# Tools and Techniques of Problem Analysis

**There are 3 techniques for problem analysis:**

1. **Algorithm**
2. **Flow chart**
3. **Pseudo code**

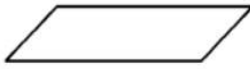## Algorithm characteristics

1. **It should have finite number of steps**. No one can be expected to execute infinite number of steps.
2. The steps must be in order and simple
3. **Each step should be defined clearly stated i.e. without un-ambiguity (without doubtfulness)**
4. Must include all required information
5. Should exhibit at least one output

| Algorithm | Flowchart | Program |
|---|---|---|
| An algorithm is defined as **sequence of steps to solve a problem (task)**. | A **flowchart is pictorial (graphical) representation of an algorithm**. | Set of instructions. Instruction is a command to the computer to do some task. |
| | | |
| Algorithm can also be defined as a plan to solve a problem and represents its logic. | A picture is worth of 1000 words. We can understand more from picture than words. | Implementation of Algorithm or flowchart |

Different algorithms have different performance characteristics to solve the same problem. Some algorithms are fast. Some are slow. Some occupy more memory space. Some occupy less memory space. Some are complex and some algorithms are simple.

**Logically algorithm, flowchart and program are the same.**

## Flow chart Notations:

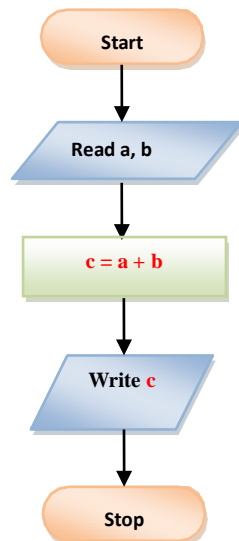| Name | Symbol | Function |
|---|---|---|
| Start/End | ⬭ | Used to markup the starting and ending point |
| Arrows | → | Used for connection |
| Input/Output | ▱ | Used for input and output information |
| Process | ▭ | Used to represent single step |
| Decision | ◇ | Used for branching or decision making |

# Examples of Algorithms and Flowcharts (with C code)

## 1. To find sum of two numbers

**Algorithm**                    **Flowchart**

1. Start
2. Read a, b
3. c = a + b
4. Print or display c
5. Stop

```
          Start
            |
         Read a, b
            |
         c = a + b
            |
         Write c
            |
          Stop
```

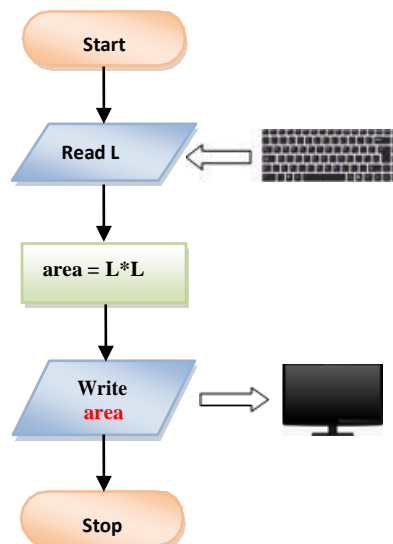## 2. Finding Area of the square

**Algorithm**                    **Flowchart**

1. Start
2. Read length, L
3. area = L*L
4. Print or display area
5. Stop

```
          Start
            |
         Read L          <=  (keyboard)
            |
        area = L*L
            |
         Write
         area            =>  (monitor)
            |
          Stop
```
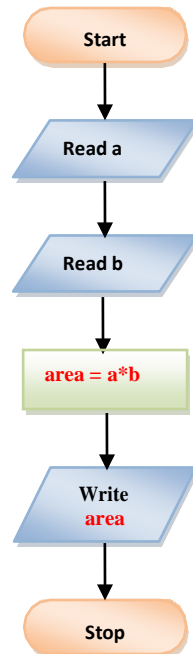
### 3. Finding Area of the rectangle

