Pseudocode 101
MIT- edX

What is pseudocode? Pseudocode is a simplified, half-English, half-code outline of a computer program.

Why use it?

- Because it can help you to clarify your thoughts, and design a routine properly, before you start to write any code.

  One of the hardest things to resist is the temptation to start writing code! Compared to typing source code, designing the functions which will make up a program seems dull, and perhaps even like a time wasting activity. But spending ten minutes to think out carefully the pros and cons of different approaches to the goal can save you hours of time debugging and refactoring your code later on.

- Pseudocode makes reviews easier. You can review detailed designs without examining source code. Pseudocode makes low-level design reviews easier and reduces the need to review the code itself.

- Pseudocode supports the idea of iterative refinement. You start with a high-level design, refine the design to pseudocode, and then refine the pseudocode to source code. This successive refinement in small steps allows you to check your design as you drive it to lower levels of detail. The result is that you catch highlevel errors at the highest level, mid-level errors at the middle level, and low-level errors at the lowest level – before any of them becomes a problem or contaminates work at more detailed levels.

- Pseudocode makes changes easier. A few lines of pseudocode are easier to change than a page of code. Would you rather change a line on a blueprint or rip out a wall and nail in the two-by-fours somewhere else? The effects aren't as physically dramatic in software, but the principle of changing the product when it's most malleable is the same. One of the keys to the success of a project is to catch errors at the "least-value stage," the stage at which the least effort has been invested. Much less has been invested at the pseudocode stage than after full coding, testing, and debugging, so it makes economic sense to catch the errors early.

- Pseudocode minimizes commenting effort. Often, students write the code and add comments afterward. When we use pseudocode, the pseudocode statements directly become the comments, so it actually takes more work to remove the comments than to leave them in.

- Pseudocode is easier to maintain than other forms of design documentation. With other approaches, design is separated from the code, and when one changes, the two fall out of agreement. With the PPP, the pseudocode statements become comments in the code. As long as the inline comments are maintained, the pseudocode's documentation of the design will be accurate.

1

Pseudocode 101

Pseudocode is a simplified, half-English, half-code outline of a computer program.

June 05, 2015