UNCLASSIFIED//FOUO

Pseudocode, Logic and Design

1

UNCLASSIFIED

- **DECISION STRUCTURES AND BOOLEAN LOGIC**

UNCLASSIFIED 2

2

Learning Objectives

4.1 Introduction to Decision Structures
4.2 Dual Alternative Decision Structures
4.3 Comparing Strings
4.4 Nested Decision Structures
4.5 The Case Structure
4.6 Logical Operators
4.7 Boolean Variables

3

3

UNCLASSIFIED

4.1 Introduction to Decision Structures (1 of 2)

A decision structure (or selection structure) allows a program to perform actions only under certain conditions

Different types of decisions include

- If, also called single alternative
- If then else, also called dual alternative
- Case structure for multiple alternative decisions

UNCLASSIFIED 4

4

4.1 Introduction to Decision Structures (2 of 3)

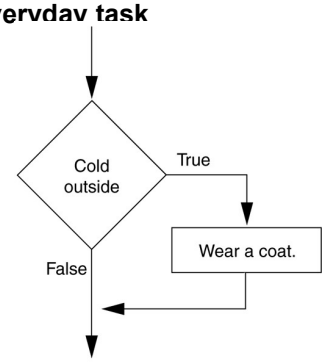
The if statement

- **An action only occurs if the decision is true**

```
If condition Then
    Statement
    Statement
End If
```

- **A diamond symbol is used in flowcharts**

Figure 4-1 A simple decision structure for an everyday task



5

5

4.1 Introduction to Decision Structures (3 of 3)

Relational Operators

- Determines whether a specific relationship exists between two values
- Used within the condition, a Boolean expression

x > y x<y x >=y x<y x <=y x==y x!=y

Table 4-1 Relational operators

Operator	Meaning
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
==	Equal to
!=	Not Equal to

6

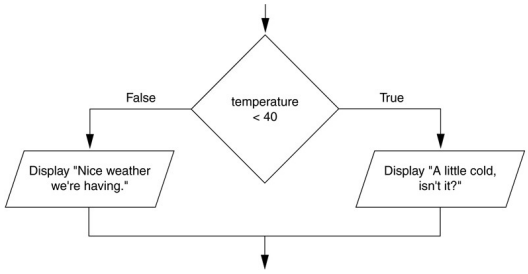
6

4.2 Dual Alternative Decision Structures (1 of 2)

If then else statement

- Executes one group of statements if it's Boolean expression is true, or another group if its Boolean expression is false

Figure 4-8 A dual alternative decision structure



7

7

4.2 Dual Alternative Decision Structures (2 of 2)

If condition Then
statement
statement
Else
statement
statement
End if

```
If temperature < 40 Then
   Display "A little cold"
   Display "Get a coat!"
Else
    Display "Nice weather"
     Display "And sunny!"
End if
```

 8

8

UNCLASSIFIED

4.3 Comparing Strings *(1 of 2)*

Most languages allow you to compare strings

Program 4-3

```
 1 // A variable to hold a password.
 2 Declare String password
 3
 4 // Prompt the user to enter the password.
 5 Display "Enter the password."
 6 Input password
 7
 8 // Determine whether the correct password
 9 // was entered.
10 If password == "prospero" Then
11     Display "Password accepted."
12 Else
13     Display "Sorry, that is not the correct password."
14 End If
```

Program Output (with Input Shown in Bold)

```
Enter the password.
ferdinand [Enter]
Sorry, that is not the correct password.
```

Program Output (with Input Shown in Bold)

```
Enter the password.
prospero [Enter]
Password accepted.
```

UNCLASSIFIED 9

9

4.3 Comparing Strings *(2 of 2)*

Other String Concerns

- **String and strings can be compared**

  name1 == name 2

- **String and string literals can be compared**

  Month != “October”

- **String comparisons are generally case sensitive**
- **You can also determine whether one string is greater than or less than another string (allows for sorting strings)**

10

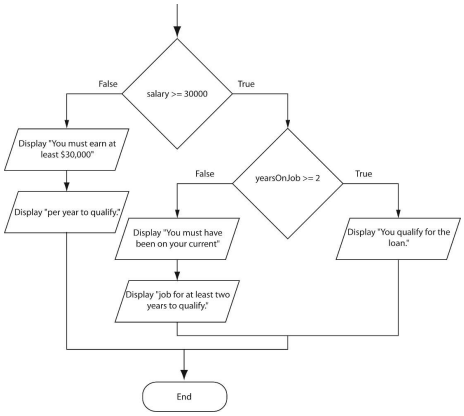
10

4.4 Nested Decision Structures (1 of 3)

Decisions are nested in order to test more than one condition

If then if else example

Figure 4-15 A nested decision structure



11

11

4.4 Nested Decision Structures (2 of 3)

The if then else if statement can make nested logic simpler to write

```
If score < 60 Then
        Display “Grade is F.”
Else If score < 70 Then
        Display “Grade is D.”
Else If score < 80 Then
        Display “Grade is C.”
Else If score < 90 Then
        Display “Grade is B.”
Else
        Display “Grade is A.”
End If
```

 12

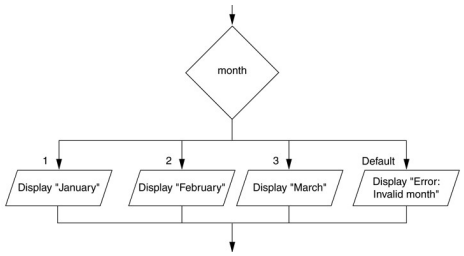
12

4.5 The Case Structure

The case structure lets the value of a variable or an expression determine which path of execution the program will take

- Can be used as an alternative to nested decisions

Figure 4-18 A case structure



13

13

4.6 Logical Operators *(1 of 4)*

Logical Operators are used between conditions to create complex Boolean expressions

- **AND – Both conditions must be true**
- **OR – Either condition must be true**
- **NOT – Reverses the truth of an expression**

14

4.6 Logical Operators (2 of 4)

AND example

```
If temperature < 20 AND minutes > 12 Then
    Display "The temperature is in the danger zone."
End If
```

OR example

```
If temperature < 20 OR temperature > 100 Then
    Display "The temperature is in the danger zone."
End If
```

NOT example

```
If NOT (temperature > 100) Then
    Display "This is below the maximum temperature."
End If
```

15

15

4.6 Logical Operators (3 of 4)

Short-Circuit Evaluation: Supported by many languages for increased performance

- **AND operator: If the expression on the left side of the AND operator is false, the expression on the right side will not be checked.**
- **OR operator: If the expression on the left side of the OR operator is true, the expression on the right side will not be checked.**

 16

16

4.6 Logical Operators (4 of 4)

Range Checking

- Often used for range checking
  - When checking for a number **inside** a range, use AND

```
If x >=20 AND x <=40 Then
        Display "The value is in the acceptable range."
End If
```

  - When checking for a number **outside** a range, use OR

```
If x < 20 OR x >40 Then
        Display "The value is outside the acceptable range."
End If
```

17

17

UNCLASSIFIED

4.7 Boolean Variables

A variable of the Boolean data type can hold one or two values: true or false

```
Declare Boolean isLunchTime
If time >=12 then
    Set isLunchTime = True
Else
    Set isLunchTime = False
End If
```

UNCLASSIFIED 18

18