Programming Basics – Part 2

CS1A

- Selection
  - IF-THEN
  - IF-THEN-ELSE
- Repetition
  - FOR Loops
  - WHILE Loops

Control / Logic Structures

All modern programming languages are based on 3 basic control structures

- Sequence
  - Instructions are executed one after another in the order they appear in the program
  - Until another control structure takes precedence
- Selection
  - Based on some condition, either one part of the program is executed or another part is executed
  - The program chooses which part to execute based on the condition
- Repetition
  - Part of the code is executed over and over (repeated)
  - This can be for a set number of times or until a condition is met

Selection Structures

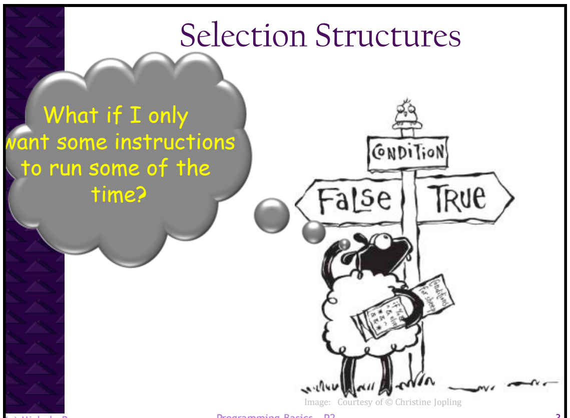

What if I only want some instructions to run some of the time?
CONDITION
False
TRUE

Image: Courtesy of © Christine Jopling

Selection Operators

Selection
→ Choosing between two or more alternative actions

- Alter the sequential flow of the instructions in a program
- Based on a Boolean Expression
  - An expression that evaluates to 1 of 2 possibilities
  - Either True or False
- The computer evaluates a Boolean Expression and determines which instruction to execute based on the result
- Boolean expressions are formed using relational operators

Relational Operators

Symbol	Name
<	Less than
>	Greater than
==	Equal
<=	Less than or equal
>=	Greater than or equal
!=	Not Equal

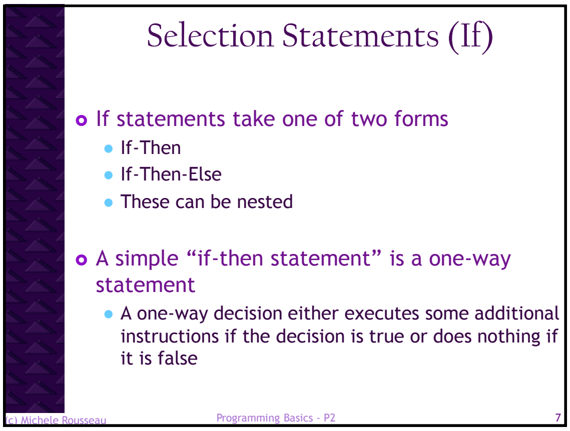
NOTE: this is not the same as =,
= ← is an assignment

We use Relational Operators to compare values in Selection Statements
→These will return a True or False value.

Relational Operators – Examples

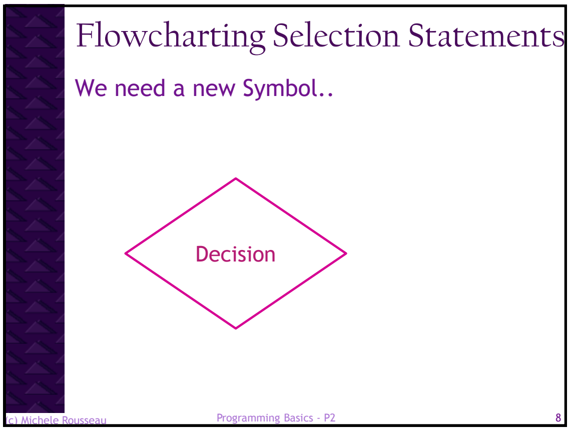
How would the computer evaluate these expressions?

4 < 7
3.1 > 3.1
11 == 8
41.1 <= 42
41.1 >= 42
12 != 12



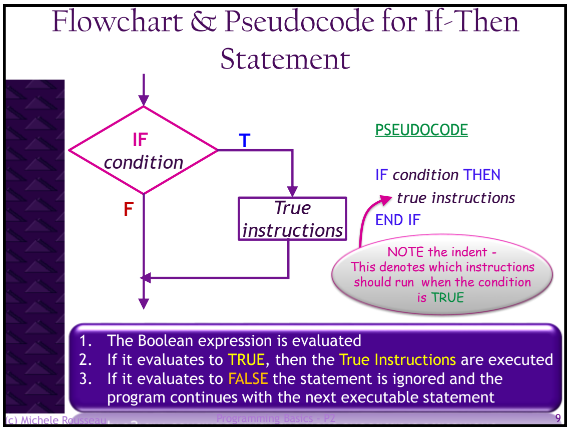
Selection Statements (If)

- If statements take one of two forms
  - If-Then
  - If-Then-Else
  - These can be nested
- A simple “if-then statement” is a one-way statement
  - A one-way decision either executes some additional instructions if the decision is true or does nothing if it is false



Flowcharting Selection Statements

We need a new Symbol..



