# Programming ..... Problem Solving + Implementation

-Zubair uddin Shaikh

# Programming….

- Programming is a two phase process…
  - **Problem Solving Phase**
    - Analysis and Specification
      - Understand (define) the problem and what the solution must do.
    - General Solution (Algorithm)
      - Logical Sequence of Steps
    - Verify
      - Dry run

# Phases…

- **Implementation Phase**
  - Concrete Solution
    - Algorithm → Program
  - Test/Evaluate
    - Manual/Automated
- **Maintenance**
  - Improvements

# Programming!!!

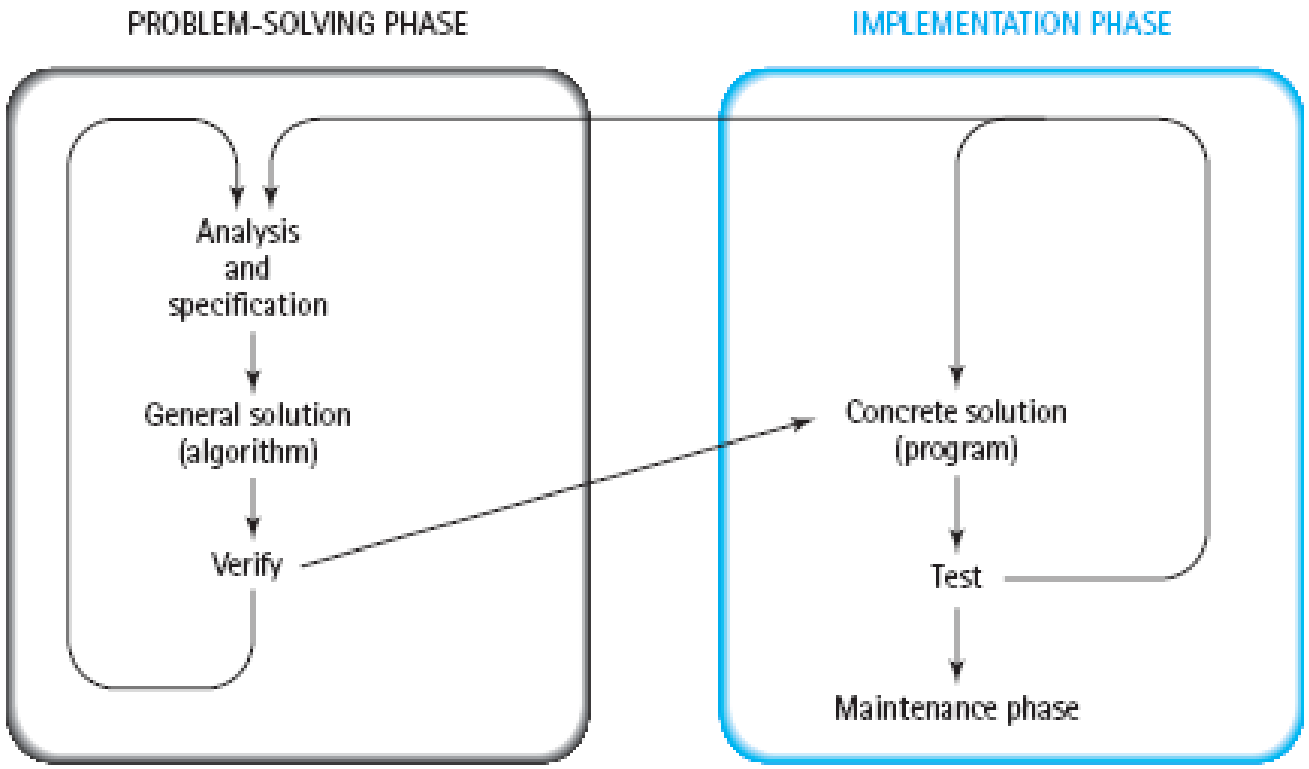
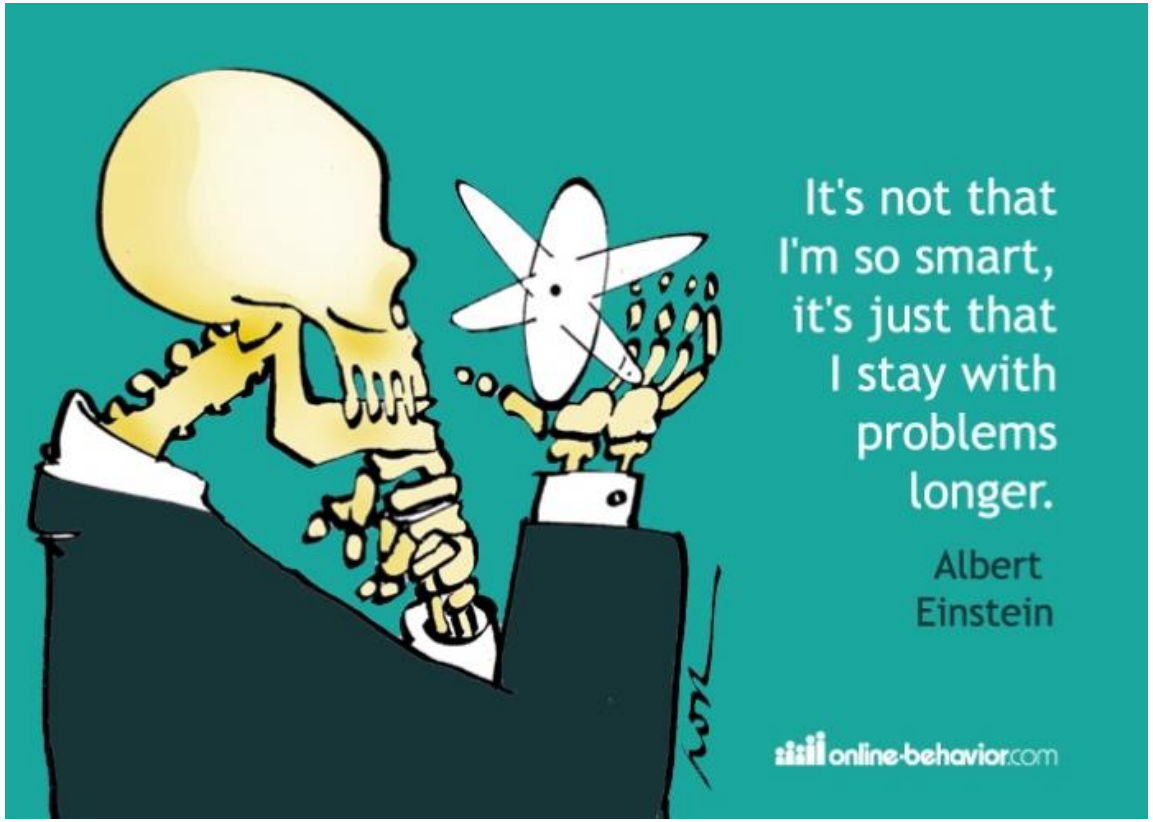



