BIL101E: Introduction to Computers and Information systems
Lecture 8

8.1 Algorithms
8.2 Pseudocode
8.3 Control Structures
8.4 Decision Making: Equality and Relational Operators
8.5 The if Selection Structure
8.6 The if/else Selection Structure
8.7 Nested control structures

3.1 Introduction

- Before writing a program:
  - Have a thorough understanding of the problem
  - Carefully plan an approach for solving it
- While writing a program:
  - Know what “building blocks” are available
  - Use good programming principles

3.2 Algorithms

- Computing problems
  - All can be solved by executing a series of actions in a specific order
- Algorithm: procedure in terms of
  - Actions to be executed
  - The order in which these actions are to be executed
- Program control
  - Specify order in which statements are to executed

3.3 Pseudocode

- Pseudocode
  - Artificial, informal language that helps us develop algorithms
  - Similar to everyday English
  - Not actually executed on computers
  - Helps us “think out” a program before writing it
    - Easy to convert into a corresponding C program
    - Consists only of executable statements

3.4 Control Structures

- Sequential execution
  - Statements executed one after the other in the order written
- Transfer of control
  - When the next statement executed is not the next one in sequence
- Bohm and Jacopini
  - All programs written in terms of 3 control structures
    - Sequence structures: Programs executed sequentially by default
    - Selection structures: C has three types: **`if`**, **`if`**/**`else`**
    - Repetition structures: C has three types: **`while`**, **`do`**/**`while`** and **`for`**

3.4 Control Structures

- Flowchart
  - Graphical representation of an algorithm
  - Drawn using certain special-purpose symbols connected by arrows called flowlines
  - Rectangle symbol (action symbol):
    - Indicates any type of action
  - Oval symbol:
    - Indicates the beginning or end of a program or a section of code
- Single-entry/single-exit control structures
  - Connect exit point of one control structure to entry point of the next (control-structure stacking)
  - Makes programs easy to build

3.5 Decision Making: Equality and Relational Operators

- Executable statements
  - Perform actions (calculations, input/output of data)
  - Perform decisions
    - May want to print **`"pass"`** or **`"fail"`** given the value of a test grade
- **`if`** control structure
  - If a condition is **`true`**, then the body of the **`if`** statement executed
    - **`0`** is **`false`**, non-zero is **`true`**
  - Control always resumes after the **`if`** structure

3.5 Decision Making: Equality and Relational Operators

Standard algebraic equality operator or relational operator	C equality or relational operator	Example of C condition	Meaning of C condition
Equality Operators			
=	==	x == y	x is equal to y
not =	!=	x != y	x is not equal to y
Relational Operators			
>	>	x > y	x is greater than y
<	<	x < y	x is less than y
>=	>=	x >= y	x is greater than or equal to y
<=	<=	x <= y	x is less than or equal to y

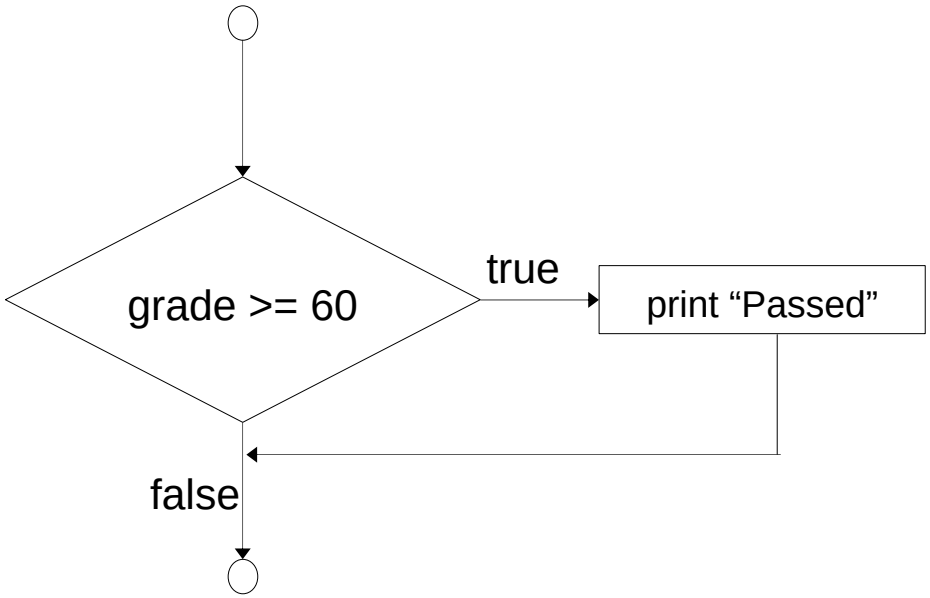
3.6 The `if` Selection Structure

- Selection structure:
  - Used to choose among alternative courses of action
  - Pseudocode:

    If student's grade is greater than or equal to 60
    Print "Passed"
- If condition **`true`**
  - Print statement executed and program goes on to next statement
  - If **`false`**, print statement is ignored and the program goes onto the next statement

3.6 The `if` Selection Structure

- **`if`** structure is a single-entry/single-exit structure



A decision can be made on any expression.

zero - **`false`**

nonzero - **`true`**

Example:

`3 - 4` is **`true`**

3.7 The `if/else` Selection Structure

- **`if`**
  - Only performs an action if the condition is **`true`**
- **`if/else`**
  - Specifies an action to be performed both when the condition is **`true`** and when it is **`false`**
- Psuedocode:

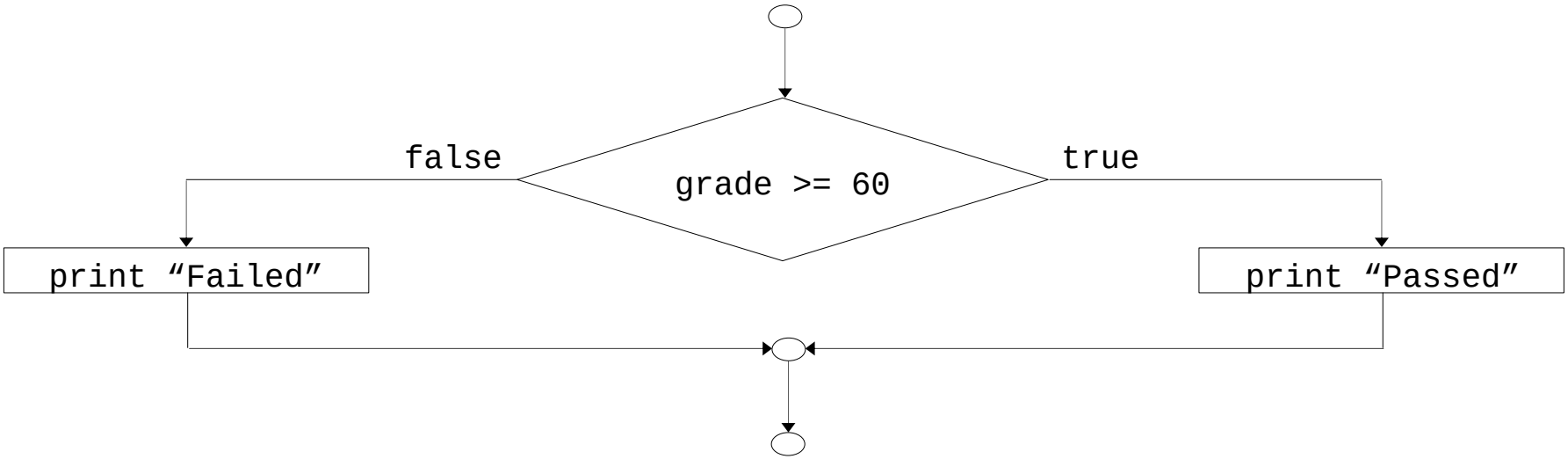
  If student's grade is greater than or equal to 60
  Print "Passed"

  else
  Print "Failed"

  - Note spacing/indentation conventions

3.7 The `if/else` Selection Structure

- Flow chart of the **`if`**/**`else`** selection structure



- Nested **`if`**/**`else`** structures
  - Test for multiple cases by placing **`if`**/**`else`** selection structures inside **`if`**/**`else`** selection structures
  - Once condition is met, rest of statements skipped

3.7 The `if/else` Selection Structure

– Pseudocode for a nested **`if`**/**`else`** structure

```
If student's grade is greater than or equal to 90
  Print "A"
else
  If student's grade is greater than or equal to 80
     Print "B"
  else
     If student's grade is greater than or equal to 70
        Print "C"
     else
        If student's grade is greater than or equal to 60
           Print "D"
        else
           Print "F"
```