**Algorithm**         **Flowchart**

1. Start
2. Read side length, a
3. Read side length b
4. area = a*b
5. Print or display area
6. Stop
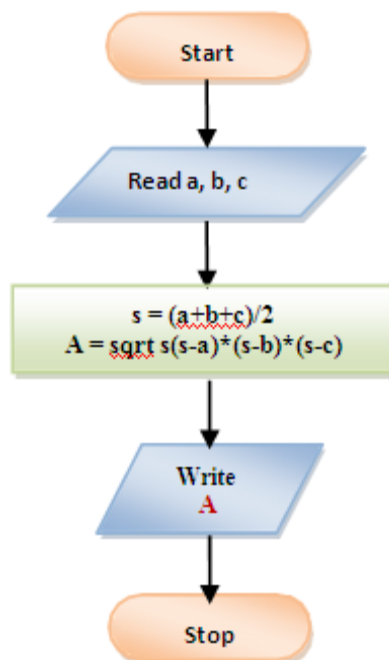
```
         ┌──────────┐
         │  Start   │
         └────┬─────┘
              ▼
         ┌──────────┐
         │  Read a  │
         └────┬─────┘
              ▼
         ┌──────────┐
         │  Read b  │
         └────┬─────┘
              ▼
         ┌────────────┐
         │ area = a*b │
         └────┬───────┘
              ▼
         ┌──────────┐
         │  Write   │
         │  area    │
         └────┬─────┘
              ▼
         ┌──────────┐
         │  Stop    │
         └──────────┘
```

### 4. Area of a triangle where three sides are given

**Algorithm**         **Flowchart**

1. Start
2. Read a, b, c
3. $s = (a+b+c)/2$
4. $A = sqrt\ (s*(s-a)*(s-b)*(s-c))$
5. Print or display A
6. Stop

```
         ┌──────────┐
         │  Start   │
         └────┬─────┘
              ▼
       ┌──────────────┐
       │ Read a, b, c │
       └──────┬───────┘
              ▼
   ┌───────────────────────────┐
   │ s = (a+b+c)/2             │
   │ A = sqrt s(s-a)*(s-b)*(s-c)│
   └──────────┬────────────────┘
              ▼
         ┌──────────┐
         │  Write   │
         │   A      │
         └────┬─────┘
              ▼
         ┌──────────┐
         │  Stop    │
         └──────────┘
```
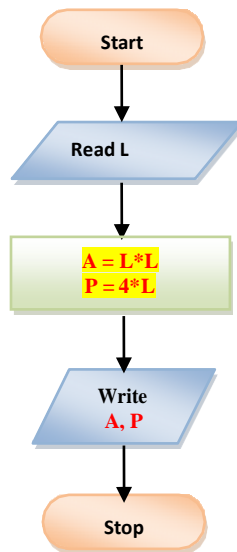
## 5. Find the area & perimeter of a square

**Algorithm**                    **Flowchart**

1. Start
2. Read length L
3. Area A = L*L
4. Perimeter P = 4*L
5. Print or display A,P
6. Stop

```
        ┌──────────┐
        │  Start   │
        └──────────┘
             │
             ▼
        ╱──────────╱
       ╱  Read L  ╱
      ╱──────────╱
             │
             ▼
        ┌──────────┐
        │ A = L*L  │
        │ P = 4*L  │
        └──────────┘
             │
             ▼
        ╱──────────╱
       ╱  Write   ╱
      ╱   A, P   ╱
             │
             ▼
        ┌──────────┐
        │  Stop    │
        └──────────┘
```
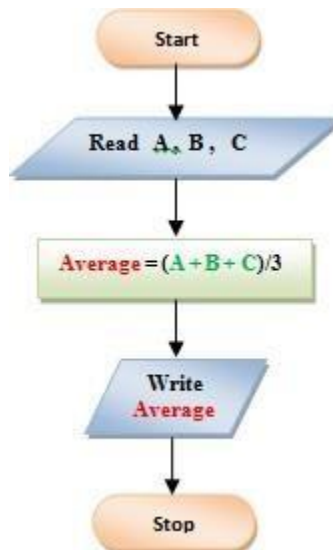
## 6. Calculating the average for 3 numbers

**Algorithm**                    **Flowchart**

1. Start
2. Read 3 numbers A, B, C
3. Calculate the average by the equation:
   Average = (A + B + C)/3
4. Print average
5. Stop

```
        ┌──────────┐
        │  Start   │
        └──────────┘
             │
             ▼
      ╱────────────────╱
     ╱  Read A, B, C  ╱
    ╱────────────────╱
             │
             ▼
      ┌────────────────────┐
      │ Average=(A+B+C)/3  │
      └────────────────────┘
             │
             ▼
        ╱──────────╱
       ╱  Write   ╱
      ╱ Average  ╱
             │
             ▼
        ┌──────────┐
        │  Stop    │
        └──────────┘
```
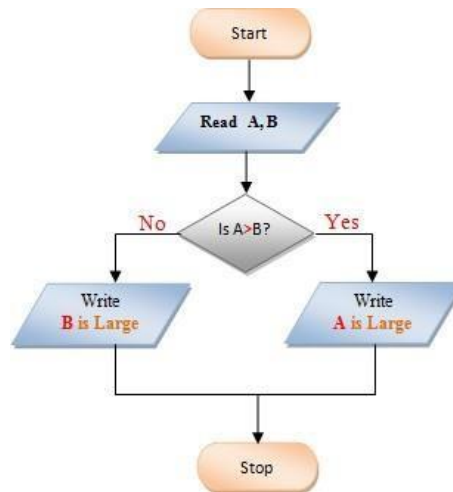
# 7. Greatest of two numbers