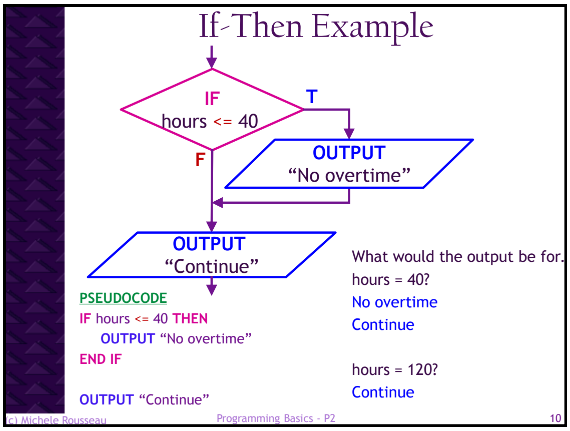
Flowchart & Pseudocode for If-Then Statement

PSEUDOCODE

```
IF condition THEN
    true instructions
END IF
```

NOTE the indent - This denotes which instructions should run when the condition is TRUE

1. The Boolean expression is evaluated
2. If it evaluates to TRUE, then the True Instructions are executed
3. If it evaluates to FALSE the statement is ignored and the program continues with the next executable statement



If-Then Example

PSEUDOCODE

```
IF hours <= 40 THEN
    OUTPUT “No overtime”
END IF

OUTPUT “Continue”
```

What would the output be for..

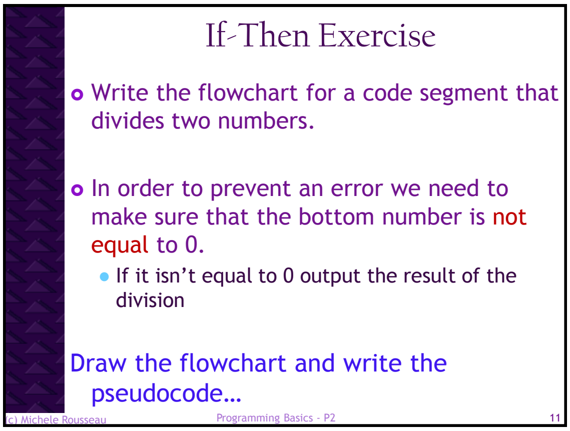
hours = 40?

No overtime

Continue

hours = 120?

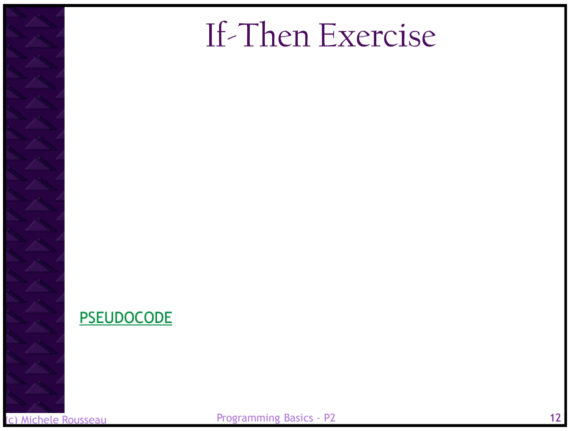
Continue



If-Then Exercise

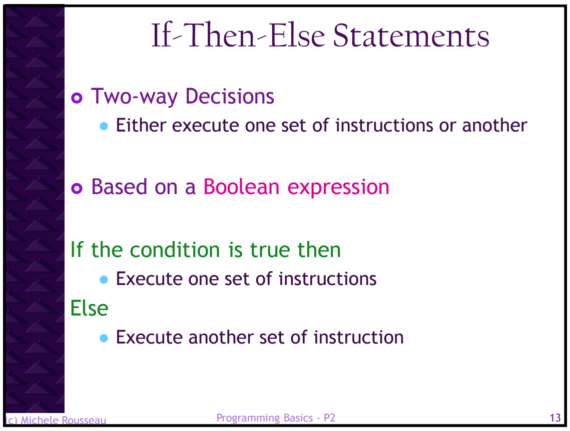
- Write the flowchart for a code segment that divides two numbers.
- In order to prevent an error we need to make sure that the bottom number is not equal to 0.
  - If it isn’t equal to 0 output the result of the division

