



Introduction to Flowcharting

This slide is based upon Appendix C from *Starting Out with C++: From Control Structures to Objects* (5th Edition), Tony Gaddis 2007, Published by Addison-Wesley

1



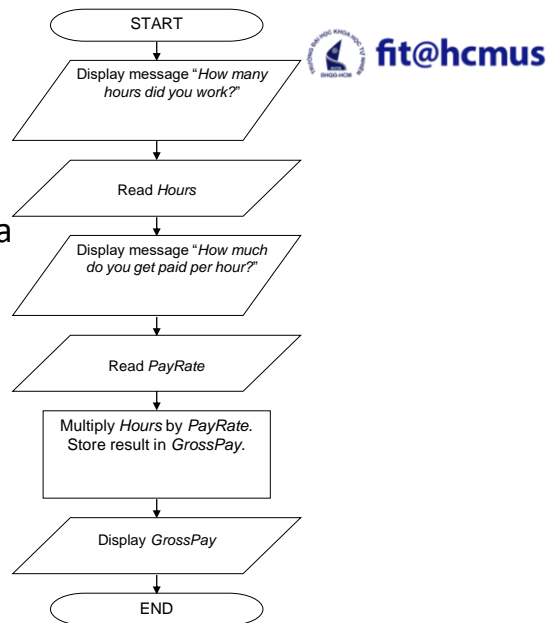
Contents

- Flowchart
- Basic flowchart symbols
- Stepping through the flowchart
- Three flowchart structures
- Examples
- Exercises

2

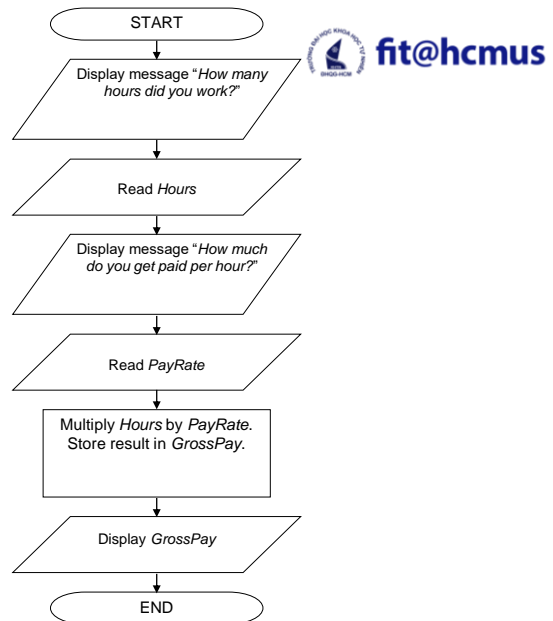
What is a Flowchart?

- A flowchart is a diagram that depicts the “flow of control” of a program.



What is a Flowchart?

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



Basic Flowchart Symbols

- Three types of symbols in this flowchart:
 - rounded rectangle
 - parallelogram
 - rectangle
- Each symbol represents a different type of operation.

Basic Flowchart Symbols

- Terminals
 - represented by rounded rectangles
 - indicate a starting or ending point

START

END

Basic Flowchart Symbols

- Input/Output Operations
 - represented by parallelograms
 - indicate an input or output operation

Display message
"How many hours
did you work?"

Read Hours

Basic Flowchart Symbols

- Processes
 - represented by rectangles
 - indicates a process such as a mathematical computation or variable assignment

Multiply *Hours* by *PayRate*.
Store result in *GrossPay*.

Three Flowchart Structures

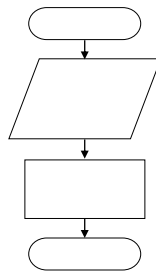
Sequence

Selection

Iteration

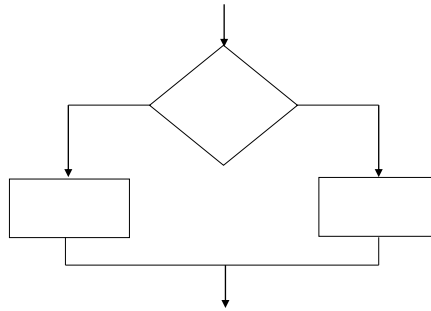
Sequence Structure

- A series of actions are performed in sequence
- The pay-calculating example was a sequence flowchart.



