

Criterion A

Word count: 308

1 Problem

Pseudocode is an integral part of IB computer science curriculum. It is used to during lessons and during exams, so students need to be proficient at it. The problem is that the syntax of this pseudocode is quite strict, so during lessons we are assessing whether the code is correct manually. Despite there being an existing online interpreter for IB pseudocode to check for logic errors ¹, it lacks in crucial feature which is error detection, which leaves students tracking subtle syntax errors for code to even run. Both myself and my friends have been frustrated with this state of things, which forces us to waste time on mindless searching for errors that are purely syntactic and do not improve our skill [See Appendix A for concrete complaints from my friends]. In order to remedy that situation, I came forward with proposition to build a fully-fledged interpreter with lexical, syntax and run-time error detection. Due to there not being a strict standard for IB pseudocode, I would write the whole grammar for IB pseudocode, which could also serve as learning resource. Because I had to go outside the curriculum to find everything about how interpreters work, I constructed success criteria based on technical aspects of the project.

2 Success criteria

Type of interpreter : tree-walk

Language of implementation : C++14

Features:

- executing IB pseudocode consistent with developed standard based on documentation and exam booklets made by IB:
 - methods
 - recursion
 - containers and ADT's
 - * fixed-sized, multidimensional arrays
 - * stacks
 - * queues
 - while and “from to” loops
 - if else statements
 - arithmetics with operator precedence
- throwing lexical errors (undefined characters)
- throwing syntax errors
- throwing run-time errors
- flag to toggle printing abstract syntax tree
- flag to toggle logging and printing call stack
- cheat-sheet available with a flag

¹link to existing interpreter: <http://ibcomp.fis.edu/pseudocode/pcode.html>

Additional goals:

- whole syntax defined in BNF (Backaus-Naur form)