**Algorithm**                                          **Flowchart**

1. Start
2. Read A,B
3. If A > B then
        Print A is large
   else
        Print B is large
4. Stop

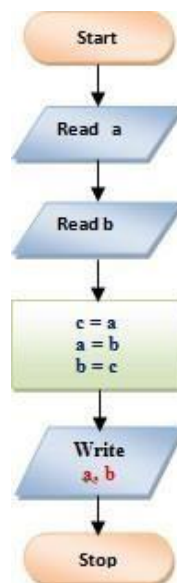# 8. Interchange the value of two numbers

**Algorithm**                                          **Flowchart**

1. Start
2. Read two values into two variables a, b
3. Declare third variable, c
        **c = a**
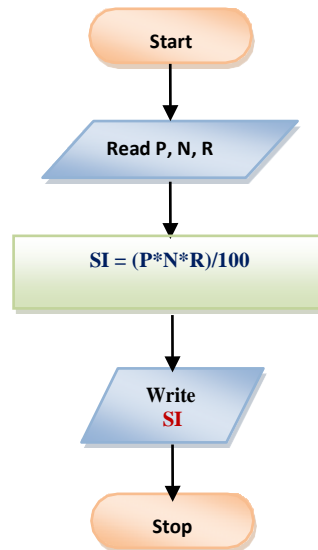        **a = b**
        **b = c**
4. Print or display a, b
5. Stop

## 9. Calculate simple interest using the expression (SI=PNR/100)

**Algorithm**                    **Flowchart**

1. Start
2. Read P, N, R
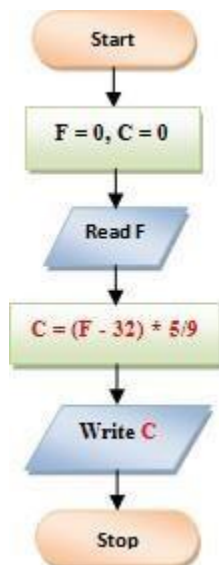3. SI=(PNR)/100
4. Print SI
5. Stop

Start

Read P, N, R

$$SI = (P*N*R)/100$$

Write
SI

Stop

## 10. Convert temperature from Fahrenheit to Celsius

**Algorithm**                    **Flowchart**

1. Start
2. Initialize F = 0, C = 0
3. Read F
4. C = (F-32) * 5/9
5. Write C
6. Stop

Start

$F = 0, C = 0$

Read F

$C = (F - 32) * 5/9$

Write C

Stop

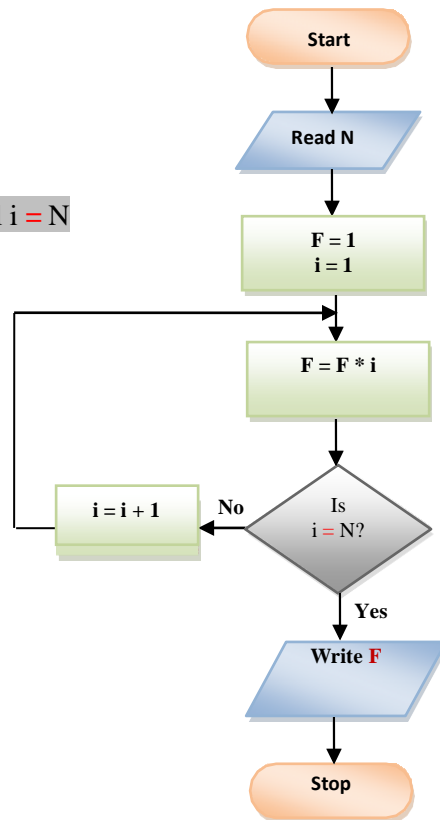## 11. Draw a flowchart for computing factorial N, where N! = 1 * 2 * 3 * …… N

