

Program Development

Define the problem

- Carefully read and re-read the problem until you understand completely what is required.
- Break down the problem in three separate components:
 - **input, output, processing steps** to produce the required output.
- **(tools: paper and pen)**

Outline the solution

- Break up the problem into smaller tasks (steps) and establish an outline solution. This rough draft of the solution may include
- Major processing steps
- Major subtasks
- Major control structures
- Major variables and data structures
- Main-line logic
- **(hierarchy or structure chart)**

Develop the outline into an algorithm

- A set of precise steps that describe exactly the tasks to be performed and the order in which they are to be carried out.
- **(pseudo-code, flowchart, structured English)**

Test the algorithm for correctness

- Desk checking: test data needs to be walked through each step in the algorithm to check that the instructions will actually do what they are designed to.
- **(walkthrough, keep track of all major variables on a sheet of paper.)**

Code the algorithm into a specific programming language

Program Data: Variables, constants and literals

Run the program on the computer

Document and maintain the program

- Internal documentation: prologue (header), inline comments
- External documentation: hierarchy charts, solution algorithm, test data results

What must be designed?

1. Logical flow of instructions
2. Mathematical procedures
3. Appearance of screens
4. Way information is presented to user
5. User-friendliness
6. Manuals and other forms of documentation