2) \end{array}$$

Define the height of a Tree τ, written $h(\tau)$, as follows:

$$\begin{array}{rcc} h(\mathsf{Leaf}) & \equiv & 1 \\ h\big(\mathsf{Node}\,(\tau_1,\tau_2)\big) & \equiv & 1 + \mathsf{max}\big(h(\tau_1), h(\tau_2)\big) \end{array}$$

Your task: Prove that for all trees τ, $h(\tau) \leq s(\tau)$.

Problem 4: Hoare Logic, 25 points

Consider the following program `plusabs` in the core programming language:

```
if (b > 0) {
  c = a + b;
} else {
  c = a - b;
}
```

Your task: Give a proof for the following Hoare triple (10 points).

$$\vdash_{\mathsf{par}} \{\top\} \ \texttt{plusabs} \ \{c = a + |b|\}$$

Next consider the program `downfac` in the core programming language:

```
a = x;
y = 1;
while (a > 0) {
  y = y * a;
  a = a - 1;
}
```

Your task: Give a proof for the following Hoare triple (15 points).

$$\vdash_{\mathsf{par}} \{x \geq 0\} \ \texttt{downfac} \ \{y = x!\}$$

Recall that a proper proof in the proof calculus annotates every line with the name of the rule applied to derive that line. Use the linear style of writing down the proof, mixing program lines with formulas that serve as pre- and post-conditions. Indicate formulas using the syntax $\{\cdots\}$.

Problem 5: Separation Logic, 5 points

Suppose we have some (sequential) code C and we are able to prove the triple {**emp**} C {**emp**} using the rules of separation logic. Indicate exactly those statements among the following that **must** be true.

1. C does not terminate.
2. C does not use any memory.
3. C does not allocate any memory.
4. C does not leak memory.
5. C does not allow pointers to be aliased.