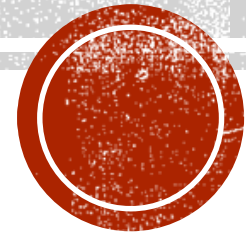


DATA STRUCTURES & ALGORITHMS

Lecture 03
Fahad Zafar



ALGORITHMS - FINAL DEFINITION

An algorithm is a **finite** set of **unambiguous, executable** instructions that directs a **terminating** activity.



ALGORITHM REQUIREMENTS

- **Requirements for an algorithm:**
 - Input
 - Output
 - Unambiguous
 - Generality
 - Correctness
 - Finite
 - Efficiency



ALGORITHM REPRESENTATION

- Pseudo-code
- Flow chart



PSEUDO-CODE

- Pseudo-code is a semi-formal, English-like language with a limited vocabulary that can be used to design and describe algorithms.
- The main purpose of a pseudo-code is to define the procedural logic of an algorithm in a simple, easy-to-understand manner for its readers, who may or may not be proficient in computer programming.



PSEUDO-CODE

- Used in designing algorithms.
- Used in communicating to users.
- Used in implementing algorithms as programs.
- Used in debugging logic errors in programs.
- Used in documenting programs for future maintenance and expansion purposes.



PSEUDO-CODE

- Must have a limited vocabulary.
- Must be easy to learn.
- Must produce simple, English-like narrative notation.
- Must be capable of describing all algorithms, regardless of their complexity.



CONTROL STRUCTURES

- Sequence
- Selection
- Repetition



SEQUENCE

- Series of steps or statements that are executed in the order they are written.
- Example:
 - Read taxable income
 - Read filing status
 - Compute income tax
 - Print income tax



SELECTION

- Defines one or two courses of action depending on the evaluation of a condition.
- A condition is an expression that is either true or false.
- Example
 - if condition (is true)
 - then-part
 - else
 - else-part
 - end_if



NESTED SELECTION

```
if status is equal to 1
    print "Single"
else
    if status is equal to 2
        print "Married filing jointly"
    else
        if status is equal to 3
            print "Married filing separately"
        end_if
    end_if
end_if
```



REPETITION

- Specifies a block of one or more statements that are repeatedly executed until a condition is satisfied.

- Example:

```
while condition
```

```
    loop-body
```

```
end_while
```



CONVENTIONS

- Each pseudo-code statement consists of keywords that describe operations and some appropriate, English-like description of operands.
- Each statement should be written on a separate line.
Continuation lines should be indented



CONVENTIONS II

- Sequence statements should begin with unambiguous words (compute, set, initialize).
- Selection statements then-part and else-part should be indented.
- Selection statements end with the keyword `end_if`.



CONVENTION III

- Repetition statements end with `end_while`.
- Loop-bodies are indented.
- All words in a pseudo-code statement must be chosen to be unambiguous, and as easy as possible to understand by non-programmers.
- Enclose comments between `/*` and `*/`



EXAMPLE 1

If student's grade is greater than or equal to 60

Print "passed"

else

Print "failed"

End_if



EXAMPLE 2 — CLASS AVERAGE

Set total to zero

Set grade counter to one

While grade counter is less than or equal to ten

 Input the next grade

 Add the grade into the total

End_while

Set the class average to the total divided by ten

Print the class average.



BASIC SYMBOLS



Rounded box - use it to represent an event which occurs automatically. Such an event will trigger a subsequent action, for example 'receive telephone call', or describe a new state of affairs.



Rectangle or box - use it to represent an event which is controlled within the process. Typically this will be a step or action which is taken. In most flowcharts this will be the most frequently used symbol.



Diamond - use it to represent a decision point in the process. Typically, the statement in the symbol will require a 'yes' or 'no' response and branch to different parts of the flowchart accordingly.



Circle - use it to represent a point at which the flowchart connects with another process. The name or reference for the other process should appear within the symbol.



FLOWCHART

- A flowchart is a diagrammatic representation that illustrates the sequence of operations to be performed to get the solution of a problem

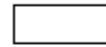


GUIDE LINES

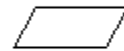
- Flowcharts are usually drawn using some standard symbols; however, some special symbols can also be developed when required



Start or end of the program. Computational steps or processing function of a program



Input or output operation, Decision making and branching
Connector or joining of two parts of program



