CSCE 330 Fall 2004
Example of Axiomatic Semantics
2004-09-21

1. Find a loop invariant for this loop:
```
while (i <> n) do
  begin
    sum := sum+i;
    i := i+1
  end
```
   with precondition $i = 1 \wedge sum = 0 \wedge n \geq 1$ and postcondition $sum = \sum_{j=1}^{n-1} j$.

   Answer: The invariant (which we will call I) is: $sum = \sum_{j=1}^{i-1} j$. The following establishes that I is a loop invariant.

   (a) I holds at beginning of the while loop. More formally, the precondition entails I. This is because the precondition requires i to be 1 and sum to be 0, and I becomes $0 = \sum_{j=1}^{0} j$ when i is 1 and sum is 0.

   (b) I, together with the negation of the loop condition, entails the postcondition. That translates to: $sum = \sum_{j=1}^{i-1} j \wedge i = n$ entails $sum = \sum_{j=1}^{n-1} j$, which is clearly true.

   (c) I is invariant with respect to the body of the loop. More explictly, if I holds at the beginning of the loop and the body of the loop is executed, then I holds at the end of the loop. To prove this, it is ncessary to distinguish the values of the variables in the program from the variables themselves. Following convention, we use the names of the variables to indicate their values before execution of the loop body and the names primed to indicate their values after execution of the loop body. By inspection (or, more formally, by using the axioms for assignment), we have that $sum' = sum + i$, and $i' = i + 1$. We need to show that $sum = \sum_{j=1}^{i-1} j \wedge i <> n$ entails $sum' = \sum_{j=1}^{i'-1} j$. This is shown by replacing sum' with $sum + i$ and i' with i in $sum' = \sum_{j=1}^{i'-1} j$.

2. Show that the above program terminates.

   Answer: Consider the quantity $n - i$, where again we indentify the names of variables with their values. This quantity starts nonnegative, because the precondition $i = 1 \wedge sum = 0 \wedge n \geq 1$ entails $n - i \geq 0$ at the beginning of the loop. This quantity always decreases as the body of the loop is executed, because i increases by one, while n remains the same. This quantity never becomes negative, because the loop body is excuted only if $n <> i$. So, as the program executes, $n - i$ decreases monotonically, and $n-i$ cannot become negative. Therefore, the program must stop excuting.