# Problem Solving

# Advanced C

## Problem Solving – What?

- An approach which could be taken to reach to a solution
- The approach could be ad hoc or generic with a proper order
- Sometimes it requires a creative and out of the box thinking to reach to perfect solution

# Advanced C

## Problem Solving

- Introduction to SDLC
- Polya's Rules
- Algorithm Design Methods

# Advanced C

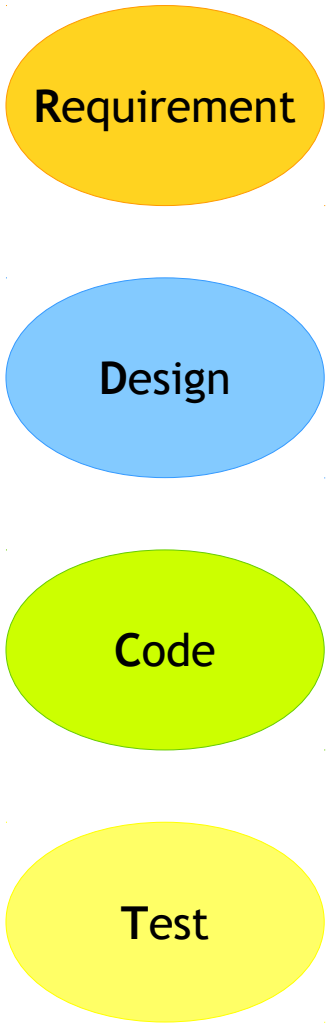
## Problem Solving - SDLC – A Quick Introduction

- Never jump to implementation. Why?
  - You might not have the clarity of the application
  - You might have some loose ends in the requirements
  - Complete picture of the application could be missing

  and many more...

# Advanced C

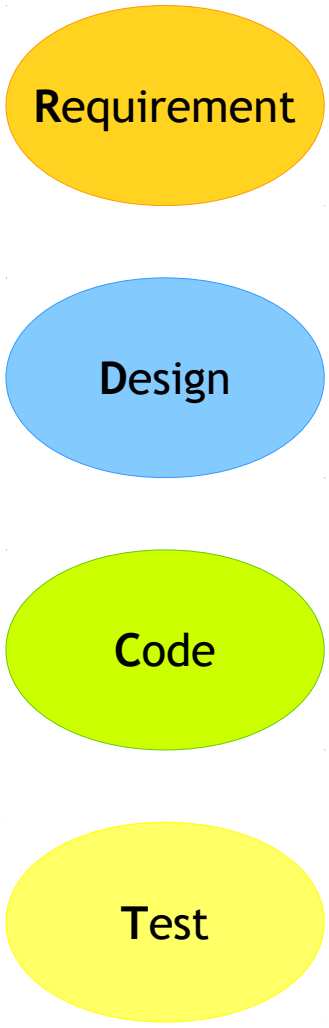
## Problem Solving - SDLC – A Quick Introduction




- Understand the requirement properly
- Consider all the possible cases like inputs and outputs
- Know the boundary conditions
- Get it verified

# Advanced C

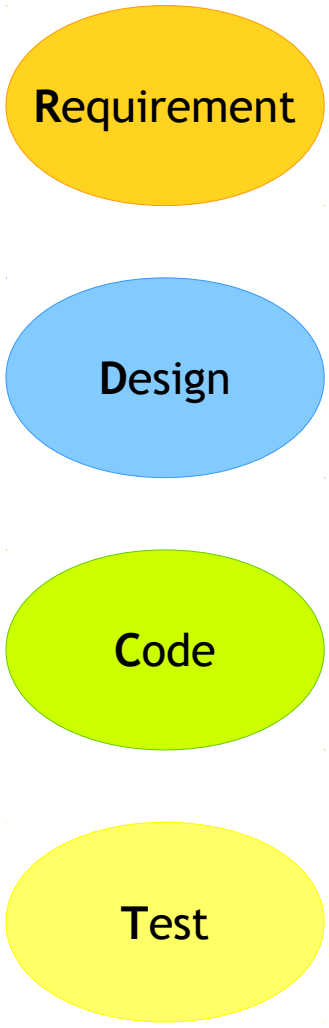
## Problem Solving - SDLC – A Quick Introduction




