CONFIDENTIAL

C Programming Introduction

Week 6:Branches statement

Lecturers : Cao Tuan Dung

Dept of Software Engineering

Hanoi University of Technology

For HEDSPI Project

Topic of this week

- Branches
  - Class Lecture Review
    - If selection structure.
    - Switch selection structure.
  - Programming Exercises

The if Selection Structure

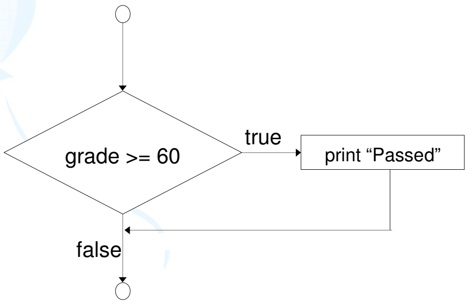
- Selection structure:
  - Used to choose among alternative courses of action
  - Pseudocode: *If student's grade is greater than or equal to 60*
    Print "Passed"
- If condition **true**
  - Print statement executed and program goes on to next statement.
  - If **false**, print statement is ignored and the program goes onto the next statement.
  - Indenting makes programs easier to read
    - C ignores whitespace characters.
- Pseudocode statement in C:

```
if ( grade >= 60 )
   printf( "Passed\n" );
```

  - C code corresponds closely to the pseudocode

The if Selection Structure (II)

- Diamond symbol (decision symbol) - indicates decision is to be made
  - Contains an expression that can be **true** or **false**
  - Test the condition, follow appropriate path
- **if** structure is a single-entry/single-exit structure.



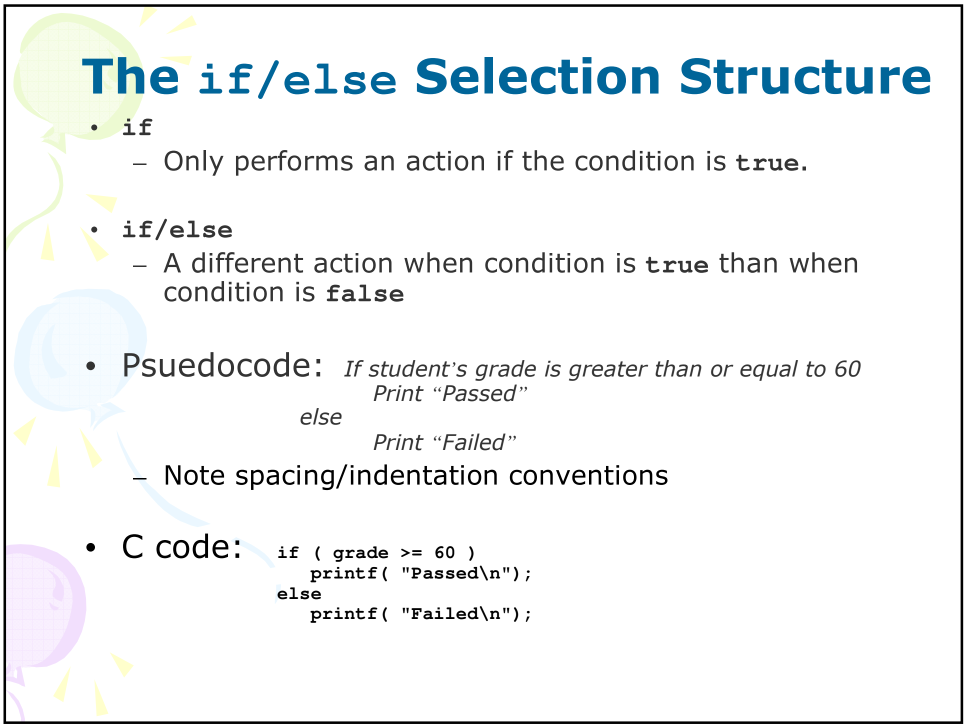
A decision can be made on any expression.

zero - **false**

nonzero - **true**

Example:

3 - 4 is **true**



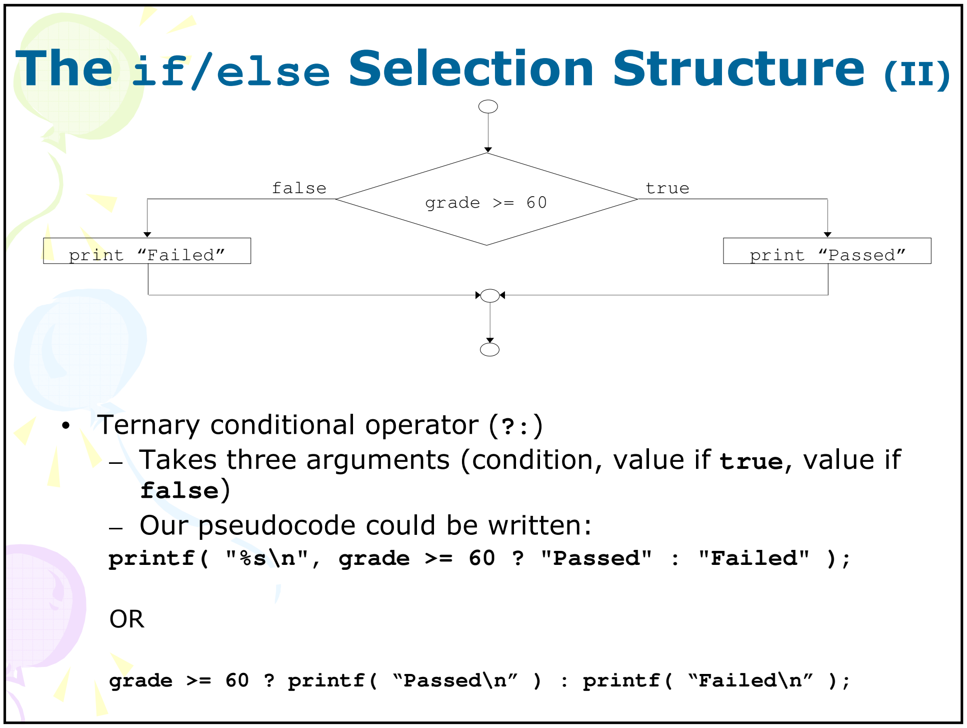
The `if/else` Selection Structure

- **`if`**
  - Only performs an action if the condition is **`true`**.
- **`if/else`**
  - A different action when condition is **`true`** than when condition is **`false`**
- Psuedocode: *If student's grade is greater than or equal to 60*

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

  - Note spacing/indentation conventions
- C code:

```
if ( grade >= 60 )
   printf( "Passed\n");
else
   printf( "Failed\n");
```



The `if/else` Selection Structure (II)

false

grade >= 60

true

print "Failed"

print "Passed"

- Ternary conditional operator (**`?:`**)
  - Takes three arguments (condition, value if **`true`**, value if **`false`**)
  - Our pseudocode could be written:

```
printf( "%s\n", grade >= 60 ? "Passed" : "Failed" );
```

OR

```
grade >= 60 ? printf( "Passed\n" ) : printf( "Failed\n" );
```

The `if/else` Selection Structure (III)

- Nested **`if/else`** structures
  - Test for multiple cases by placing **`if`**/**`else`** selection structures inside **`if`**/**`else`** selection structures

```
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”
```

  - Once condition is met, rest of statements skipped
  - Deep indentation usually not used in practice

The `if/else` Selection Structure (IV)

- Compound statement:
  - Set of statements within a pair of braces
  - Example:

```
if ( grade >= 60 )
   printf( "Passed.\n" );
else {
   printf( "Failed.\n" );
   printf( "You must take this course again.\n" );
}
```

  - Without the braces,

```
printf( "You must take this course again.\n" );
```

would be automatically executed

- Block: compound statements with declarations

The `if/else` Selection Structure (V)