Draw the flowchart and write the pseudocode…



If-Then Exercise

PSEUDOCODE



If-Then-Else Statements

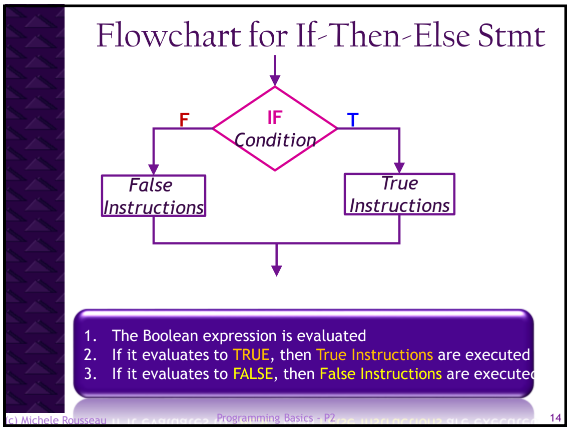
- Two-way Decisions
  - Either execute one set of instructions or another
- Based on a Boolean expression

If the condition is true then

- Execute one set of instructions

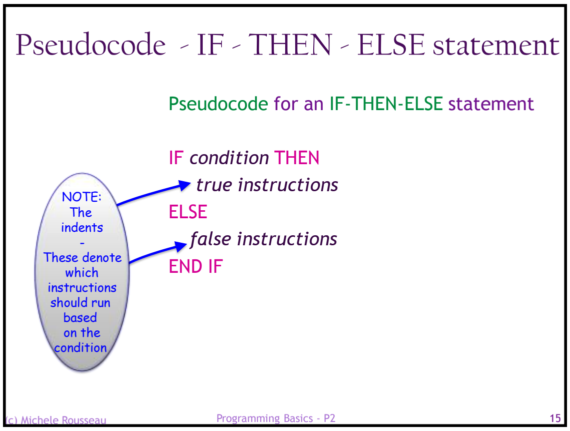
Else

- Execute another set of instruction



Flowchart for If-Then-Else Stmt

1. The Boolean expression is evaluated
2. If it evaluates to TRUE, then True Instructions are executed
3. If it evaluates to FALSE, then False Instructions are execute

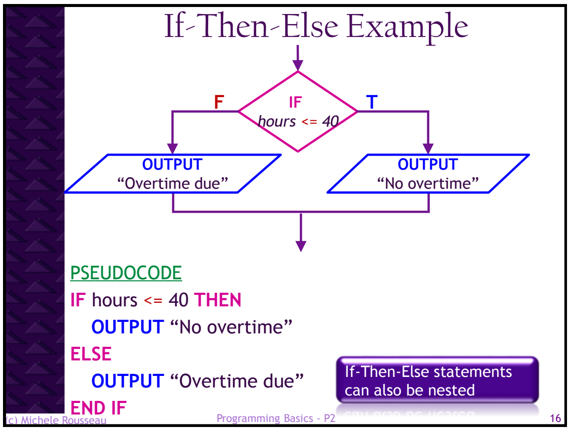


Pseudocode - IF - THEN - ELSE statement

Pseudocode for an IF-THEN-ELSE statement

```
IF condition THEN
    true instructions
ELSE
    false instructions
END IF
```

NOTE: The indents - These denote which instructions should run based on the condition

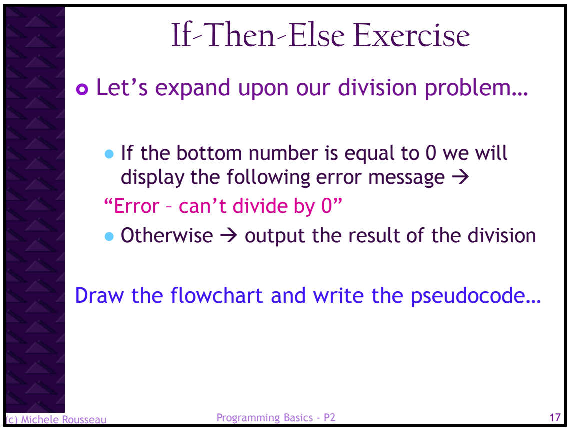


If-Then-Else Example

PSEUDOCODE

```
IF hours <= 40 THEN
   OUTPUT "No overtime"
ELSE
   OUTPUT "Overtime due"
END IF
```

If-Then-Else statements can also be nested

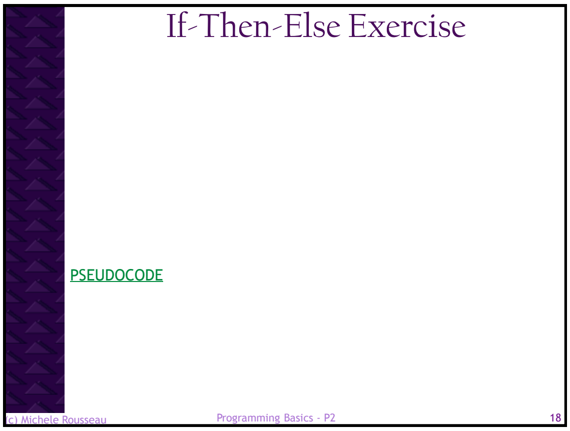


If-Then-Else Exercise

- Let's expand upon our division problem…
  - If the bottom number is equal to 0 we will display the following error message →