- Have a proper design plan
- Use some algorithm for the requirement
  - Use paper and pen method
- Use a flow chart if required
- Make sure all the case are considered

# Advanced C

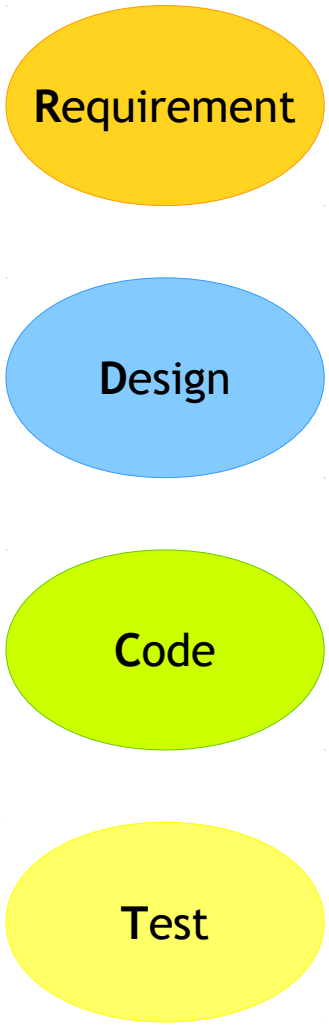
## Problem Solving - SDLC – A Quick Introduction




- Implement the code based on the derived algorithm
- Try to have modular structure where ever possible
- Practice neat implementation habits like
  - Indentation
  - Commenting
  - Good variable and function naming's
  - Neat file and function headers

# Advanced C

## Problem Solving - SDLC – A Quick Introduction




- Test the implementation thoroughly
- Capture all possible cases like
  - Negative and Positive case
- Have neat output presentation
- Let the output be as per the user requirement

# Advanced C

## Problem Solving - How?

- Polya's rule
  - Understand the problem
  - Devise a plan
  - Carryout the Plan
  - Look back

# Advanced C

## Problem Solving – Algorithm – What?

- A procedure or formula for solving a problem
- A sequence of unambiguous instructions for solving a problem, i.e., for obtaining a required output for any legitimate input in a finite amount of time.

# Advanced C

## Problem Solving - Algorithm - Need?

- Algorithms is needed to generate correct output in finite time in a given constrained environment
  - Correctness of output
  - Finite time
  - Better Prediction

# Advanced C

## Problem Solving – Algorithm – How?

- Natural Language
- Pseudo Codes
- Flowcharts etc.,

# Advanced C

## Problem Solving - Algorithm - Daily Life Example

- Let's consider a problem of reaching this room
- The different possible approach could be thought of
  - Take a Walk
  - Take a Bus
  - Take a Car
  - Let's Pool
- Lets discuss the above approaches in bit detail

# Advanced C

## Algorithm – Reaching this Room – Take a Walk

The steps could be like

1. Start a 8 AM
2. Walk through street X for 500 Mts
3. Take a left on main road and walk for 2 KM
4. Take a left again and walk 200 Mts to reach

# Advanced C

## Algorithm - Reaching this Room - Take a Walk

- Pros
  - You might say walking is a good exercise :)
  - Might have good time prediction
  - Save some penny
- Cons
    - Depends on where you stay (you would choose if you stay closer)
    - Should start early
    - Would get tired
    - Freshness would have gone

# Advanced C

## Algorithm – Reaching this Room – Take a Bus

The steps could be like

1. Start a 8 .30 AM
2. Walk through street X for 500 Mts
3. Take a left on main road and walk for 100 Mts to bus stop
4. Take Bus No 111 and get down at stop X and walk for 100 Mts
5. Take a left again and walk 200 Mts to reach

# Advanced C

## Algorithm - Reaching this Room - Take a Bus

- Pros
  - You might save some time
  - Less tiredness comparatively
- Cons
  - Have to walk to the bus stop
  - Have to wait for the right bus (No prediction of time)
  - Might not be comfortable on rush hours