Figure 1.1 *Programming process*

# Problem Solving…. Clue for Learning (CFL)


- Finding Solution is not enough….
- The Process to Solution
  - AND
- The Effectiveness of Solution
  - Are As much Important

# Problem solving Requires Determination……..CFL



# Problem Solving……Mistakes (CFLs:)

- You will make mistakes;
  - Accept it.
  - This is normal and you learn from them.
  - Do not avoid risking making them.
  - The Process is MORE important than ANSWER ☺
- Solution is not always complex;
  - Sometimes a simple solution can fix a complex problem.
- “An expert is a man who has made all the mistakes which can be made in a very narrow field.”**Niels Bohr**
- “Anyone who has never made a mistake has never tried anything new.” **Albert Einstein**

# Problem Solving……Simplicity (CFLs:)

- “Any intelligent fool can make things bigger and more complex... It takes a touch of genius - and a lot of courage to move in the opposite direction.”

  **Albert Einstein**

- “Everything should be made as simple as possible, but not simpler.”
  **Albert Einstein**

- **Take the problem and break it down in to smaller and smaller chunks that are now manageable.**

# Formalizing the Thinking Process!!!!

- In day to day life, we come to a lot of problems and find solutions to them…..
  - Finding a solution starts with thinking….
- We are going to formalize … What we are already doing ☺
- The First step/exercise: Transform the thoughts to Paper → Natural Language transformation of thoughts….
- Swap contents of two containers… 1st Example