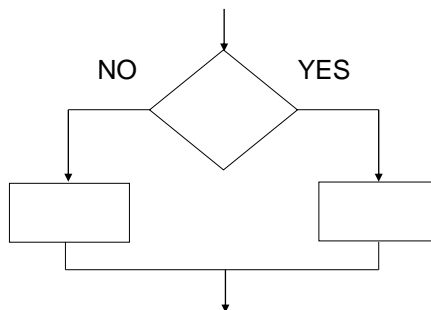
Selection Structure

- One of two possible actions is taken, depending on a condition.



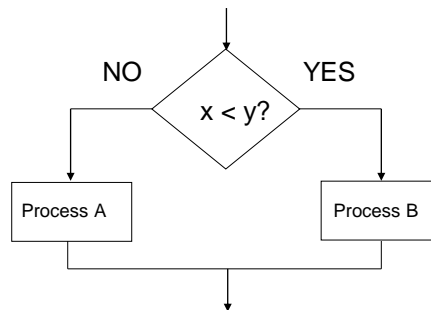
Selection Structure

- A new symbol, the **diamond**, indicates a **yes/no question**.
 - If the answer to the question is yes, the flow follows one path.
 - If the answer is no, the flow follows another path



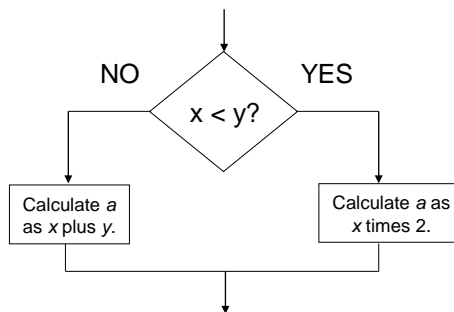
Selection Structure

- In the flowchart segment below, the question “is $x < y$?” is asked.
 - If the answer is no, then process A is performed.
 - If the answer is yes, then process B is performed.



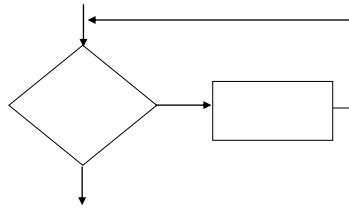
Selection Structure

- The flowchart segment below shows a decision structure is expressed in C++ as an if/else statement.



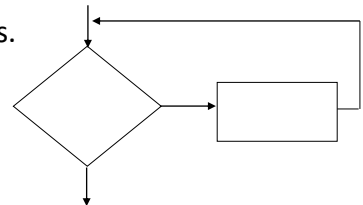
Iteration Structure

- An **iteration** structure represents part of the program that **repeats**. This type of structure is commonly known as a **loop**.



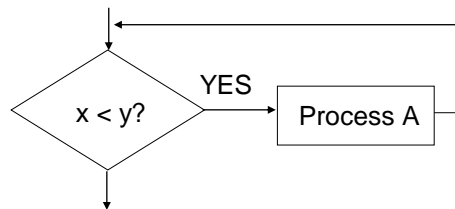
Iteration Structure

- Notice the use of the diamond symbol.
- A loop tests a condition,
 - and if the condition exists, it performs an action.
 - Then it tests the condition again. If the condition still exists, the action is repeated.
 - This continues until the condition no longer exists.



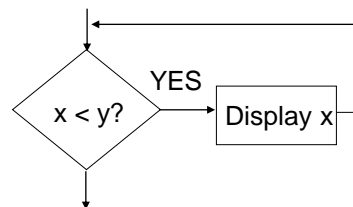
Iteration Structure

- In the flowchart segment, the question “is $x < y$?” is asked.
 - If the answer is yes, then **Process A** is performed.
 - The question “is $x < y$?” is asked again. Process A is repeated as long as x is less than y .
 - When x is no longer less than y , the iteration stops and the structure is exited.