- Syntax errors
  - Caught by compiler

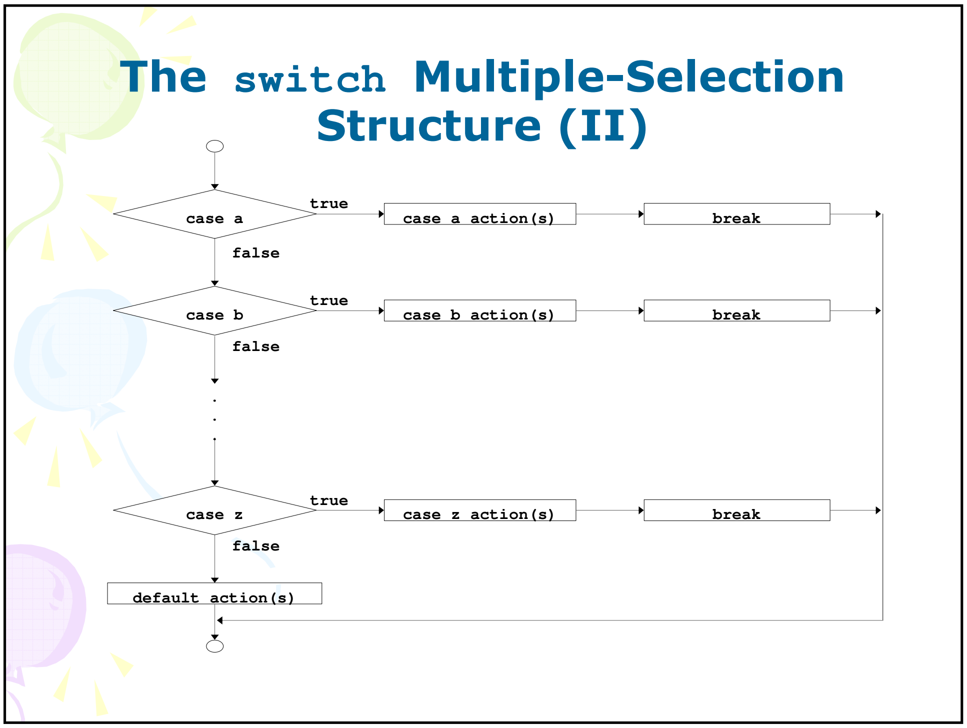
- Logic errors:
  - Have their effect at execution time
  - Non-fatal: program runs, but has incorrect output
  - Fatal: program exits prematurely

The `switch` Multiple-Selection Structure

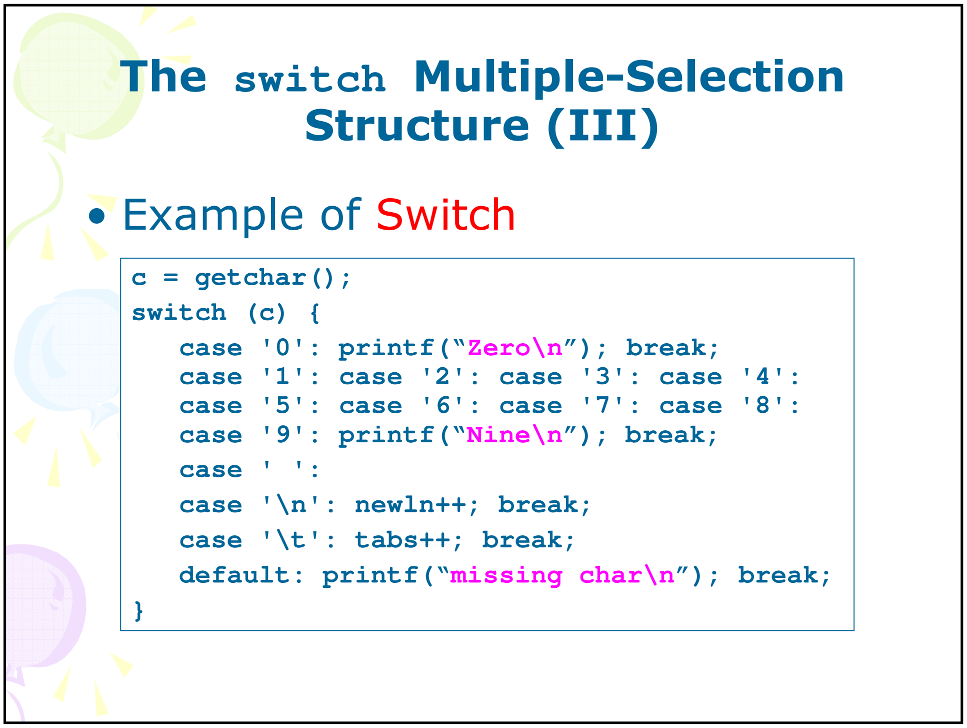
- **`switch`**
  - Useful when a variable or expression is tested for all the values it can assume and different actions are taken.
- Format
  - Series of **`case`** labels and an optional **`default`** case

```
switch ( value ){
    case '1':
        actions
    case '2':
        actions
    default:
        actions
    }
```

  - **`break;`** causes exit from structure



The switch Multiple-Selection Structure (II)



The switch Multiple-Selection Structure (III)

- Example of Switch

```
c = getchar();
switch (c) {
   case '0': printf(“Zero\n”); break;
   case '1': case '2': case '3': case '4':
   case '5': case '6': case '7': case '8':
   case '9': printf(“Nine\n”); break;
   case ' ':
   case '\n': newln++; break;
   case '\t': tabs++; break;
   default: printf(“missing char\n”); break;
}
```