    "Error - can't divide by 0"
  - Otherwise → output the result of the division

Draw the flowchart and write the pseudocode…



If-Then-Else Exercise

PSEUDOCODE

Selection Exercise

Problem Statement:

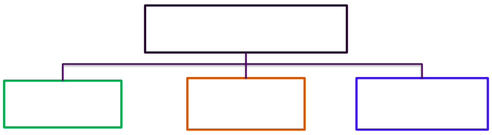
- Farmer Pete is trying to determine which animals to store in the larger pen. He needs to determine if he has more sheep or more pigs. His program should state which animal is most populous.
- Design the algorithm using a HIPO chart followed by pseudocode and a flowchart.

- What is the INPUT?
  - Count of sheep
  - Count of pigs
- What is the OUTPUT?
- What PROCESSING do we need to do?

Image: Courtesy of © Christine Jopling

Draw the HIPO chart

HIPO Chart



Remember we should refine to the level of 1 instruction

The HIPO chart should show the structure of the code→ note the way decisions are handled

Write the PSEUDOCODE

Draw the FLOWCHART

What variables do we need?

VARIABLE LIST

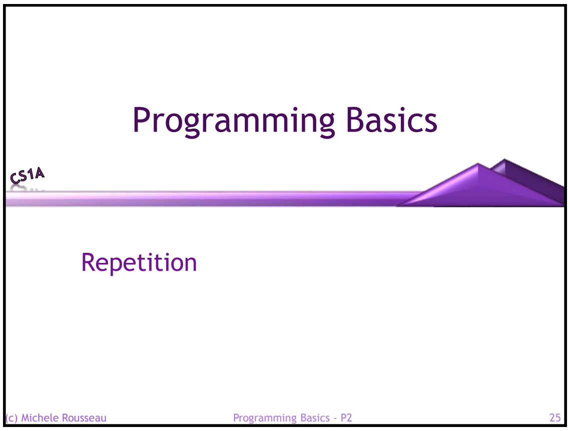
INPUT

OUTPUT

PROCESSING

Nesting If-Then-Else Statements

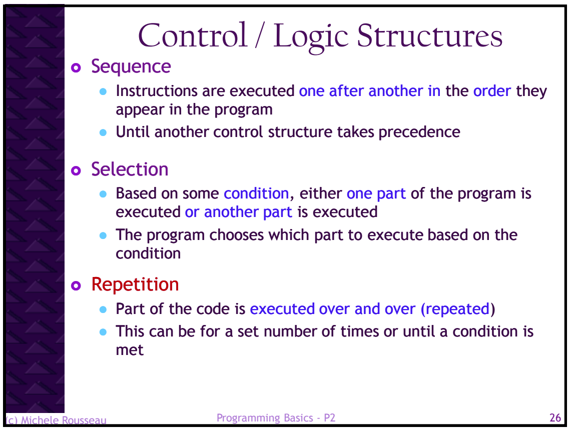
PSEUDOCODE



Programming Basics

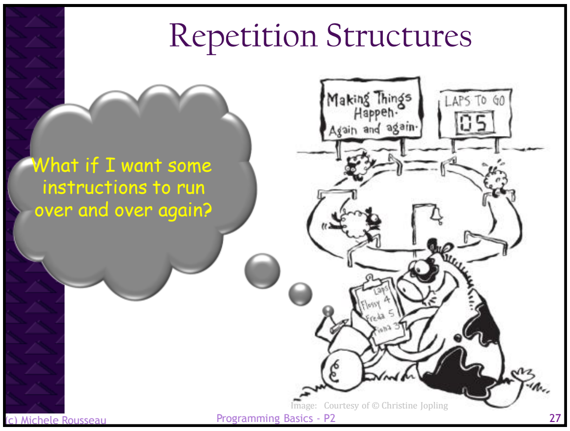
CS1A

Repetition



Control / Logic Structures

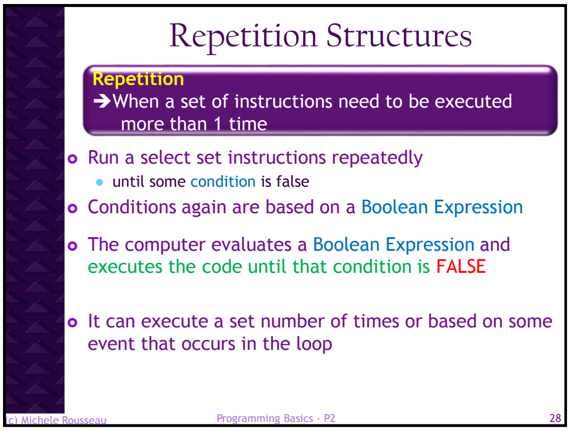
- Sequence
  - Instructions are executed one after another in the order they appear in the program
  - Until another control structure takes precedence
- Selection
  - Based on some condition, either one part of the program is executed or another part is executed
  - The program chooses which part to execute based on the condition
- Repetition
  - Part of the code is executed over and over (repeated)
  - This can be for a set number of times or until a condition is met



Repetition Structures

