# CPSC 331 — Solution for Question #3 on the Practice Midterm Test

This question concerned the "Slytherin Number Computation" problem and the `cSlytherin` algorithm shown on the *second* page of the lecture supplement.

(a) You were first asked to describe how the values of `hocus` and `pocus` are related to the value of `i` at the *beginning* and the *end* of the $j^{\text{th}}$ execution of the body of the `while` loop, for every positive integer $j$ such that the body of the `while` loop is executed at least $j$ times.

**Relationship Between the Values of These Variables:** These variables satisfy the following properties at the beginning and end of the $j^{\text{th}}$ execution of the body of the `while` loop:

- $\mathtt{i}$ is an integer such that $1 \le \mathtt{i} \le \mathtt{n}$.
- $\mathtt{hocus} = S_{\mathtt{i}-1}$.
- $\mathtt{pocus} = S_{\mathtt{i}}$.

(b) You were next asked to give a ***loop invariant*** for the `while` loop in this algorithm.

**Loop Invariant:**

i. $\mathtt{n}$ is an integer input such that $\mathtt{n} \ge 2$.
ii. $\mathtt{i}$ is an integer variable such that $1 \le \mathtt{i} \le \mathtt{n}$.
iii. $\mathtt{hocus} = S_{\mathtt{i}-1}$.
iv. $\mathtt{pocus} = S_{\mathtt{i}}$.

(c) You were asked to apply a theorem, stated in class, to show that your loop invariant is correct.

**Solution:** Loop Theorem #1 can be used to show this.

- Suppose that this program is executed when the precondition for the "Slytherin Number Computation" problem is satisfied, so that a nonnegative integer $\mathtt{n}$ is given as input. The loop is only reached if the tests at lines $1$ and $3$ are checked and failed, so $\mathtt{n} \ge 2$, as needed to establish part (i) of the loop invariant.
  Since $\mathtt{i}$ is an integer variable whose value is set to be $1$ at line $7$, $1 \le \mathtt{i} \le 2 \le \mathtt{n}$ when the loop is reached, establishing part (i) of the loop invariant.
  As noted in the answer to part (a), $\mathtt{hocus} = S_{\mathtt{i}-1}$ and $\mathtt{pocus} = S_{\mathtt{i}}$, establishing parts (iii) and (iv) of the loop invariant.
  Thus the loop is invariant when the loop is reached (if it is ever reached at all).

- Suppose that the loop is body is executed when the loop invariant is initially satisfied.

  Part (i) was satisfied at the start and, since none of the statements in the loop change the value of $\mathtt{n}$, part (i) is still satisfied at the end.

  Since part (ii) was satisfied, $\mathtt{i}$ is an integer variable such that $1 \le \mathtt{i} \le \mathtt{n}$ at the beginning of this execution of the loop body — but $\mathtt{i} < \mathtt{n}$ as well, because the loop test at line $8$ was checked and passed. Since $\mathtt{i}$ and $\mathtt{n}$ both have integer values, $\mathtt{i} \le \mathtt{n} - 1$. The steps at lines $10$ and $11$ do not change either $\mathtt{i}$ or $\mathtt{n}$, but the step at line $12$ increases the value of $\mathtt{i}$ by one, so that $2 \le \mathtt{i} \le \mathtt{n}$ when the end of the loop body is reached, establishing part (ii).

  Since parts (iii) and (iv) of the loop invariant at the beginning of the execution of the loop body, $\mathtt{hocus} = S_{\mathtt{i}-1}$ and $\mathtt{pocus} = S_{\mathtt{i}}$. After the executions of the steps at lines $9$ and $10$, $\mathtt{shazam} = S_{\mathtt{i}-1}$ and $\mathtt{hocus} = S_{\mathtt{i}}$. Thus, after the execution of the step at line $11$, $\mathtt{pocus} = 2 \times S_{\mathtt{i}} - S_{\mathtt{i}-1}$. However, since $\mathtt{i} \ge 1$, $\mathtt{i} + 1 \ge 2$, and this implies that $\mathtt{pocus} = S_{\mathtt{i}+1}$.

  Since the step at line line $12$ increases the value of $\mathtt{i}$ by one, $\mathtt{shazam} = S_{\mathtt{i}-2}$, $\mathtt{hocus} = S_{\mathtt{i}-1}$ and $\mathtt{pocus} = S_{\mathtt{i}}$ when the end of the loop both is reached, as needed to establish parts (iii) and (iv). Thus the loop invariant is also satisfied at the end of this execution of the loop body.

- Since the loop test at line $8$ has no side-effects (that is, it does not change the values of any variables) it now follows by Loop Theorem #1 that the loop invariant, given above, is correct.

(d) You were asked to give a brief proof of the partial correctness of the algorithm, assuming that the assertion for the end of the loop is correct.

**Brief Proof of Partial Correctness:** Suppose that the `cSlytherin` algorithm is executed with a nonnegative integer $\mathtt{n}$ as input. One can see by inspection of the code that the algorithm halts, returning $S_{\mathtt{n}}$ as input, if $\mathtt{n} = 0$ or $\mathtt{n} = 1$, so that it is sufficient to consider the case that $\mathtt{n} \ge 2$.

In this case, if the execution of the algorithm halts then the beginning of the loop is reached (because $\mathtt{n} \ge 2$), and the execution of the loop eventually ends, since the execution of the algorithm does. At this point $\mathtt{i} \le \mathtt{n}$, since part (ii) of the loop invariant holds, but $\mathtt{i}$ is not less than $\mathtt{n}$, because the loop test failed. Thus $\mathtt{i} = \mathtt{n}$ and $\mathtt{pocus} = S_{\mathtt{i}} = S_{\mathtt{n}}$ (by part (iv) of the loop invariant) at this point, so that $S_{\mathtt{n}}$ is returned as output, as required, after the execution of the `return` statement at line 13.

Thus $S_{\mathtt{n}}$ is returned as output if the execution of the algorithm ends. Since the execution of this algorithm has no undesired side-effects it follows that this algorithm is partially correct.