Magnetic Tape
Off-page connector



Flow line



Annotation

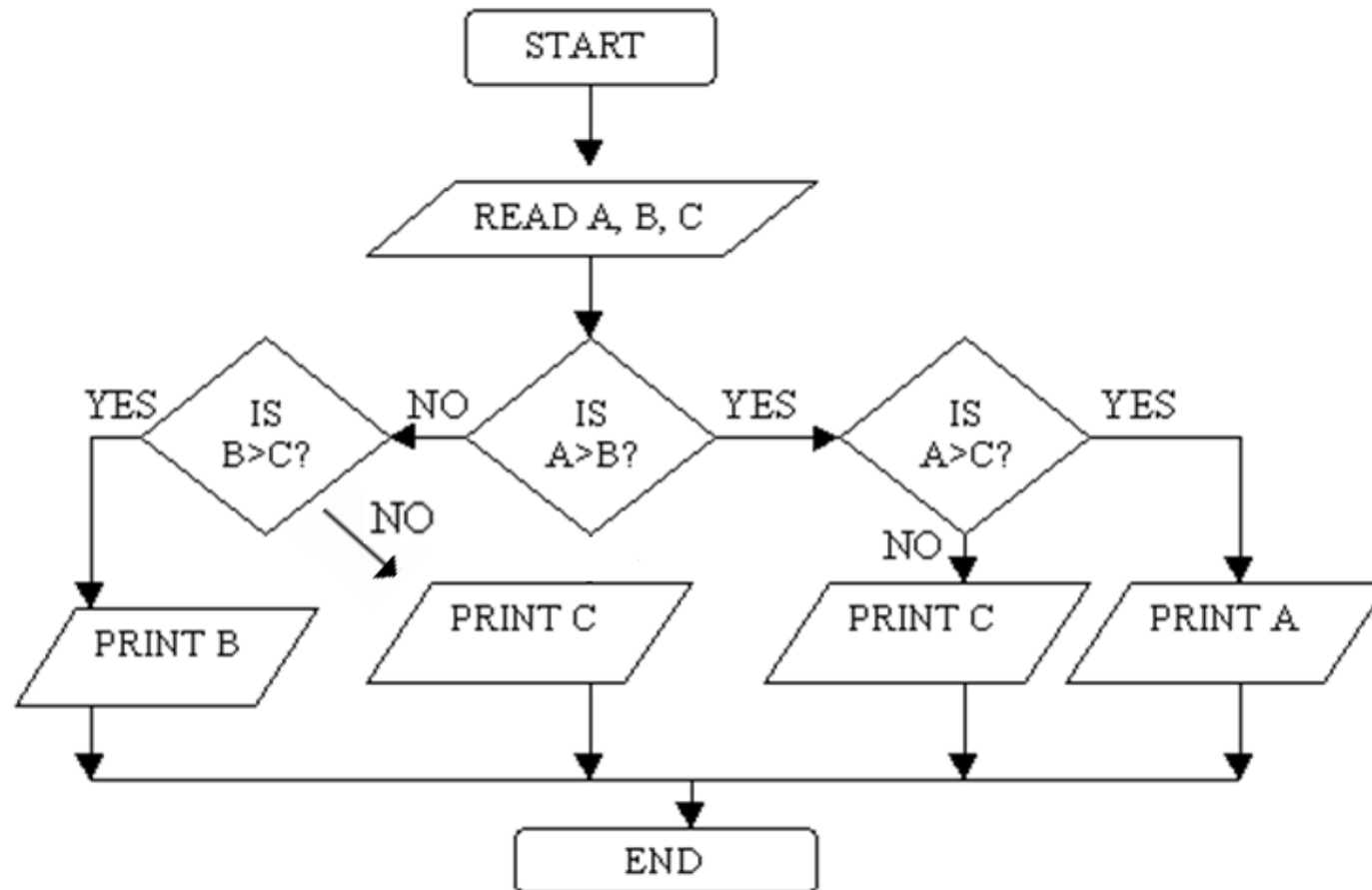


Display



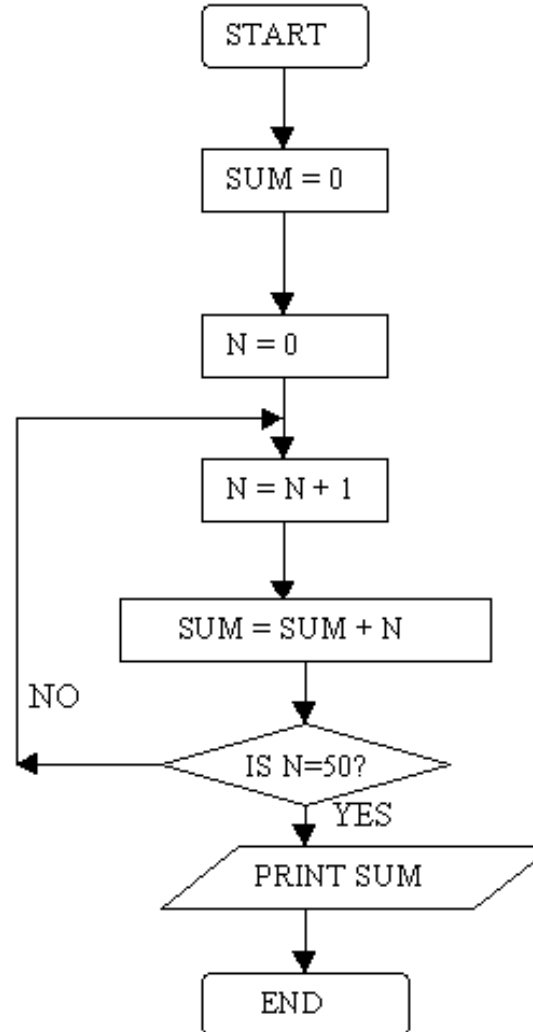
EXAMPLE

- Draw a flowchart to find the largest of three numbers A,B, and C.



EXAMPLE

- Draw a flowchart to find the sum of first 50 natural numbers.



THANKS!!