**Algorithm**

1. Start
2. Read N
3. Initialize F =1, i = 1
4. F = F * i;
5. Increment i by 1
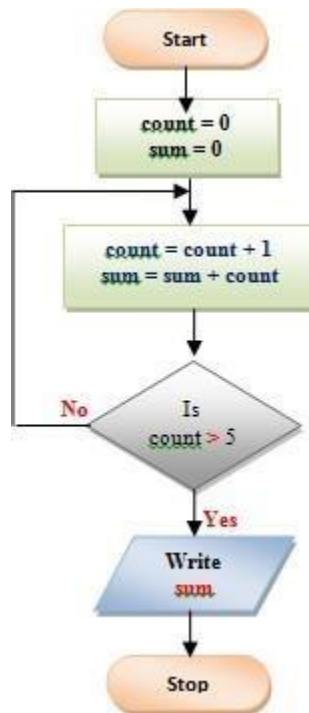6. Repeat step 4 & 5 until i = N
7. Print F
8. Stop

**Flowchart**

```
            Start
              |
              v
          Read N
              |
              v
          F = 1
          i = 1
              |
   ---------->|
   |          v
   |      F = F * i
   |          |
   |          v
 i = i + 1  No  / Is   \
   <--------/  i = N?  \
              \        /
                  | Yes
                  v
              Write F
                  |
                  v
              Stop
```

## 12.   Find the Sum of First Five Natural Numbers

**Algorithm**                                      **Flowchart**

1. Start
2. Initialize count = 0, sum = 0
3. count = count + 1
4. sum = sum + count
5. Repeat steps 3,4 until count > 5
6. Print sum
7. Stop



## 13.   Calculating sum of integers 1 to 100

**Algorithm**                                      **Flowchart**

1. Start
2. Initialize count i = 1, sum = 0
3. sum = sum + i
4. Increment i by 1
5. Repeat steps 3 & 4 until i > 100
6. Print sum
7. Stop
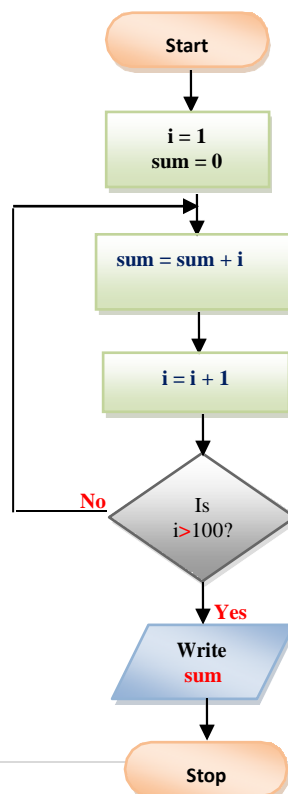
## 14. To find the sum of n natural Numbers

**Algorithm**                    **Flowchart**

1. Start
2. Read n
3. count=0
4. sum=0
5. count = count + 1
6. sum = sum + count
7. Repeat steps 5 & 6 until
   count > n
8. Print sum
9. Stop
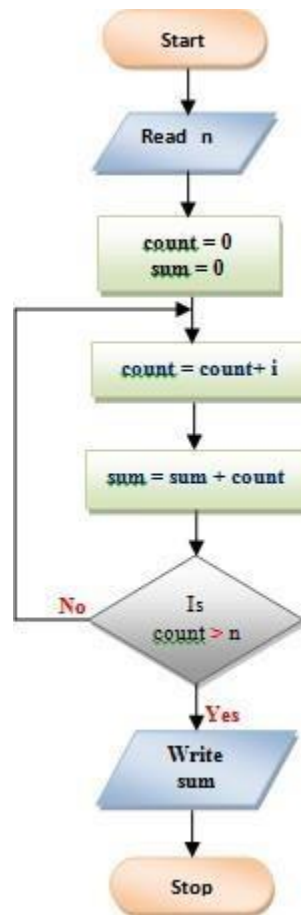
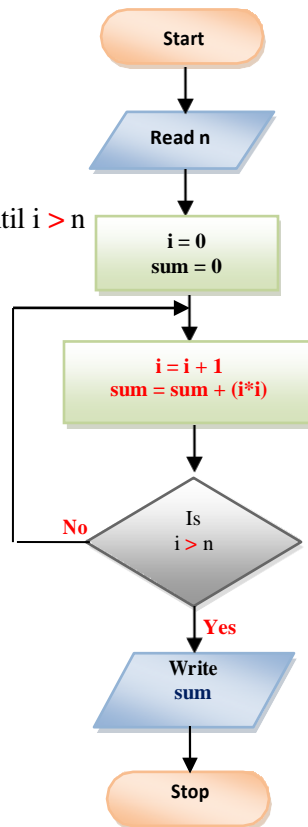# 15.   Sum of squares of n natural numbers

**Algorithm**

1. Start
2. Read n
3. i = 0, sum = 0
4. i = i + 1
5. sum = sum + (i*i)
6. Repeat steps 4 and 5 until i > n
7. Print sum
8. Stop