# Formalizing the Thinking Process!!!!

- **Bottle Opening Problem [Class activity]**
- Write down a detailed list of instructions on how to open a bottle. [2 column form]
- Have you considered all situations?
  - For example:
    - What if the bottle is upside does your routine still work?
    - What if the top is already off, does you routine still work?
    - Is your routine sufficiently detailed that if you gave it to someone they would follow exactly as you expected?
    - Is there any other way it could have been done?
    - What type of bottle is being opened?
    - Does it work for all bottles?
- You have produced the solution based on assumptions

# Formalizing the Thinking Process!!!!

- **Making a cup of tea [Class activity]**

- Write down a detailed list of instructions on how to make a cup of tea. [2 column form]

- Have you considered all situations?
  - Give the list of steps to another class mate
  - Let's see, how much the steps are clear to him
- Have you made any assumptions?????
- Let's compare the lists of steps produced by students…..

# What is an algorithm?

- Before a computer can perform a task, it must have an algorithm that tells it what to do.
- Informally: “An algorithm is a set of steps that define how a task is performed.”
- Formally: “An algorithm is an ordered set of unambiguous executable steps, defining a terminating process.”
  - Ordered set of steps: structure!
  - Executable steps: doable!
  - Unambiguous steps: follow the directions!
  - Terminating: must have an end!

# What is an algorithm? (Cont'd)

- An algorithm is an ordered set of
  - unambiguous,
  - executable steps,
  - defining a terminating process.

# Important Properties of Algorithms

- Correct
  - always returns the desired output for all legal instances of the problem.
- Unambiguous
- Precise
- Efficient
  - Can be measured in terms of
    - Time
    - Space
  - Time tends to be more important

# Representation of Algorithms

- A single algorithm can be represented in many ways:
  - Formulas: F = (9/5)C + 32
  - Words: Multiply the Celsius by 9/5 and add 32.
  - Flow Charts.
  - Pseudo-code.
- In each case, the algorithm stays the same; the implementation differs!

# Expressing Algorithms

More easily expressed

- English description
- Flow Chart
- Pseudo-code
- High-level programming language

More precise

# Problem Solving….

- 1- Analysis and Specification
  - Input-Process-Output (IPO)
    - Problem Analysis Chart (PAC)
  - Examples
  - Problem sets
- 2- Algorithm
  - Algorithmic Constructs
  - Flow Chart
  - Pseudo code
- 3- Test/Evaluate
  - Dry run

# Analyze the Problem…

- Thoroughly understand the problem
- Understand problem requirements
  - Does program require user interaction?
  - Does program manipulate data?
  - What is the output?
- If the problem is complex, divide it into subproblems
  - Analyze each subproblem as above

# Analyze the Problem…

- Example: **Payroll Problem**
  - Calculate the salary of an employee who works by hourly basis. The formula to be used is

Salary = Hour works * Pay rate

| Input | Processing | Output |
|---|---|---|
| Hours work,<br>Pay rate | Salary = Hours work * payrate | Salary |

# Problem 1

Write a Problem Analysis Chart (PAC) to convert the distance in miles to kilometers where 1.609 kilometers per mile.

| Input | Processing | Output |
|---|---|---|
| Distance in miles | Kilometers = 1.609 x miles | Distance in kilometers |

# Problem 2

- Write a Problem Analysis Chart (PAC) to find an area of a circle where area = pi * radius * radius

| Input | Processing | Output |
|---|---|---|
| radius | area = 3.14 x radius x radius | area |

# Problem 3

- Write a Problem Analysis Chart (PAC) to compute and display the temperature inside the earth in Celsius and Fahrenheit. The relevant formulas are

  Celsius = 10 x (depth) + 20

  Fahrenheit = 1.8 x (Celsius) + 32

| Input | Processing | Output |
|---|---|---|
| depth | Step1: celsius = 10 x (depth) + 20<br>Step2: fahrenheit = 1.8 x (celsius) + 32 | Display celsius,<br>Display fahrenheit |

# Problem 4

- Write a problem analysis chart (PAC) that asks a user to enter the distance of a trip in miles, the miles per gallon estimate for the user's car, and the average cost of a gallon of gas. Calculate and display the number of gallons of gas needed and the estimated cost of the trip.

| Input | Processing | Output |
| --- | --- | --- |
| distance,<br>miles per gallon,<br>cost per gallon | Step1: gas needed = distance / miles per gallon.<br><br>Step2: estimated cost = cost per gallon x gas needed | Display gas needed<br><br>Display estimated cost |