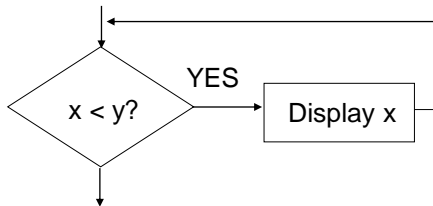
Controlling an Iteration Structure

- The action performed by an iteration structure **MUST** eventually cause the loop to terminate.
- Otherwise, an **infinite loop** is created.
- In this flowchart segment, x is never changed. Once the loop starts, it will never end.



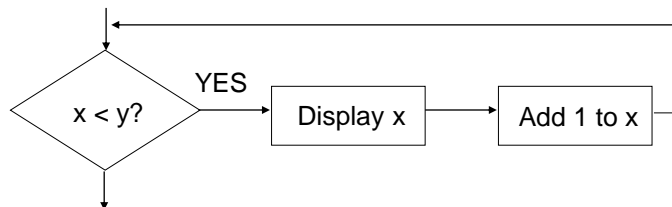
Controlling an Iteration Structure

- x is never changed. Once the loop starts, it will never end.
- **How can this flowchart be modified so it is no longer an infinite loop?**



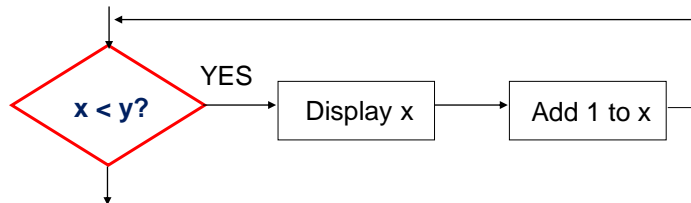
Controlling an Iteration Structure

- By adding an action within the iteration that **changes the value of x** .



A Pre-Test Iteration Structure

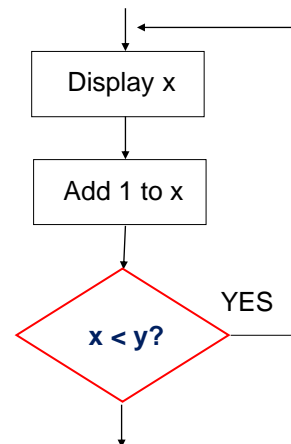
- This type of structure is known as a **pre-test iteration structure**. The condition is tested **BEFORE** any actions are performed.



- if the condition does not exist, the loop will never begin.

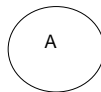
A Post-Test Iteration Structure

- This flowchart segment shows a **post-test iteration structure**.
- The condition is tested **AFTER** the actions are performed.
- A post-test iteration structure always performs its actions **at least once**.



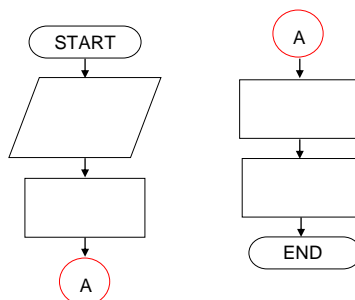
Connectors

- Sometimes a flowchart will not fit on one page.
- A **connector** (represented by a small circle) allows you to connect two flowchart segments.



Connectors

- The “A” connector indicates that the second flowchart segment begins where the first segment ends.



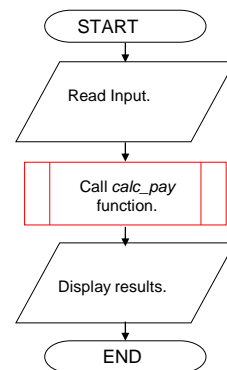
Modules

- A program module, such as a subprogram (or function in C++), is represented by a special symbol.



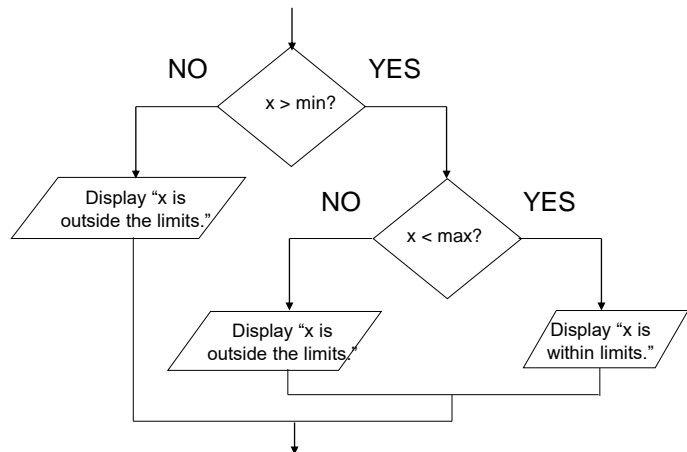
Modules

- The position of the module symbol indicates the point the module is executed.
- A separate flowchart can be constructed for the module.



