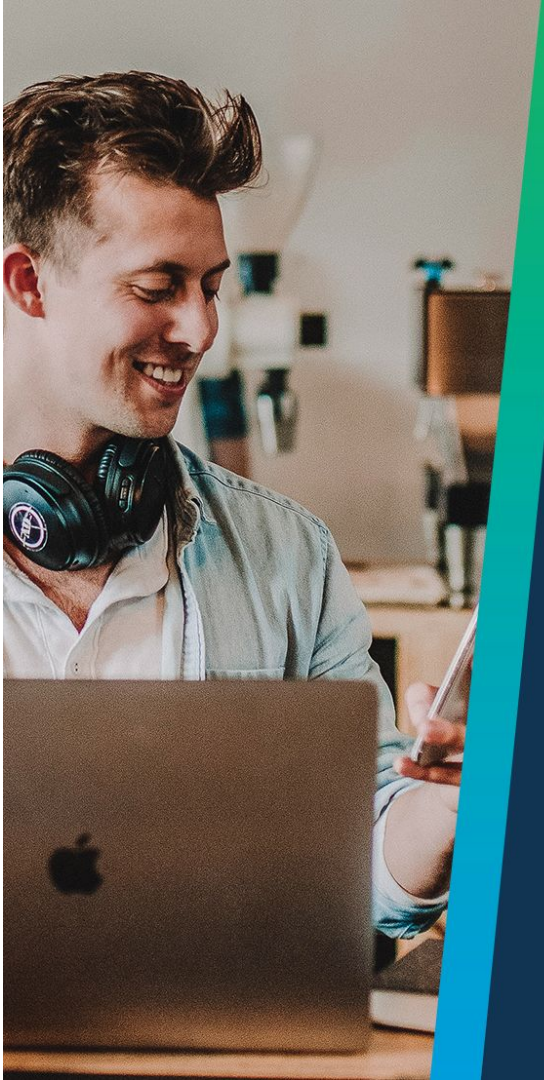


Software Engineering Bootcamp

Hyperiondev

Pseudo Code

Introduction to Pseudocode

- Bridging Problem Statements and Code
- Learn how pseudo code can help you clarify your thoughts and properly plan your programs before writing any code.

Why Pseudocode ?

- ★ An unambiguous series of steps that leads to the solution of a problem.
- ★ Can be displayed / represented with natural language.
- ★ Pseudo code is a representation of an Algorithm.
- ★ "Pseudocode is like a rough draft for coding. It helps you plan out your logic without worrying about syntax."
- ★ Example: "Before you make breakfast, you mentally break down the steps: get ingredients, cook, and serve. That's similar to pseudocode!"

Pseudo Code

- ★ Described previously as a representation of an algorithm.
- ★ Written in short, plain English phrases to describe code for programs.
- ★ Used to create programming statements that achieve the required results for a program.

Why Pseudo Code?

Pseudo code is an alternative way to write up algorithms for programmers to understand.

It is popular because:

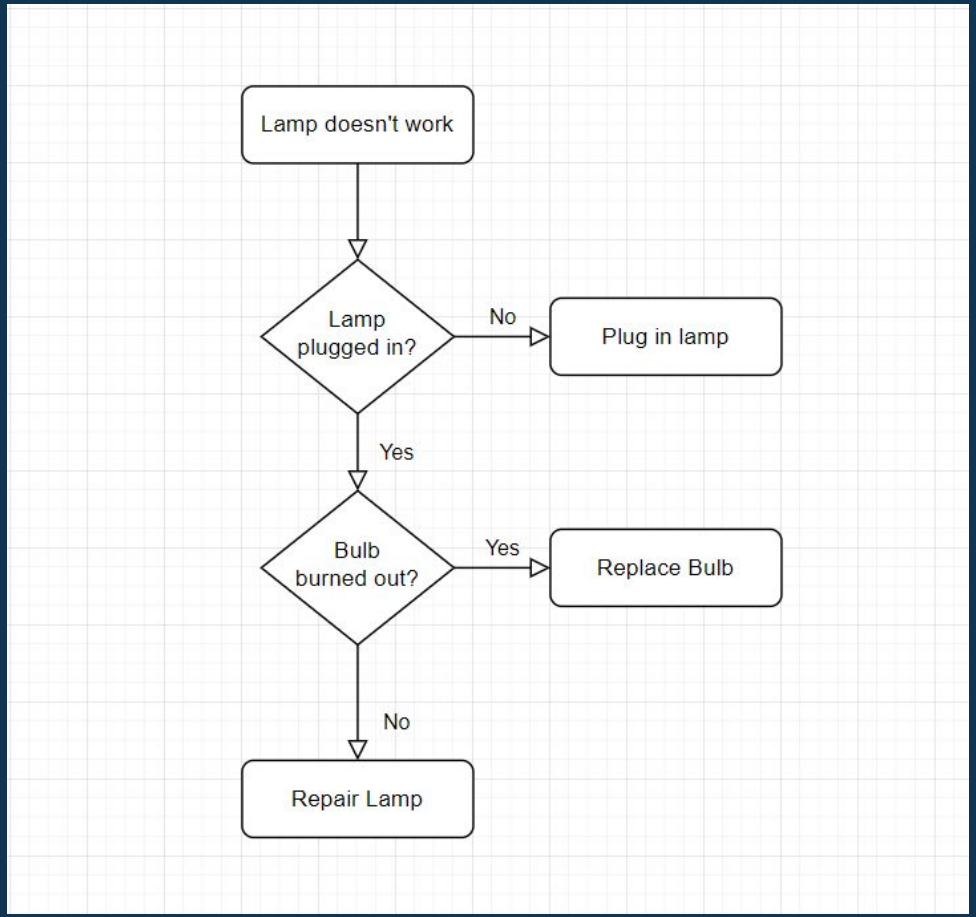
- ★ It is easy to read and write.
- ★ It can focus on the core logic of the program.
- ★ It is structured in plain English.