# Advanced C

## Algorithm – Reaching this Room – Take a Car

The steps could be like

1. Start a 9 AM
2. Drive through street X for 500 Mts
3. Take a left on main road and drive 2 KM
4. Take a left again and drive 200 Mts to reach

# Advanced C

## Algorithm - Reaching this Room - Take a Car

- Pros
  - Proper control of time and most comfortable
  - Less tiresome
- Cons
  - Could have issues on traffic congestions
  - Will be costly

# Advanced C

## Algorithm – Reaching this Room – Let's Pool

The steps could be like

1. Start a 8.45 AM
2. Walk through street X for 500 Mts
3. Reach the main road wait for you partner
4. Drive for 2 KM on the main road
5. Take a left again and drive 200 Mts to reach

# Advanced C

## Algorithm - Reaching this Room - Let's Pool

- Pros
  - You might save some time
  - Less costly comparatively
- Cons
  - Have to wait for partner to reach
  - Could have issues on traffic congestions

# Advanced C

## Algorithm – Daily Life Example - Conclusion

- All the above solution eventually will lead you to this room
- Every approach some pros and cons
- It would be our duty as a designer to take the best approach for the given problem

# Advanced C

## Algorithm – A Computer Example1

- Let's consider a problem of adding two numbers
- The steps involved :

  Start

  Read the value of A and B

  Add A and B and store in SUM

  Display SUM

  Stop

- The above 5 steps would eventually will give us the expected result

# Advanced C

## Algorithm – A Computer Example1 – Pseudo Code

- Let's consider a problem of adding two numbers
- The steps involved :

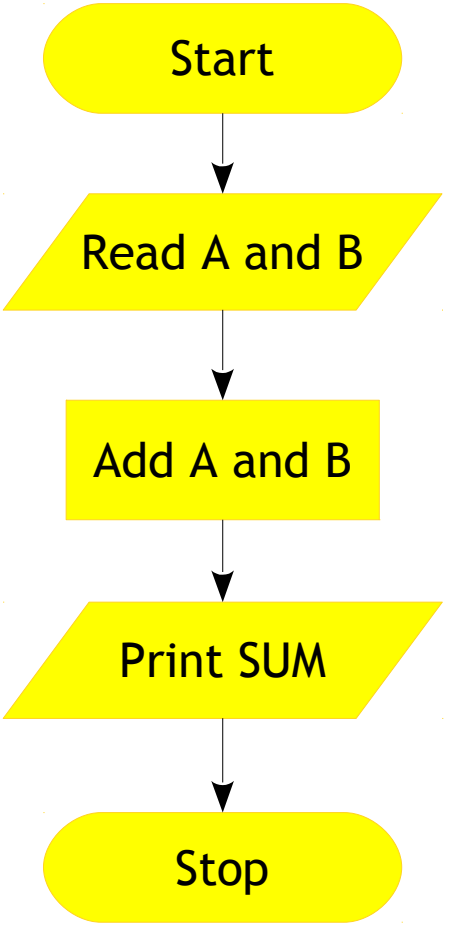
```
BEGIN

Read A, B

SUM = A + B

Print SUM

END
```

- The above 5 steps would eventually will give us the expected result

# Advanced C

## Algorithm – A Computer Example1 – A Flow Chart

- Let's consider a problem of adding two numbers



# Advanced C

## Algorithm - DIY - Pattern

- Write an algorithm to print the below pattern

```
**********
*        *
*        *
*        *
*        *
*        *
*        *
**********
```

# Advanced C

## Algorithm - DIY - Pattern

- Write an algorithm to print number pyramid

```
123454321
1234___4321
123_____321
12_______21
1_________1
```

# Advanced C

## Algorithm – DIY

- Finding largest of 2 numbers
- Find the largest member of an array

# Advanced C

## Algorithm – Home Work

- Count the number of vowels
- Count the number of occurances of each vowel
- To find the sum of n – natural numbers
- Convert a number from base 10 to base N