Examination for "Logic and Computability (185.A45)"

30 October, 2012 **2nd Exam**

Matrikelnummer	FAMILY NAME	First Name

Remark on Notation:
We use $\supset$ for material implication.

Task 1:
Formalize the following sentence in predicate logic:

> "If all politicians are showmen and no showman is sincere then some politicians are insincere."

Is the resulting formula valid in classical logic? Prove your answer in terms of semantical arguments.

~~**Task 2:**~~
Let $sk(\cdot)$ be the skolemization operator discussed in the course and let F be the formula $\exists x\, P(x)$.

Are the formulas

1. $sk(F) \supset F$
2. $F \supset sk(F)$

valid? If not, then provide a counterexample (i.e., an interpretation I together with a domain D and a proof that the formula evaluates to false under I). Otherwise argue in detail, why the implication holds. Are the formulas valid for arbitrary closed formulas F?

Task 3:
Given the formula

$$\exists x(P \supset Q(x)) \supset (P \supset \exists x Q(x))$$

3.1 Provide a sequent calculus or a natural deduction proof for the formula, in case x is not in $FV(P)$ (= free variables of P)

3.2 Exhibit an interpretation that falsifies the formula, in case $P := P(x)$

Task 4:
Are the sets

- $\{x \mid \exists y \Phi_x(y) = 0\}$
- $\{x \mid \Phi_x(5) \uparrow \wedge x \leq 5\}$

recursive, r.e. or none of them? Motivate your answers.

Task 5:
Show that the formula $(\Diamond A \vee \Diamond \neg B) \supset \Diamond(B \supset A)$ is valid in all Kripke frames. (Use the truth conditions for $\Diamond$, $\vee$, $\supset$ directly. Do *not* reformulate the formula!)

Task 6:

For sake of concreteness, let *arithmetic programs* be defined as follows (with the obvious semantics):

$$
\begin{array}{rcl}
\langle \mathit{program} \rangle & ::= & \langle \mathit{assignment} \rangle \mid \underline{\textbf{begin}}\ \langle \mathit{program} \rangle ; \langle \mathit{program} \rangle\ \underline{\textbf{end}} \mid \\
& & \underline{\textbf{while}}\ \langle \mathit{cond} \rangle\ \underline{\textbf{do}}\ \langle \mathit{program} \rangle\ \underline{\textbf{od}} \mid \underline{\textbf{if}}\ \langle \mathit{cond} \rangle\ \underline{\textbf{then}}\ \langle \mathit{program} \rangle\ \underline{\textbf{else}}\ \langle \mathit{program} \rangle\ \underline{\textbf{endif}} \\
\langle \mathit{assignment} \rangle & ::= & \langle \mathit{variable} \rangle \leftarrow \langle \mathit{term} \rangle \\
\langle \mathit{cond} \rangle & ::= & \langle \mathit{term} \rangle = \langle \mathit{term} \rangle \mid \langle \mathit{term} \rangle < \langle \mathit{term} \rangle \mid \neg \langle \mathit{cond} \rangle \mid (\langle \mathit{cond} \rangle\ \underline{\textbf{and}}\ \langle \mathit{cond} \rangle) \mid (\langle \mathit{cond} \rangle\ \underline{\textbf{or}}\ \langle \mathit{cond} \rangle) \\
\langle \mathit{term} \rangle & ::= & 0 \mid 1 \mid \langle \mathit{variable} \rangle \mid (\langle \mathit{term} \rangle + \langle \mathit{term} \rangle) \mid (\langle \mathit{term} \rangle \cdot \langle \mathit{term} \rangle) \\
\langle \mathit{variable} \rangle & ::= & x_1 \mid x_2 \mid \ldots
\end{array}
$$

An arithmetic program π is called *partially correct* with respect to precondition A and postconditon B, if B is true in every state where π terminates when started in a state where A holds. (A and B are artihmetic formulas with free variables that may be manipulated by π.)

What follows about the provability of partial correctness of arithmetic programs from Gödel's first incompleteness theorem? Explain!