What if I want some instructions to run over and over again?

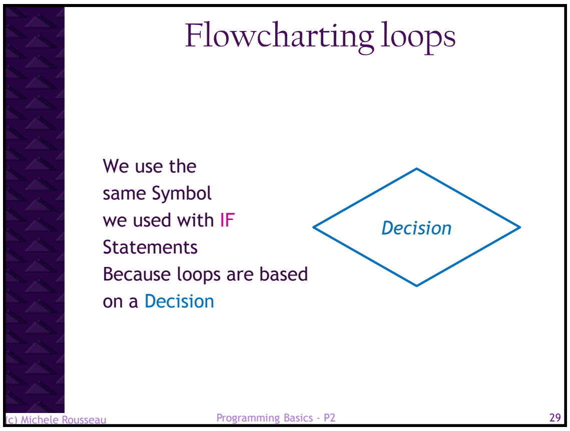
Image: Courtesy of © Christine Jopling



Repetition Structures

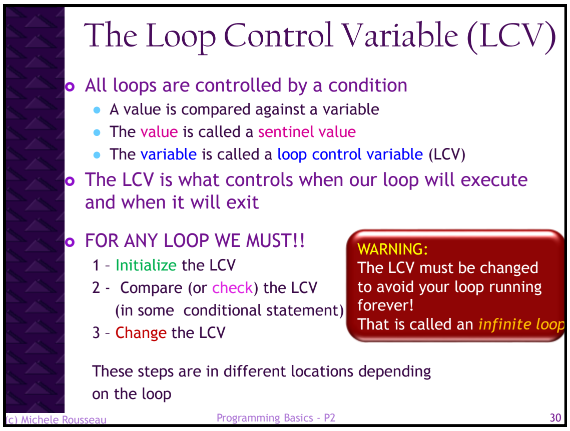
Repetition
➔When a set of instructions need to be executed more than 1 time

- Run a select set instructions repeatedly
  - until some condition is false
- Conditions again are based on a Boolean Expression
- The computer evaluates a Boolean Expression and executes the code until that condition is FALSE
- It can execute a set number of times or based on some event that occurs in the loop



Flowcharting loops

We use the same Symbol we used with IF Statements
Because loops are based on a Decision



The Loop Control Variable (LCV)

- All loops are controlled by a condition
  - A value is compared against a variable
  - The value is called a sentinel value
  - The variable is called a loop control variable (LCV)
- The LCV is what controls when our loop will execute and when it will exit
- FOR ANY LOOP WE MUST!!
  1 - Initialize the LCV
  2 - Compare (or check) the LCV
  (in some conditional statement)
  3 - Change the LCV

These steps are in different locations depending on the loop

WARNING:
The LCV must be changed to avoid your loop running forever!
That is called an *infinite loop*

3 Basic Repetition Structures

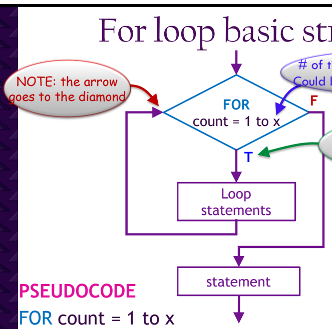
- For Loop
  - Part of a program is executed a given number of times.
- While Loop
  - Part of a program is executed while some condition is true: *While some condition is true execute these instructions*
- Do While Loop — We'll get to this later
  - Part of a program is executed at least one time and then repeats until some condition is false.