Exercise 6.1

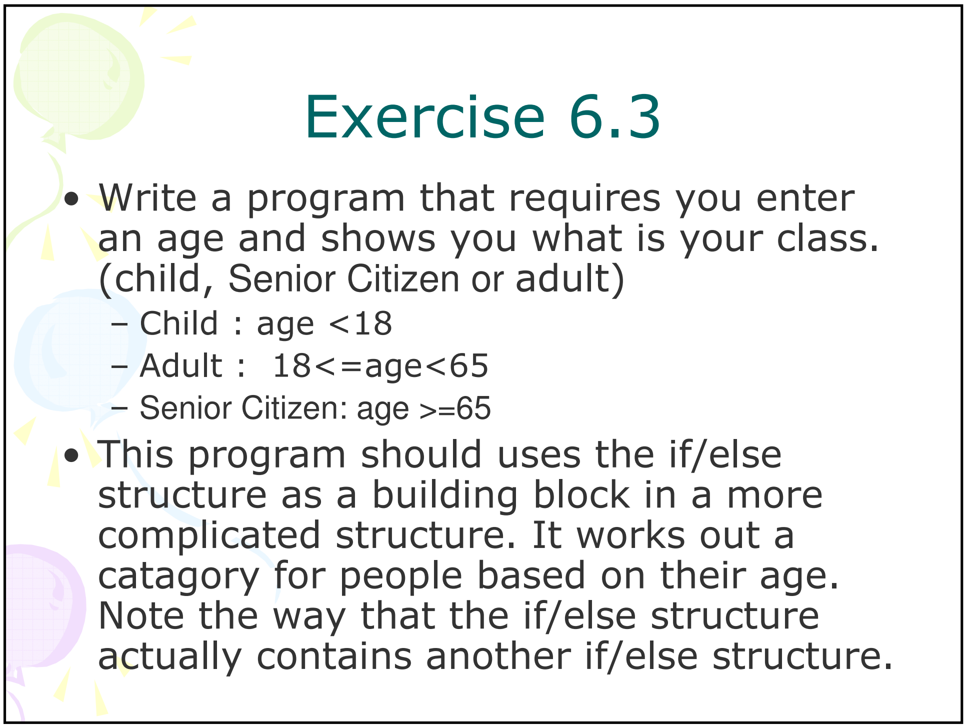
- Write a program that finds and displays the alphabetically first letter in a sequence (e.g. type IBK and it returns B).
- Note the use of a normal if/else and then an if on its own.

Exercise 6.2

- Write a program that transforms a compass heading to a compass bearing using this table:

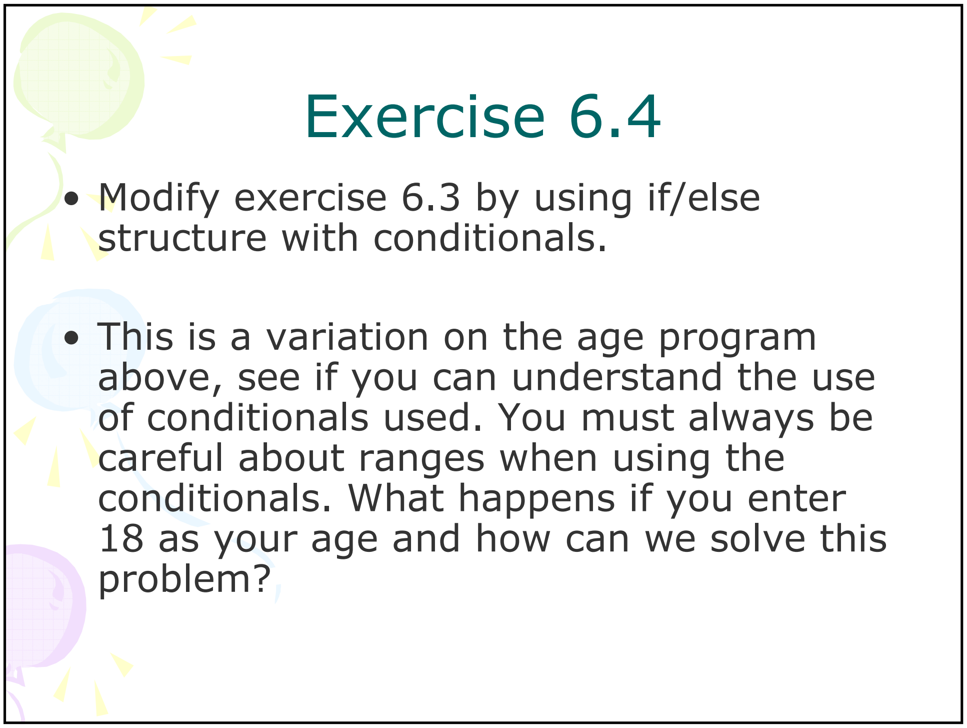
HEADING IN DEGREES	BEARING COMPUTATION
0 - 89.999...	north (heading) east
90 - 179.999...	south (180.0 - heading) east
180 - 269.999...	south (heading - 180.0) west
270 - 360	north (360.0 - heading) west

- The compass heading is entered by user. Use if/else..if structure.



