MA 5219 - Logic and Foundations of Mathematics 1
Homework due in Week 10, Tuesday.

Frank Stephan. Departments of Computer Science and Mathematics, National University of Singapore, 10 Lower Kent Ridge Road, S17#07-04.

Email fstephan@comp.nus.edu.sg
Webpage `http://www.comp.nus.edu.sg/~fstephan/mathlogic.html`
Telephone office 65162759
Office hours Thursday 14.00-15.00h

Hand in each starred homework; 1 mark per homework (if it is correct), up to 10 marks in total for homework.

11.1* Axioms of Integers.
Provide a finite set X of axioms for addition $+$, subtraction $-$ and an ordering $<$ plus a set of terms of the form 0, 1, $1+1$, $1+1+1$, $\ldots$ and -1, $-1-1$, $-1-1-1$, $\ldots$ such that for any model of X it holds that a member a of the domain A of the model is either equal to a term $\underline{z}$ of above form ($z \in \mathbb{Z}$) or satisfies $a < \underline{z}$ for all $z \in \mathbb{Z}$ or satisfies $a > \underline{z}$ for all $z \in \mathbb{Z}$.

11.2 Axioms Q1–Q5 of Successor and Addition.
Recall the axioms Q1–Q5 from page 234 (without multiplication):
Q1: $\forall x\, [Succ(x) \neq 0]$;
Q2: $\forall x \forall y\, [Succ(x) = Succ(y) \rightarrow x = y]$;
Q3: $\forall x \exists y\, [x = 0 \vee x = Succ(y)]$;
Q4: $\forall x\, [x + 0 = x]$;
Q5: $\forall x \forall y\, [x + Succ(y) = Succ(x + y)]$.
Is there a model of Q1–Q5 such that addition is not commutative in this model, that is, is there a model with elements i, j satisfying $i + j \neq j + i$.

11.3 Recursive and Partial-Recursive Functions.
Recall that a partial function is partial-recursive iff it is computed by a register machine. Is there a partial-recursive function ψ such that there is no total recursive function f with $\psi(x) \leq f(x)$ for all x in the domain of ψ?

11.4* Do-Programs.
Assume that one does not have goto-commands in the program but just loops of the form “Do n times Begin ... End” where the number of times the loop is done depends on the value of the variable n when reaching the word “Do”; loops can be nested. Furthermore, one can write statements of the form “If ... Then ... Else ...” where the Else-part can be omitted. Any other statements permitted are updates of variables as linear combinations of other variables and constants (in the natural numbers) and call of previously defined subfunctions but no recursive call of the current function. How does this concept of programming relate to primitive recursive, recursive and partial-recursive functions?