

# Programming Fundamentals

## Pseudocode

# Introduction

- There are two commonly used tools to help to document program logic (the algorithm).
  - Flowcharts
  - Pseudo code
- Algorithm : Procedure for solving a problem as a set of actions to be executed and the order in which those actions are to be executed.
- Pseudo code : Informal language that helps programmers develop algorithms.

# Pseudocode

- It is easy to read and write, and allows the programmer to concentrate on the logic of the problem.
- Pseudocode is really structured English.
- It uses English that has been formalized and abbreviated to look like the high-level computer languages.

# Rules for Pseudocode

- 1) Write only one statement per line
- 2) Capitalize initial keyword
- 3) Indent to show hierarchy
- 4) End multiline structures
- 5) Keep statements language independent

# 1) One Statement Per Line

- Each statement in Pseudocode should express just one action for the computer.
- If the task list is properly drawn, then in most cases each task will correspond to one line of Pseudocode.

### Task List

**Input name, hours worked, rate of pay**

**Perform calculations**

**gross = hours worked \* rate of pay**

**Print name, hours worked, gross**

### Pseudocode

**INPUT name, hoursWorked, payRate**

**gross = hoursWorked \* payRate**

**PRINT name, hoursWorked, gross**

## 2) Indent to Show Hierarchy

- Each design structure uses a particular indentation pattern
  - Sequence:
    - Keep statements in sequence all starting in the same column

- Selection:
  - Indent statements that fall inside selection structure, but not the keywords that form the selection
- Loop: (Repetition)
  - Indent statements that fall inside the loop but not keywords that form the loop

```
INPUT name, grossPay, taxes
IF taxes > 0
    net = grossPay – taxes
ELSE
    net = grossPay
ENDIF
PRINT name, net
```



### 3) End Multiline Structures

```
INPUT name, grossPay, taxes
```

```
IF taxes > 0
```

```
    net = grossPay – taxes
```

```
ELSE
```

```
    net = grossPay
```

```
ENDIF
```

```
PRINT name, net
```

- ▶ See the IF/ELSE/ENDIF as constructed above, the ENDIF is in line with the IF.
- ▶ The same applies for WHILE/ENDWHILE etc...

## 4) Language Independence

- Resist the urge to write in whatever language you are most comfortable with, in the long run you will save time.
- Remember you are describing a logic plan to develop a program, you are not programming!

# Basic Elements

- Structure

Start

Actions

end

How to do in pseudo code?

1. Variables
2. Input Data
3. Output data
4. Perform Arithmetic operations
5. Control Structures

# Variables

- Creating a variable
  - Declare var\_name
- Assigning a value
  - Key words :
    - SET, =, STORE, INITIALIZE
  - Var\_name=value

Eg 1:

Declare age

Set age to 20

or

age=20

Eg2:

Declare mark

mark=75

# Input data

- Key words in pseudo-code
  - Input/Read/Get

Eg:

- Input marks
- Read student name
- Get system date
- Read number\_1, number\_2
- Get tax-code

# Output data

- Key words used in pseudo-code
  - **Print**, **Write**, Put, **Output** or **Display**
- Eg:
  - Print 'Program Completed'
  - Write customer record to master file
  - Output name, address and postcode
  - Output total-tax
  - Display 'End of data'

# Perform Arithmetic Operations

- write a mathematical calculation or formula either actual mathematical symbols or the words for those symbols can be used.

Eg:

- Add one to total
- $\text{one} = \text{one} + \text{number}$

# Control Structures

In programming there are 3 main Control Structures.