Exercise 6.3

- Write a program that requires you enter an age and shows you what is your class. (child, Senior Citizen or adult)
  - – Child : age <18
  - – Adult : 18<=age<65
  - – Senior Citizen: age >=65
- This program should uses the if/else structure as a building block in a more complicated structure. It works out a catagory for people based on their age. Note the way that the if/else structure actually contains another if/else structure.

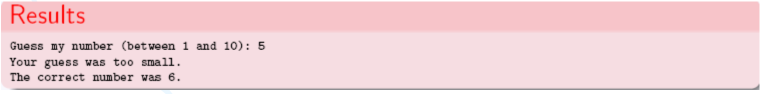


Exercise 6.4

- Modify exercise 6.3 by using if/else structure with conditionals.
- This is a variation on the age program above, see if you can understand the use of conditionals used. You must always be careful about ranges when using the conditionals. What happens if you enter 18 as your age and how can we solve this problem?

Exercise 6.5

- Write a program to play “High/Low". The program “picks" a number. The human player tries to guess it. The program indicates if the guess is too high, too low, or correct. Then it stops.
- Sample outputs:



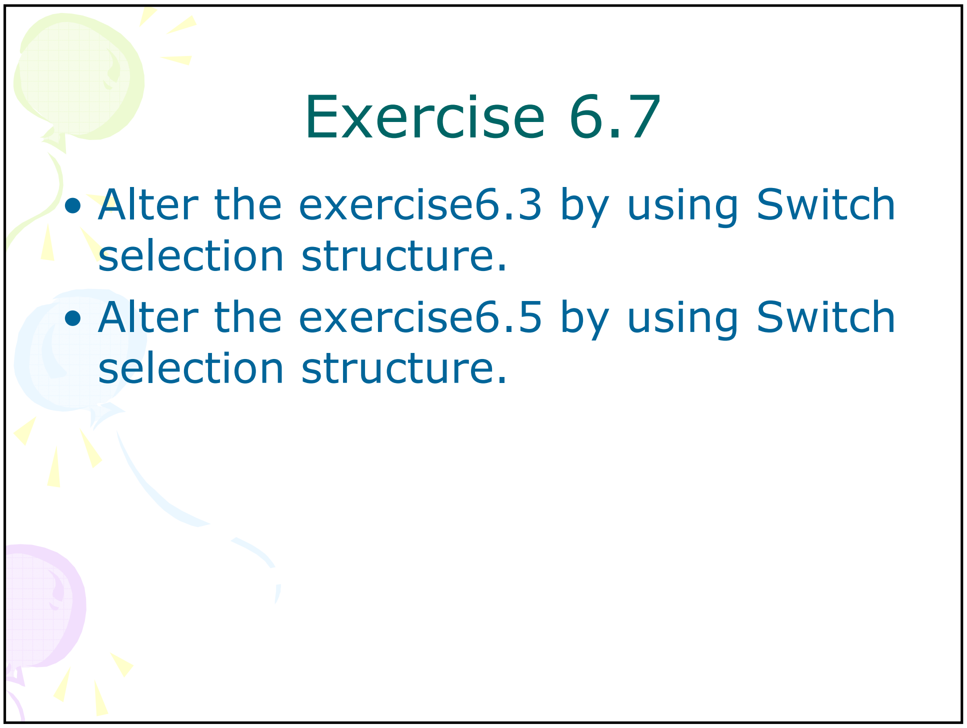
Results

```
Guess my number (between 1 and 10): 5
Your guess was too small.
The correct number was 6.
```

- Use rand() function to pick a random number.

Exercise 6.6

- Write a program that reads in three integers. and then determine which one is the smallest, and display it.
- If the values are a, b, and c, there are four cases:
  - a is smallest if a < b and a < c
  - b is smallest if b < a and b < c
  - c is smallest if c < a and c < b
  - No smallest When?



Exercise 6.7

- Alter the exercise6.3 by using Switch selection structure.
- Alter the exercise6.5 by using Switch selection structure.