ALGORITHMS AND FLOWCHARTS



ALGORITHMS AND FLOWCHARTS

A typical programming task can be divided into two phases:

Problem solving phase

- produce an ordered sequence of steps that describe solution of problem
- □ this sequence of steps is called an algorithm

Implementation phase

□ implement the program in some programming language



Steps in Problem Solving

- First produce a general algorithm (one can use pseudocode)
- Refine the algorithm successively to get step by step detailed algorithm that is very close to a computer language.
- Pseudocode is an artificial and informal language that helps programmers develop algorithms. Pseudocode is very similar to everyday English.

м

Pseudocode & Algorithm

Example 1: Write an algorithm to determine a student's final grade and indicate whether it is passing or failing. The final grade is calculated as the average of four marks.



Pseudocode & Algorithm

- Input a set of 4 marks
- Calculate their average by summing and dividing by 4
- if average is below 50 Print "FAIL" else Print "PASS"

M

Pseudocode & Algorithm

Detailed Algorithm

```
Step 1: Input M1,M2,M3,M4
```

Step 2: GRADE \leftarrow (M1+M2+M3+M4)/4

Step 3: if (GRADE < 50) then

Print "FAIL"

else

Print "PASS"

endif



The Flowchart

- (Dictionary) A schematic representation of a sequence of operations, as in a manufacturing process or computer program.
- (Technical) A graphical representation of the sequence of operations in an information system or program. Information system flowcharts show how data flows from source documents through the computer to final distribution to users. Program flowcharts show the sequence of instructions in a single program or subroutine. Different symbols are used to draw each type of flowchart.



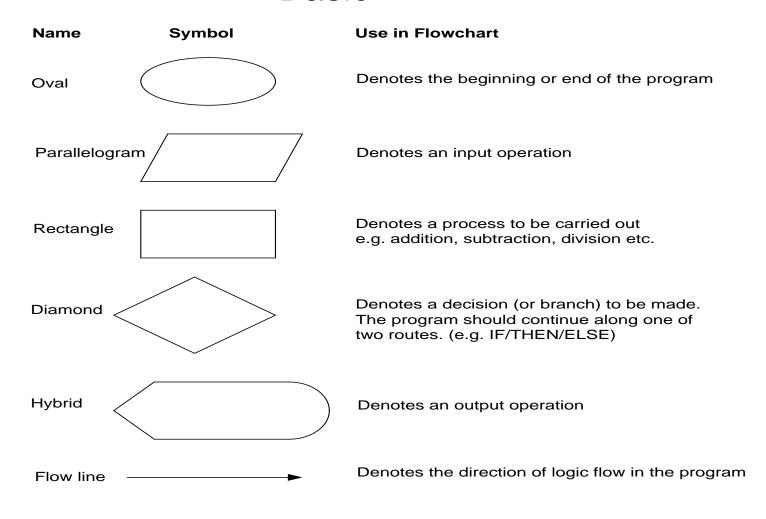
The Flowchart

A Flowchart

- □ shows logic of an algorithm
- emphasizes individual steps and their interconnections
- □ e.g. control flow from one action to the next

Flowchart Symbols

Basic



м

Pseudocode & Algorithm

Example 1: Write an algorithm to determine a student's final grade and indicate whether it is passing or failing. The final grade is calculated as the average of four marks.

М

Pseudocode & Algorithm

- Input a set of 4 marks
- Calculate their average by summing and dividing by 4
- if average is below 50 Print "FAIL" else Print "PASS"

м

Pseudocode & Algorithm

Detailed Algorithm

```
Step 1: Input M1,M2,M3,M4
```

```
Step 2: GRADE \leftarrow (M1+M2+M3+M4)/4
```

```
Step 3: if (GRADE < 50) then
```

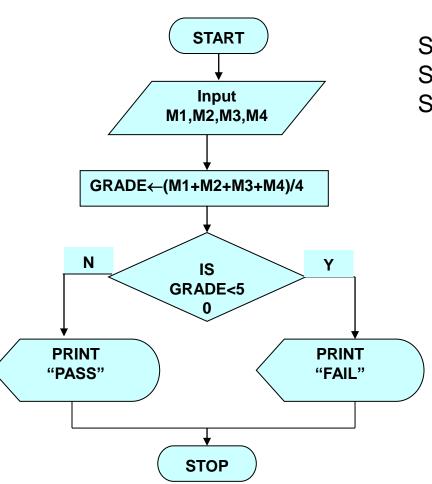
Print "FAIL"

else

Print "PASS"

endif





Step 1: Input M1,M2,M3,M4

Step 2: GRADE \leftarrow (M1+M2+M3+M4)/4

Step 3: if (GRADE <50) then

Print "FAIL"

else

Print "PASS"

endif



Write an algorithm and draw a flowchart to convert the length in feet to centimeter.

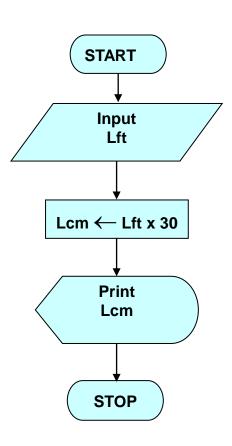
- Input the length in feet (Lft)
- Calculate the length in cm (Lcm) by multiplying LFT with 30
- Print length in cm (LCM)



Algorithm

- Step 1: Input Lft
- Step 2: Lcm ← Lft x 30
- Step 3: Print Lcm

Flowchart





Write an algorithm and draw a flowchart that will read the two sides of a rectangle and calculate its area.

- Input the width (W) and Length (L) of a rectangle
- Calculate the area (A) by multiplying L with W
- Print A

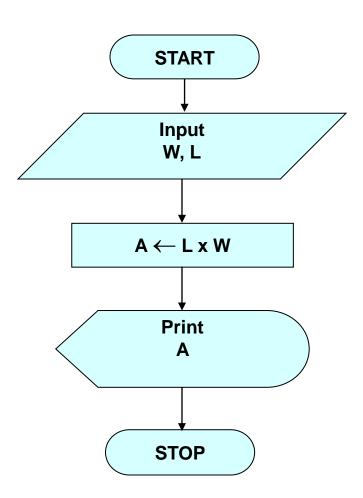


Algorithm

Step 1: Input W,L

Step 2: A ← L x W

Step 3: Print A



v

Example 4

- Write an algorithm and draw a flowchart that will calculate the roots of a quadratic equation $ax^2 + bx + c = 0$
- Hint: $\mathbf{d} = \operatorname{sqrt} (b^2 4ac)$, and the roots are: $\mathbf{x1} = (-b + d)/2a$ and $\mathbf{x2} = (-b d)/2a$

M

Example 4

- Input the coefficients (a, b, c) of the quadratic equation
- Calculate d
- Calculate x1
- Calculate x2
- Print x1 and x2

7

Example 4

Algorithm:

Step 1: Input a, b, c

■ Step 2: $d \leftarrow \text{sqrt} (b \times b - 4 \times a \times c)$

■ Step 3: $x1 \leftarrow (-b + d) / (2 \times a)$

■ Step 4: $x^2 \leftarrow (-b - d) / (2 \times a)$

■ Step 5: Print *x*1, *x*2