1. Sequence Control Structure
2. Selection Control Structure (Decision)
3. Iteration or Repetition Structure

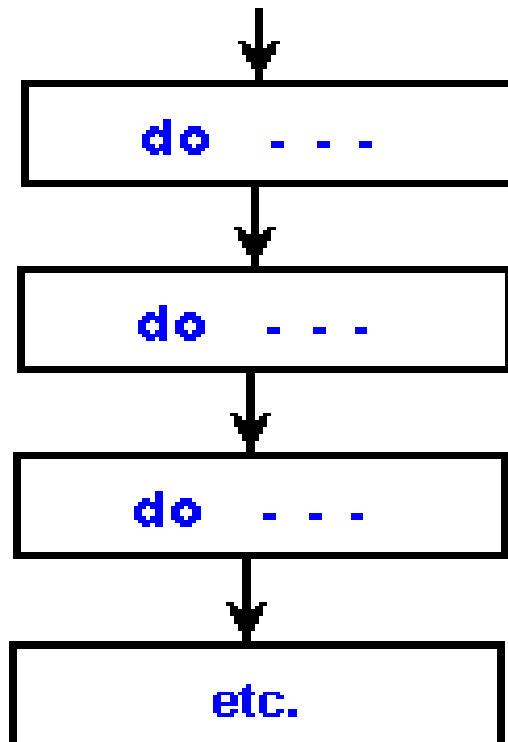


# 1) Sequence Control Structure

- The statements in code are sequential (even when the sequences are run in parallel or concurrently).
- The programme statements are executing one after the other
- Eg: Continue

# Sequence Control Structure

## Flowchart



Pseudocode →

## Structured English

1st statement

2nd statement

3rd statement

etc.

# Exercise 1

- Write a pseudocode to input two numbers and output the total.

# Exercise 1

- Write a pseudocode to input two numbers and output the total.

```
START  
DECLARE no1, no2, sum  
INPUT no1, no2  
sum = no1 + no2  
PRINT sum  
END
```

## Exercise 2

- Write a pseudocode to input height and width of a rectangle and output the area and perimeter.

area = height x width

perimeter = 2 x ( height + width )

## Exercise 2

- Write a pseudocode to input height and width of a rectangle and output the area and perimeter.

area = height x width

perimeter = 2 x ( height + width )

START

DECLARE h, w, area, p

INPUT h, w

area = h \* w

p = 2 \* (h + w)

PRINT area, p

END

## Exercise 3

- Write a pseudocode to input hours and minutes and output the total minutes.

E.g. 1 hour 30 minutes = 90 minutes

## Exercise 3

- Write a pseudocode to input hours and minutes and output the total minutes.

E.g. 1 hour 30 minutes = 90 minutes

```
START  
DECLARE h, m, total  
INPUT h, m  
total = (h * 60) + m  
PRINT total  
END
```



## Exercise 4

- Write a pseudocode to input item quantity, unit price and display the item amount.

## Exercise 4

- Write a pseudocode to input item quantity, unit price and display the item amount.

```
START  
DECLARE unit, qty, amount  
INPUT unit, qty  
amount = unit * qty  
PRINT amount  
END
```

# Selection Control Structure

- Keywords used in pseudo code:
  - IF, THEN, ELSE IF ,ELSE and END IF

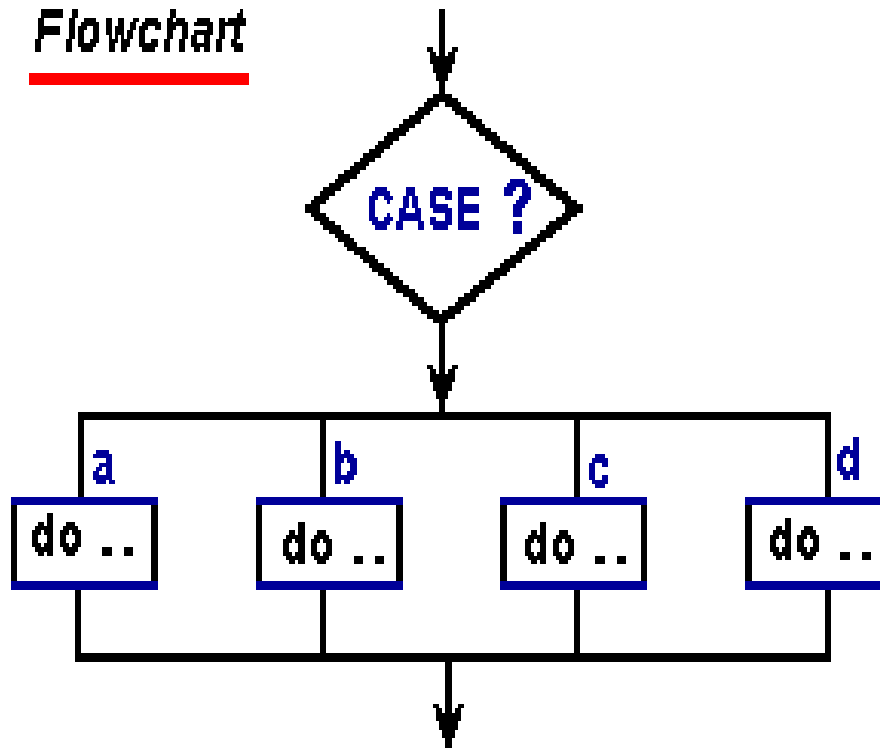
```
IF (condition) THEN
    true actions
ELSE
    false actions
END IF
```

```
IF (condition) THEN
    action 1
ELSE IF (condition) THEN
    action 2
Else
    action 3
END IF
```