Combining Structures

- This flowchart segment shows two selection structures combined.



Examples

Example 01

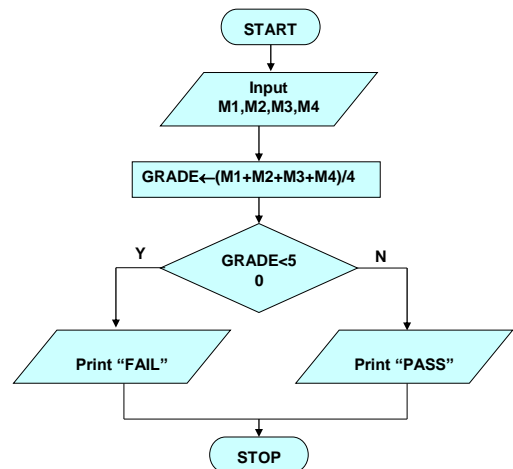
```

Step 1:   Input M1,M2,M3,M4
Step 2:   GRADE ← (M1+M2+M3+M4) / 4
Step 3:   if (GRADE < 50) then
            Print "FAIL"
        else
            Print "PASS"
        endif
    
```

Example 01

```

Step 1:   Input M1,M2,M3,M4
Step 2:   GRADE ← (M1+M2+M3+M4) / 4
Step 3:   if (GRADE < 50) then
            Print "FAIL"
        else
            Print "PASS"
        endif
    
```



Example 02

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

- Pseudocode:

```

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

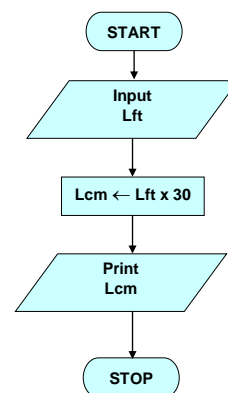
Example 02

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

- Pseudocode:

```

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



Example 03

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

Example 03

- Write an algorithm and draw a flowchart that will calculate the roots of a quadratic equation.

Example 03

- Write an algorithm and draw a flowchart that will calculate the roots of a quadratic equation.

- Hint:

$d = \sqrt{b^2 - 4ac}$,

if $d < 0$, there is no root.

else if $d == 0$, $x_1 = x_2 = -b/2a$

else the roots are: $x_1 = (-b + d)/2a$ and $x_2 = (-b - d)/2a$

Example 04

- Write an algorithm that reads two values, determines the largest value and prints the largest value with an identifying message.

Example 05

- Write an algorithm that reads **three** numbers and prints the value of the largest number.

Exercises

Exercises

- Check whether an input year is a *leap* year.
 - Ref: https://en.wikipedia.org/wiki/Leap_year#Algorithm

Exercises

- Electricity cost calculator:
 - Ref: <https://www.evn.com.vn/c3/evn-va-khach-hang/Bieu-gia-ban-le-dien-9-79.aspx>
 - Ref: <https://www.evn.com.vn/c3/calc/Cong-cu-tinh-hoa-don-tien-dien-9-172.aspx>

d) Sinh hoạt

TT	Nhóm đối tượng khách hàng	Giá bán điện (đồng/kWh)
1	Giá bán lẻ điện sinh hoạt	
	Bậc 1: Cho kWh từ 0 - 50	1.678
	Bậc 2: Cho kWh từ 51 - 100	1.734
	Bậc 3: Cho kWh từ 101 - 200	2.014
	Bậc 4: Cho kWh từ 201 - 300	2.536
	Bậc 5: Cho kWh từ 301 - 400	2.834
	Bậc 6: Cho kWh từ 401 trở lên	2.927

Questions and Answers

fit@hcmus | Programming 1 | 2023

51