**Flowchart**

Start

Read n

i = 0
sum = 0

i = i + 1
sum = sum + (i*i)

Is
i > n

No

Yes

Write
sum
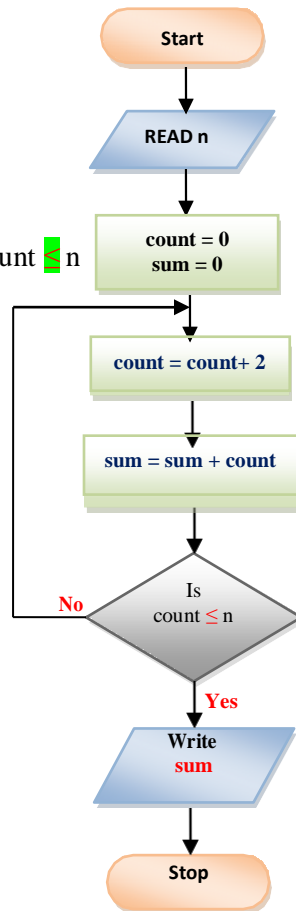
Stop

## 16.  To find the sum of all even numbers up to 'n'

**Algorithm**                    **Flowchart**

1. Start
2. Read n
3. count=0
4. sum=0
5. count = count + 2
6. sum = sum + count
7. Repeat steps 5 & 6 until count ≤ n
8. Print sum
9. Stop

Start

READ n

count = 0
sum = 0

count = count+ 2

sum = sum + count

Is
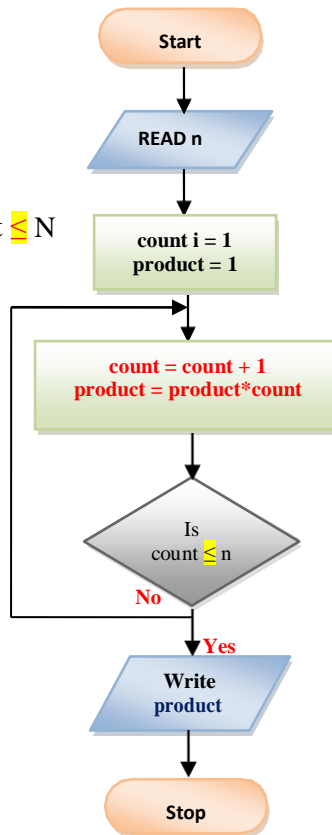count ≤ n

No

Yes

Write
sum

Stop

## 17.  To find Product of numbers up to N

### Algorithm

1. Start
2. Read n
3. count  i = 1
4. product = 1
5. product=count*product
6. count = count + 1
7. Repeat steps 5,6 until count $\leq$ N
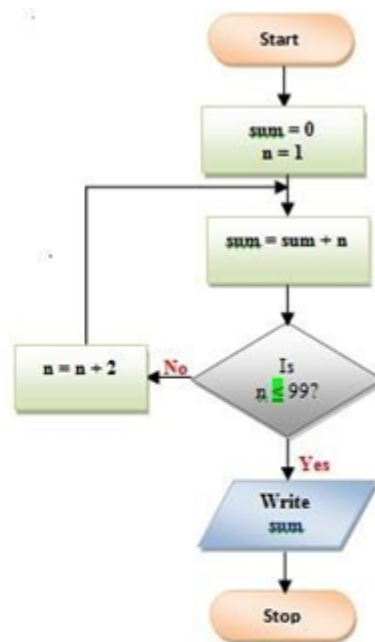8. Print product
9. Stop

### Flowchart

Start

READ n

count i = 1
product = 1

count = count + 1
product = product*count

Is
count $\leq$ n

No

Yes

Write
product

Stop

# 18. Sum of first 50 odd numbers

**Algorithm**                                    **Flowchart**

1. Start

2. sum=0, n = 1

3. sum=sum + n

4. n = n + 2

5. Repeat steps 4 and 5
   until n ≤ 99
6. Print sum
7. Stop

Current sum        Previous s

| Initial count = 0, sum = 0 | |
|---|---|
| count | sum = sum + cou |
| 1 | sum = 0 + 1 = 1 |
| 2 | sum = 1 + 2 = 3 |
| 3 | sum = 3 + 2 = 5 |
| 4 | sum = 5 + 2 = 7 |
| 5 | sum = 7 + 2 = 9 |
| Up to n<=99 | |

Start

sum = 0
n = 1

sum = sum + n

n = n + 2   ← No   Is n ≤ 99?

Yes

Write sum

Stop