# Selection Control Structure

Pseudocode →

Flowchart



Structured English

**CASE**

**a** do statement

**b** do statement

**c** do statement

**d** do statement

**ENDCASE**

## Exercise 5

- Write a pseudocode to input a number and display whether it is odd or even.

## Exercise 5

- Write a pseudocode to input a number and display whether it is odd or even.

```
START  
DECLARE no  
INPUT no  
IF (no % 2 == 0) THEN  
    PRINT "Even"  
ELSE  
    PRINT "Odd"  
END IF  
END
```

## Exercise 6

- Write a pseudocode to input two numbers and display the highest among them.

## Exercise 6

- Write a pseudocode to input two numbers and display the highest among them.

```
START  
DECLARE no1, no2  
INPUT no1, no2  
IF (no1 > no2) THEN  
    PRINT no1  
ELSE  
    PRINT no2  
END IF  
END
```



## Exercise 7

- Write a pseudocode to input a number and display whether it is positive, negative or zero.

## Exercise 7

- Write a pseudocode to input a number and display whether it is positive, negative or zero.

```
START
DECLARE no
INPUT no
IF (no > 0) THEN
    PRINT "Positive"
ELSE IF (no < 0) THEN
    PRINT "Negative"
ELSE
    PRINT "Zero"
END IF
END
```

## Exercise 8

- Write a pseudocode to input a mark a student has obtained and display the grade.

<u>Mark</u>	<u>Grade</u>
mark < 40	F
40 <= mark < 50	C
50 <= mark < 70	B
mark >= 70	A

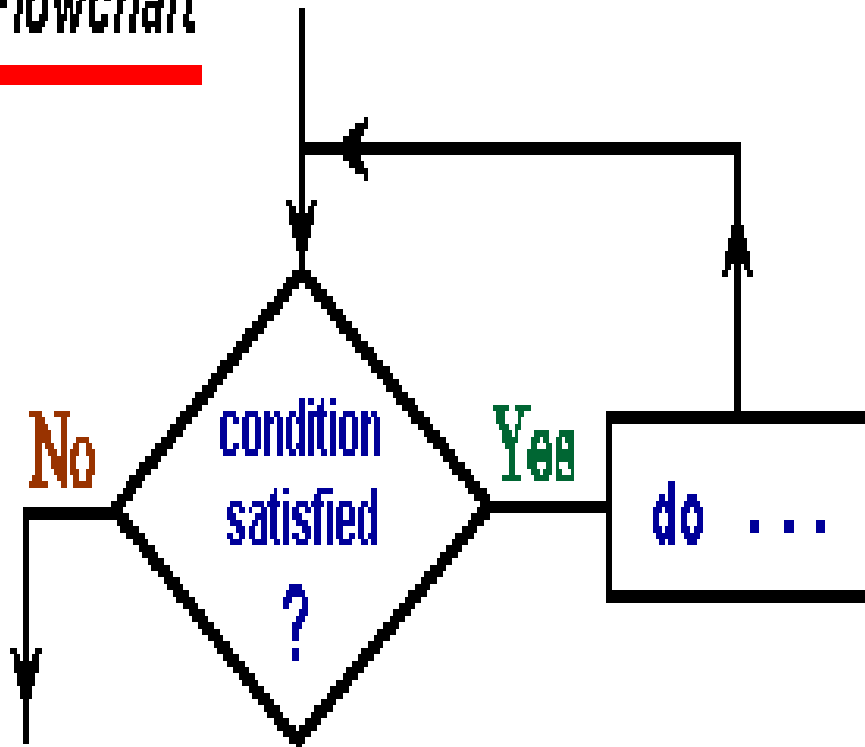
```
START
DECLARE mark
INPUT mark
IF (mark < 40) THEN
    PRINT "F"
ELSE IF (mark < 50) THEN
    PRINT "C"
ELSE IF (mark < 70) THEN
    PRINT "B"
ELSE
    PRINT "A"
END IF
END
```

