Worksheet 2 – Pseudo-code

Pseudo-code is a technique Computer Scientists developed as an aid in developing an *algorithm*. Pseudo-code can be extremely useful to students learning how to program. It allows you to move from a general *problem* description to a structured *solution* description. One strength of Pseudo-code is that it is not written in a programming language, allowing for the development of a solution without worrying about implementing each line in a programming language.

Pseudo-code is the general "vocabulary of the problem domain, not of the implementation domain. The pseudo-code is a narrative for someone who knows the requirements (problem domain) and is trying to learn how the solution is organized." [1]

Core Constructs

- SEQUENCE – one statement sequentially follows another statement.
- WHILE – repetition, continue executing body until condition is false
- IF-THEN-ELSE – conditionally execute on block or another
- REPEAT-UNTIL – repetition, execute body, repeat until condition is false
- CASE – conditionally execute on the special case being true
- FOR – repetition, execute for a fixed number of times
- PROCEDURE – group a series of statements together as a logical unit

Common Statements

- Input: READ, GET, ASK, PROMPT
- Output; WRITE, PRINT, DISPLAY, SHOW
- Compute: COMPUTE DETEMRINE, INCREMENT, DECREMENT
- Initialize: SET, INIT

Other Syntax

- "gets" operator : ← (LHS *gets* value of RHS)
- "comment" Δ : text is for information purposes only

Goals

Write pseudo-code that describes how to solve the problem. Pseudo-code should be descriptive (i.e. written in a language that you understand). Good pseudo-code omits details that are only relevant to a specific programming language (i.e. omits class definitions, etc).

Bibliography

1. John Dalbey, *Pseudo-code Standard*, http://www.csc.calpoly.edu/~jdalbey/SWE/pdl_std.html, downloaded 1/12/2004.