For right now we will focus on the For Loop

Initlialize, Check, Change the LCV

- Initialize occurs when the for loop is first entered
  - It initializes the variable to the first value specified
- Check occurs after the initialize then after each update / change
- Change occurs each successive time the decision box is entered
  - The variable is updated by 1

For loop basic structure



# of times you want to repeat
Could be a variable or a literal

Always loop on the True condition

For the FOR loop the LCV is:
→Initialized
→Checked
→and Changed

All within the FOR STATEMENT

PSEUDOCODE

```
FOR count = 1 to x
    <loop statements indented>
END FOR
<additional statements not in the loop>
```

EXAMPLE: FOR LOOP

- For loop →
  - repeats statements a pre-determined number of times

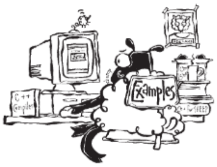


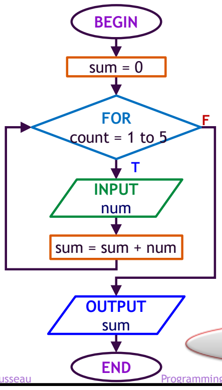
Image: Courtesy of © Christine Jopling

Example

let's say we want to input and sum 5 numbers

Let's write the flowchart

For loop Example



- Example:
  - Input and sum 5 numbers
  - Output the result

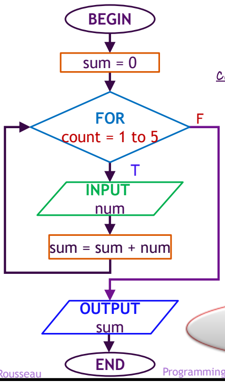
What is the INPUT?
num

What is the OUTPUT?
sum

What PROCESSING do we need to do?
How do we sum the nums?
sum = sum + num
sum is an accumulator

Let's do a Desk Check!
Use 5, 8, 22, 3, 1 as input

For loop Example – Desk Check

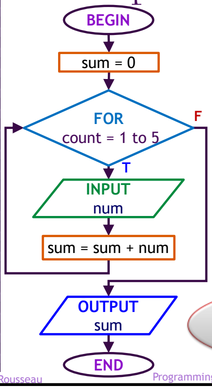


Input: 5, 8, 22, 3, 1
Expected Output:

count	num	sum	OUTPUT

Now you try it with inputs
4, 8, 11, 5, 2

For loop Example – Desk Check



Input: 4, 8, 11, 5, 2

Expected Output:

count num sum OUTPUT

Modify the flowchart so that it also calculates and outputs the average - Validate with a desk check

For Loop Exercise

For Loop Exercise - Pseudocode

3 Basic Repetition Structures

- For Loop
  - Part of a program is executed a given number of times.
- While Loop
  - Part of a program is executed while some condition is true: *While some condition is true execute these instructions*
- Do While Loop

  We'll get to this later
  - Part of a program is executed at least one time and then repeats until some condition is false.

For right now we will focus on the While Loop

While Loop

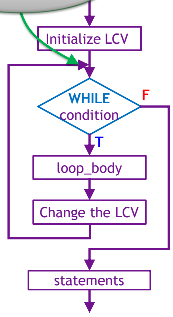
- What if we don't know how many times we need to run our loop?
- The code segment will run WHILE some condition is true
- The condition is tested at the top of the loop → making it a pre-test loop
  - if the condition evaluates to TRUE
    - the loop is entered
  - if the condition evaluates to FALSE
    - the loop is bypassed
- Event-controlled loop - a loop that terminates based on a condition and a *sentinel value* - *this loop executes an unspecified number of times*

LCV in While loops

- With a while loop we must
  - First determine which variable to use as the LCV
  - What the sentinel value should be
    - the value that is compared with the LCV
- The LCV needs to be initialized before entering the loop
  - The ***while statement*** *checks the LCV*
    - *Compares it to a sentinel value*
  - The LCV should be changed at the end of the loop
- We use the while loop
  - → the LCV is modified dynamically within the loop
- The LCV needs to be initialized before entering the loop

Flowchart for While Loop

Note: on the flow chart the arrow goes above the loop



1. First it evaluates the condition.

2. → If the condition is TRUE it executes the loop_body and returns to step 1

→If the condition is FALSE it exits the loop and does not execute the loop_body, it moves on to the statement immediately following the loop

The loop_body will continue to be executed as long as the condition is true

Pseudocode for the While Loop

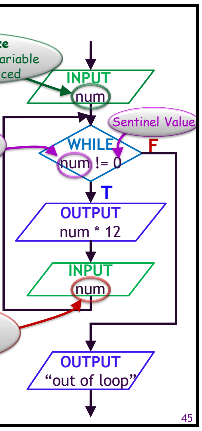
```
<initialize up here somewhere>
WHILE <condition>
    <loop statements indented>
    <change LCV in here somewhere>
END WHILE

<rest of pseudocode>
```

While loop Example

#1 - Initialize
The Loop Control Variable
Must be initialized

#2 - CHECK
The LCV is compared

Sentinel Value



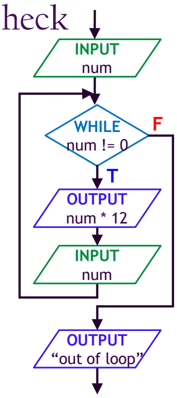
What does this do?