### 3) Iteration or Repetition Structure

- This is used when a given sequence is repeated.
- In this structure it repeats a group of steps in the pseudocode.
- Also program statements are repeating over & over again until one condition becomes true or false.

# Iteration or Repetition Structure

## Flowchart



## Structured English

```

WHILE condition
do statement
ENDWHILE
  
```

## Exercise 9

- Write a pseudocode to display the following number sequence.

1      2      3      4      5      6      7      8      9      10

## Exercise 9

- Write a pseudocode to display the following number sequence.

1      2      3      4      5      6      7      8      9      10

```
START  
DECLARE count=1  
WHILE (count <= 10)  
    PRINT count  
    count = count + 1  
END WHILE  
END
```



## Exercise 10

- Write a pseudocode to display the following number sequence.

10    9    8    7    6    5    4    3    2    1

## Exercise 10

- Write a pseudocode to display the following number sequence.

10    9    8    7    6    5    4    3    2    1

```
START
DECLARE count=10
WHILE (count >= 1)
    PRINT count
    count = count - 1
END WHILE
END
```

## Exercise 11

- Write a pseudocode to input 10 numbers and display the total and average of them.

## Exercise 11

- Write a pseudocode to input 10 numbers and display the total and average of them.

```
START
DECLARE count=1, no, total=0, avg
WHILE (count <= 10)
    INPUT no
    total = total + no
    count = count + 1
END WHILE
avg = total / 10
PRINT total, avg
END
```

## Exercise 12

- Write a pseudocode to input 100 numbers and display the number of odd and even numbers in the series.

```
START
DECLARE count=1, no, oc=0, ec=0
WHILE (count <= 100)
    INPUT no
    IF (no % 2 == 0) THEN
        ec = ec + 1
    ELSE
        oc = oc + 1
    END IF
    count = count + 1
END WHILE
PRINT oc, ec
END
```

## Exercise 13

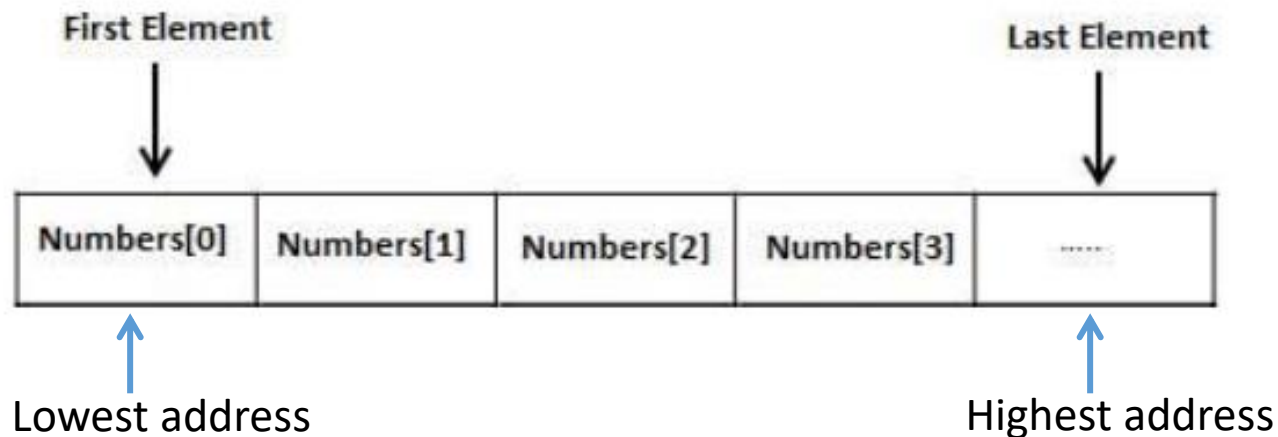
- Write a pseudocode to input a series of numbers which terminates by -1 and display the number of positive, negative and zeros in the series.

```
START
DECLARE no, pc=0, nc=0, zc=0
INPUT no
WHILE (no != -1)
    IF (no > 0) THEN
        pc = pc + 1
    ELSE IF (no < 0) THEN
        nc = nc + 1
    ELSE
        zc = zc + 1
    END IF
    INPUT no
END WHILE
PRINT pc, nc, zc
END
```