Flow Diagrams

Flow diagrams (or flowcharts) visually represent a process or algorithm.

Why do we use Flow Diagrams?

- ★ Break down complex processes into manageable steps.
- ★ Easily share ideas with others, including non-programmers.
- ★ Identify and correct errors in logic before coding.
- ★ Serve as a reference for coding and future maintenance.

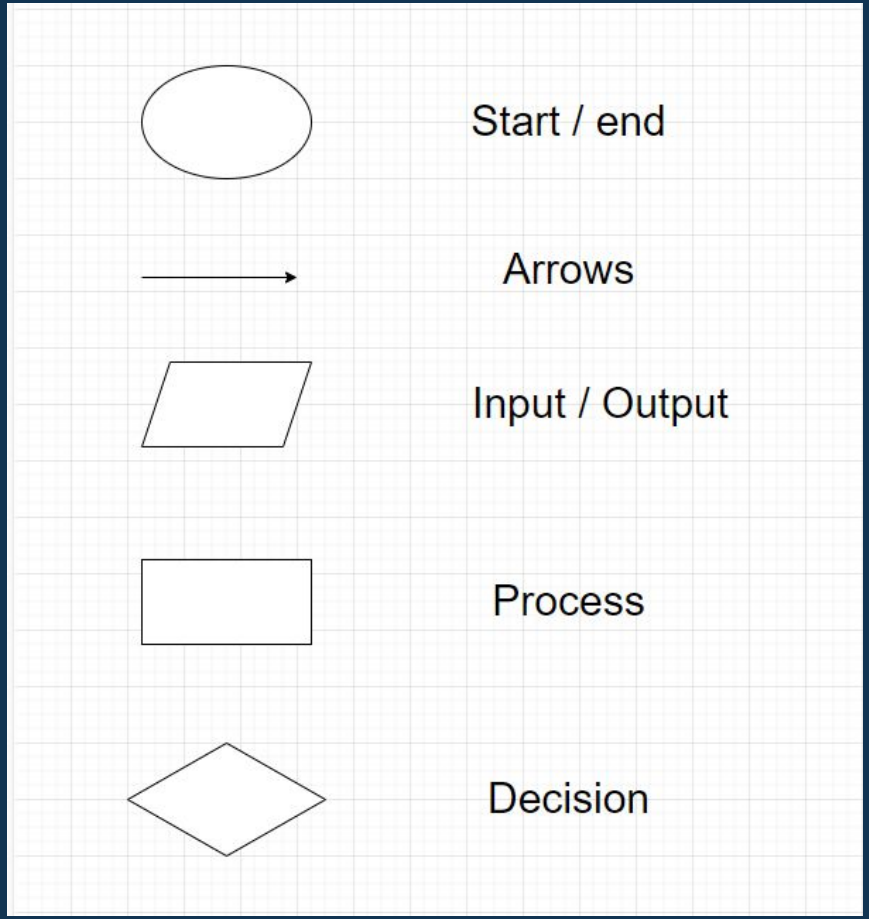

Lamp doesn't work
Lamp plugged in?
No
Plug in lamp
Yes
Bulb burned out?
Yes
Replace Bulb
No
Repair Lamp

Symbols in Flow Diagrams

The symbols aren't just there for fun, they have meaning and structure that is key to the description of the algorithm.

So What do they mean ?

- ★ **Oval (Start/End)**: Marks the beginning or end of a process.
- ★ **Rectangle (Process)**: Represents a process or action step.
- ★ **Diamond (Decision)**: Represents a decision point with yes/no or true/false outcomes.
- ★ **Arrow (Flowline)**: Shows the flow of the process.