WARNING:
You must change the LCV so that you don’t have an infinite loop!

#3 - CHANGE
the LCV in the loop

While Loop - Desk Check

Input: 1, 2, 3, 0



When would we use them?

- When you need a counter
  - Count an unknown # of inputs
- When you need an accumulator →Running totals
  - Sum an unknown # of inputs
- When the user controls how many times the loop should execute

Example

Output the average of an unknown number of ages given as input.

Exercise: While loop

Program Description

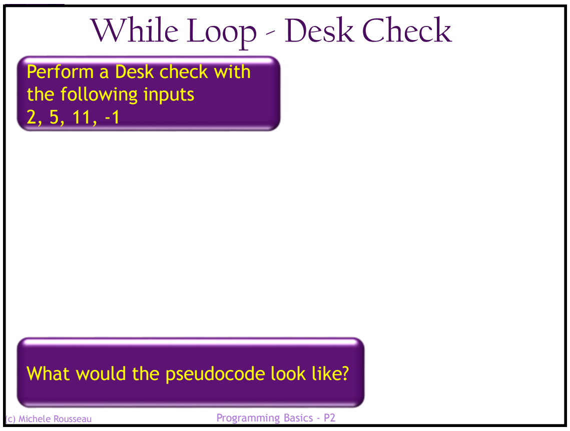
Write a program that will allow a user to input a series of ages and output the average of all ages.

What is the INPUT?

What is the OUTPUT?

What PROCESSING needs to occur?

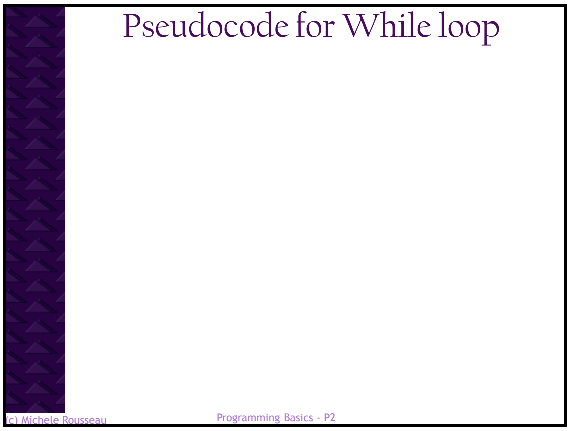
Let’s draw the flowchart.



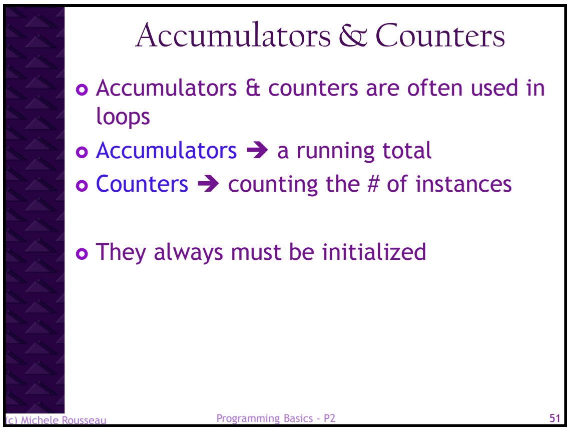
While Loop - Desk Check

Perform a Desk check with the following inputs 2, 5, 11, -1

What would the pseudocode look like?

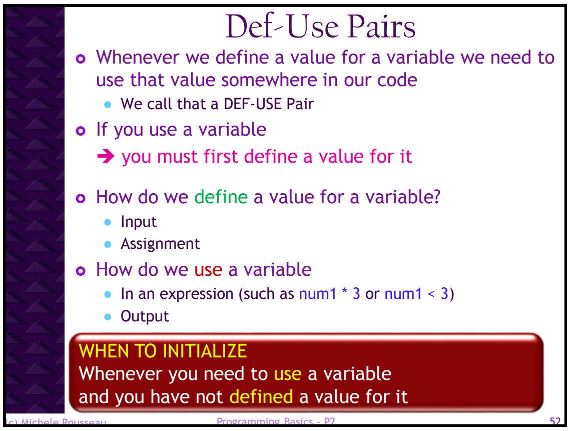


Pseudocode for While loop



Accumulators & Counters

- Accumulators & counters are often used in loops
- Accumulators ➔ a running total
- Counters ➔ counting the # of instances
- They always must be initialized