# Array

- An array is a data structure which can store a fixed size sequential collection of elements of the same type.
- A specific element in an array is accessed by an index.
- All arrays consists of contiguous memory locations.



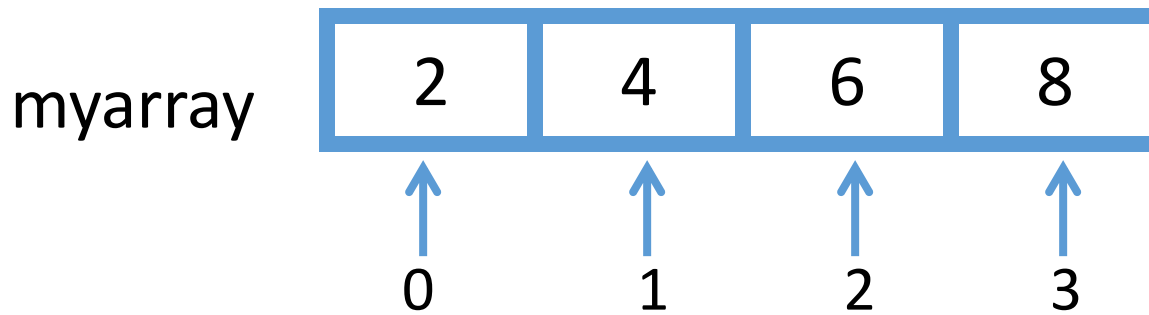
# Index of an Array



Index      0                  1                  2                  3

- Index of the array start with 0 and the last index will be (array size - 1).

# Example



## Exercise 14

- Write a pseudocode to input 10 numbers into an array and output the sum (use one loop to input values and another to find the sum).

## Exercise 15

- Write a pseudocode to input 5 values into an array. Display the maximum value (use one loop to input values and another to find the maximum).

## Exercise 16

- Write a pseudocode to input 5 values into an array. Display the minimum value (use one loop to input values and another to find the minimum).

# Question 1

- ABC company produces and sells handbags directly to their customers. Write a pseudocode to input number of bags sold each day into an array and output the number of days that the company had sales for more than 10 bags. Consider the sales for the month of January.

```
START
DECLARE count=1, no, i=0, array[], days=0
WHILE (count <= 31)
    INPUT no
    array [i] = no
    count = count + 1
    i= i + 1
END WHILE
i=0
WHILE (i <= 30)
    IF (array[i] > 10) THEN
        days = days + 1
    END IF
    i= i + 1
END WHILE
PRINT days
END
```



## Question 2

- A bank offers loans for selected customers based on their criteria. Write a pseudocode to input requested loan amount and annual interest rate and to output the total amount to be paid at the end of the year.
- (*total amount = loan + interest*)

## Question 2

- A bank offers loans for selected customers based on their criteria. Write a pseudocode to input requested loan amount and annual interest rate and to output the total amount to be paid at the end of the year.
- (*total amount = loan + interest*)

```
START  
DECLARE amount, rate, total, interest  
INPUT amount, rate  
interest = amount * rate  
total = amount + interest  
PRINT total  
END
```

## Question 3

- Write a pseudocode to input employee name, hours worked and rate per hour. Display the employee and payment by calculating the payment using one of the following formulas.

### Hours worked

Hours worked  $\leq 40$

Hours worked  $> 40$

### formula

$hw * rh$

$40 * rh + (hw - 40) * 1.5 * rh$

START

DECLARE name, hw, rh, payment

INPUT name, hw, rh

IF (hw <= 40) THEN

    payment = hw \* rh

ELSE

    payment = 40 \* rh + (hw - 40) \* 1.5 \* rh

END IF

PRINT name, payment

END