Start / end
Arrows
Input / Output
Process
Decision

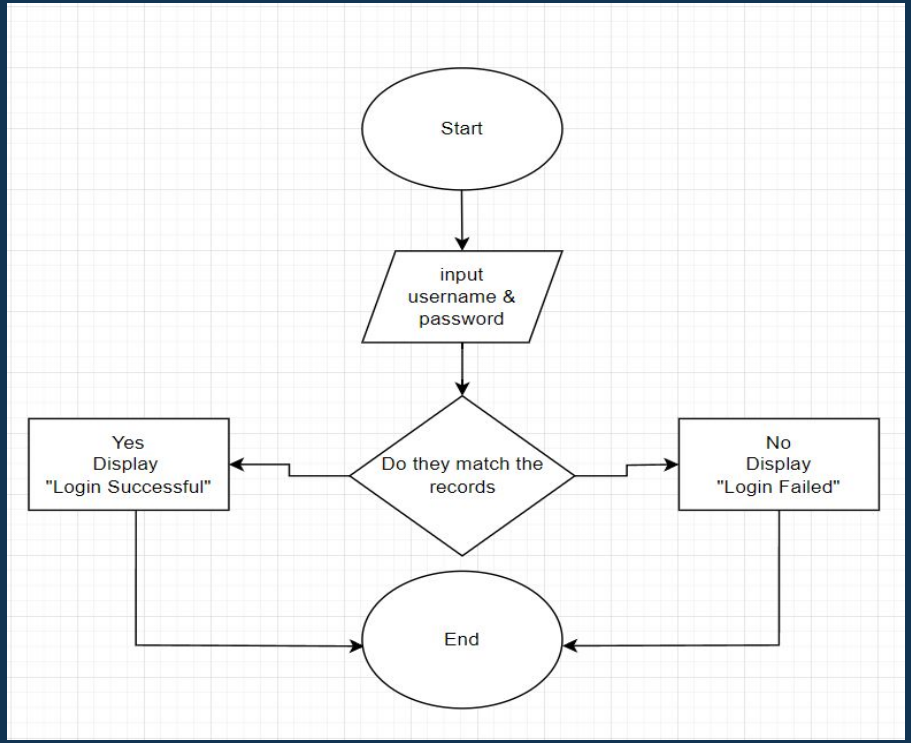
Problem Statement

Verify user login

1. Start
2. Input username and password
3. Decision: Do they match the records?
4. Yes: Display "Login Successful"
5. No: Display "Login Failed"
6. End

Flow Chart Diagram

Verify user login



Hyperiondev

Example Pseudocode

```
START
INPUT username, password
IF username AND password are correct THEN
    PRINT "Login Successful"
ELSE
    PRINT "Login Failed"
END IF
END
```

Flow Diagram vs Pseudocode

Flow Diagram: Visual representation of a process.

Pseudocode: Written outline of an algorithm in plain language.

Flow diagrams help visualize pseudocode logic before coding

Pseudo Code Conventions

- ★ Should be written in simple, plain English.
- ★ Each instruction / functionality should be written on a separate line.
- ★ Has a Start point and End point and is written from top to bottom.

As pseudo code does not follow any syntax from any programming language it is not a necessity to use indentation to outline structure, and no restrictions exist in this regard.

Pseudo Code Example

Making a cup of tea:

START

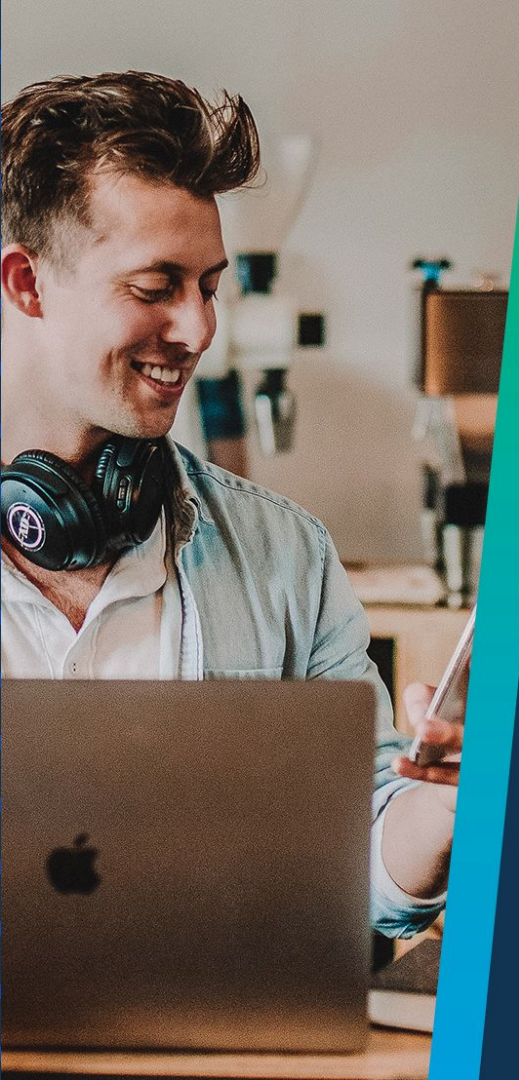
1. Organise ingredients
2. Fill and switch on kettle,
3. Put tea bag in cup,
4. Wait for water to boil,
5. Add water to cup,
6. Remove tea bag with spoon,
7. Add milk or sugar,
8. Enjoy.

END

Hyperiondev

Q & A Section

Please use this time to ask any questions relating to the topic, should you have any.



Hyperiondev

Thank You for Joining Us