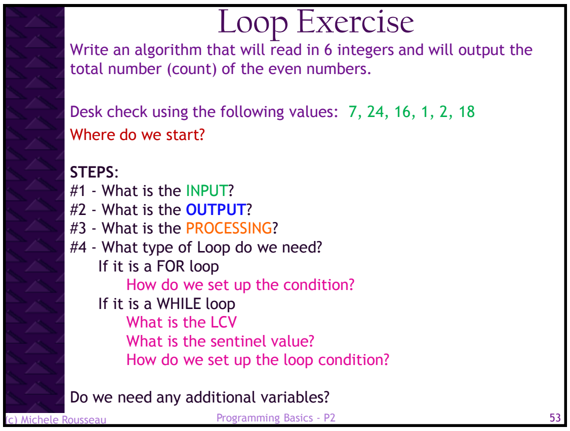


Def-Use Pairs

- Whenever we define a value for a variable we need to use that value somewhere in our code
  - We call that a DEF-USE Pair
- If you use a variable

  ➔ you must first define a value for it
- How do we define a value for a variable?
  - Input
  - Assignment
- How do we use a variable
  - In an expression (such as num1 * 3 or num1 < 3)
  - Output

WHEN TO INITIALIZE
Whenever you need to use a variable
and you have not defined a value for it



Loop Exercise

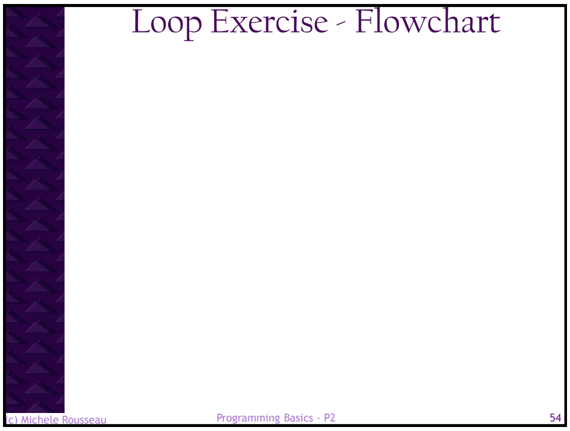
Write an algorithm that will read in 6 integers and will output the total number (count) of the even numbers.

Desk check using the following values: 7, 24, 16, 1, 2, 18

Where do we start?

STEPS:
#1 - What is the INPUT?
#2 - What is the OUTPUT?
#3 - What is the PROCESSING?
#4 - What type of Loop do we need?
If it is a FOR loop
How do we set up the condition?
If it is a WHILE loop
What is the LCV
What is the sentinel value?
How do we set up the loop condition?

Do we need any additional variables?



Loop Exercise - Flowchart

Loop Exercise - Pseudocode

Loop Exercise - Desk Check

Loop Exercise

- Write an algorithm that will accept an unknown number of positive integers (including 0) and output the total number of even integers.

STEPS:
#1 - What is the INPUT?
#2 - What is the **OUTPUT**?
#3 - What is the PROCESSING?
#4 - What type of Loop do we need?
    If it is a FOR loop
        How do we set up the condition?
    If it is a WHILE loop
        What is the LCV
        What is the sentinel value?
        How do we set up the loop condition?
Do we need any additional variables?

Loop Exercise

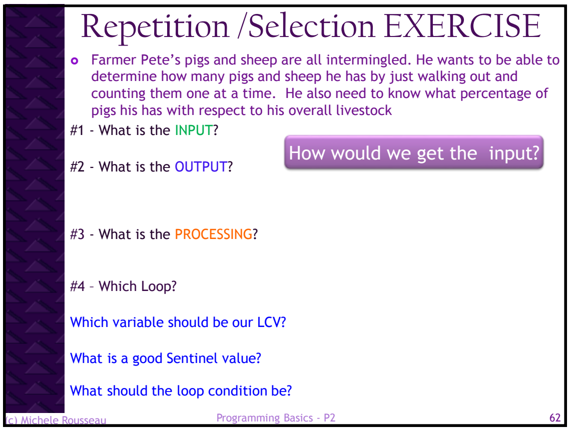
PSEUDOCODE

Repetition /Selection EXERCISE

- Farmer Pete's pigs and sheep are all intermingled. He wants to be able to determine how many pigs and sheep he has by just walking out and counting them one at a time. He also need to know what percentage of pigs his has with respect to his overall livestock.
- Since Farmer Pete doesn't know how many animals he has to count up front → which loop should we use?
- How can he enter the data?

Image: Courtesy of © Christine Jopling

NOTE:
We need to use a selection statement within the while loop?
➔ Which should we use?



Repetition /Selection EXERCISE

- Farmer Pete's pigs and sheep are all intermingled. He wants to be able to determine how many pigs and sheep he has by just walking out and counting them one at a time. He also need to know what percentage of pigs his has with respect to his overall livestock

#1 - What is the INPUT?

How would we get the input?

#2 - What is the OUTPUT?

#3 - What is the PROCESSING?

#4 – Which Loop?

Which variable should be our LCV?

What is a good Sentinel value?

What should the loop condition be?

(c) Michele Rousseau

Programming Basics - P2

62