McMaster University
Department of Computing and Software
Dr. W. Kahl

SFWR ENG 3E03
Exercise Sheet 11

Design and Selection of Programming Languages

16 November 2005

Exercise 11.1 — Correctness Proof for Gödel Numbering — 20% of Final 2004

Let `int` variables a, b, p, and i, and the following program fragment **P** in a Pascal-like programming language be given:

```
(a,i) := (p,0);
while  a > i do
   i:=i+1;
   a:=a–i
od;
b:=i–a;
```

Prove partial correctness of **P** with respect to the precondition $\{p \geq 0\}$ and the postcondition

$$p = \frac{(a+b)(a+b+1)}{2} + a \quad \wedge \quad a \geq 0 \quad \wedge \quad b \geq 0$$

doumenting all intermediate steps, and showing also the implications used.

Hint: For producing this proof, you need no creativity at all, but a high degree of diligence.

Background: **P** decodes the natural number stored in p as a pair (a, b) of two natural numbers; this encoding is a simple kind of *Gödel numbering.*

Exercise 11.2 — Axiomatic Semantics: Partial Correctness Proof — 24% of Final, 2003

Consider the following program fragment in a language providing a Java-like printing statement, given an n-element Java-like array a:

```
(i, m) := (0, 0) ;
while i ≠ n do
      (i, m) := ( i + 1 , (m ∗ i + a[i]) / (i + 1) ) ;
      println(i + "  " + m)
 od
```

(a) What is the output of this program for $n = 5$ and a containing the sequence 4, 2, 9, 1, 4?

(b) What does this program do? (Short verbal description.)

(c) For this program **without the println** statement, **prove partial correctness** with respect to the **precondition** $\{ n \geq 0 \}$ and the **postcondition** $\{ m \cdot n = \sum_{j=0}^{n-1} a[j] \}$.

Important: ***Justify*** implications you use, and pay attention to **definedness of operations**!