R 53

IF Statements & Boolean Operators - Reading

Booleans store one of two values, **True** or **False**. We can combine these values using **Boolean Operators**.

Boolean Operator	Explanation	Example	Result
AND	Both operands (inputs) need to be True for the result to be True. Some languages will use && to mean AND	True AND True True AND False False AND True False AND False	True False False False
OR	If either or both of the operands are true then the result will be True. Some languages will use || to mean OR	True AND True True AND False False AND True False AND False	True True True False
NOT	The result will be the opposite of the operand given. Some languages will use ! to mean NOT	NOT True NOT False	False True
>	The left operand is **greater than** the right operand	5 > 3 5 > 5 5 > 10	True False False
>=	The left operand is **greater than or equal to** the right operand	5 >= 3 5 >= 5 5 > 10	True True False
<	The left operand is **less than** the right operand	5 < 3 5 < 5 5 < 10	False False True
<=	The left operand is **less than or equal to** the right operand	5 <= 3 5 <= 5 5 <= 10	False True True
=	The left operand is **equal to** the right operand. Some languages use ==	5 = 3 5 = 5 5 = 10	False True False
<>	The two operands are **not equal to** each other. Some languages use !=	5 <> 3 5 <> 5 5 <> 10	True False True

When we use Boolean operators we usually use them with variables and IF statements. An **IF statement** allows us to **branch** to two different sections of code depending on the outcome of a condition. The condition will always **evaluate** to True or False.

IF statement syntax:

```
IF condition THEN
    Run the code between
    then and else.
ELSE
    Run this code
END IF
```

Example:

```
IF 8 < 3 THEN
    OUTPUT "8 is less than 3"
ELSE
    OUTPUT "8 is not less than 3"
END IF
```

```
password = "fdu64"
IF password == "fdu64" THEN
    OUTPUT "Logged in"
ELSE
    OUTPUT "Not logged in"
END IF
```

Q 53

IF Statements & Boolean Operators - Questions

1. Connect the operators on the left with their meaning on the right.

<	Less than or equal to
<=	Greater than or equal to
>=	Greater than
=	Less than
>	Equal to

[5]

2. What will the following Boolean expressions evaluate to? The first is given as an example.

Boolean expression	Evaluates to
`False OR True`	**True**
`False OR False`	
`True AND True`	
`23 > 12`	
`"Help" = "Help"`	

[4]

3. What will the output from the code below be? _______________ [1]

```
password = "fdu64"
IF password = "fdu64" THEN
    OUTPUT "Logged in"
ELSE
    OUTPUT "Not logged in"
END IF
```

___
10

4. What will the following Boolean expressions evaluate to?

Boolean expression	Evaluates to
`17 <= 18`	
`15 <> 15`	
`-3 < 2`	
`"Help" <> "help"`	
`(5<10) AND ("abc"="abc")`	

[5]

5. Look at the code on the right and answer the questions.

```
firstNum = 15
secondNum = 25
total = firstNum + secondNum
average = total / 2
IF firstNum > average THEN
    OUTPUT "firstNum is greater"
ELSE
    OUTPUT "secondNum is greater"
END IF
```

a) What is the value of average at the end of the program? ______ [1]

b) What will be printed on the screen?

______________________________ [1]

6. Look at the code on the right and answer the questions.

```
gameOver = False
IF gameOver = True THEN
    OUTPUT "Game over"
ELSE
    OUTPUT "Still playing"
END IF
```

a) What is the value of gameOver? _______ [1]

b) What will be printed on the screen? ________________ [1]

c) The condition in the line of code: `IF gameOver = True THEN` Could be written more simply. Rewrite the line of code below.

____ [1]

___
10

R 54

Nested IF & CASE-SELECT statements - Reading

IF statements, like many structures in programming, can be placed inside one another. This is called a **nested** structure.

The code on the right shows how a **nested IF statement** can be used to tell the user whether they have entered an incorrect username or an incorrect password.

```
IF username = "smithp" THEN
    IF password = "awer" THEN
        OUTPUT "Logged in"
    ELSE
        OUTPUT "Incorrect password"
ELSE
    OUTPUT "Incorrect username"
END IF
```

Within IF statements it is possible for the condition to contain full **Boolean expressions** or use brackets. This can often save having to write lots of nested IF statements. For example:

```
gameOver = False
score = 2500
IF gameOver AND score >= 2000 THEN
   OUTPUT "That's a fantastic score"
ELSE
   OUTPUT "Not too good"
END IF
```

If we wanted to give a different message to the user for every day of the week, we could do an IF statement for each day. This causes a lot of typing and isn't that clear to read.

```
INPUT dayOfWeek
IF dayOfWeek = "Monday" THEN
    OUTPUT "Hate Mondays"
ELSE
    IF dayOfWeek = "Tuesday" THEN
        OUTPUT "Great, Monday's over"
    ELSE
        IF dayOfWeek = "Wednesday" THEN
...
```

There is another structure that does branching called **CASE-SELECT**. An example is shown below that will give seven different outputs depending on which day of the week is entered.

```
INPUT dayOfWeek
SELECT CASE dayOfWeek OF
    "Monday":     OUTPUT "Hate Mondays"
    "Tuesday":    OUTPUT "Great, Monday's over"
    "Wednesday":  OUTPUT "Middle of the week"
    "Thursday":   OUTPUT "Almost Friday"
    "Friday":     OUTPUT "Great, Friday"
OTHERWISE         OUTPUT "Lie in"
END SELECT
```

In other languages these may be called **SWITCH-CASE**. The name is different but the way they work is the same.

Q 54

Nested IF & CASE-SELECT Statements - Questions

1. When an IF statement is placed inside another IF statement it is known as what? Fill in one circle.

○ Embedded IF statement ○ Nested IF statement [1]

2. Look at the code on the right and answer the questions about it below.

```
playerName = "Jim"
playerScore = 23
IF playerName = "Jim"
    IF playerScore > 50
        OUTPUT "Great score"
    ELSE
        OUTPUT "Keep trying"
    END IF
ELSE
    OUTPUT "You’re not Jim!"
END IF
```

a) The first IF statement checks which condition? Fill in one circle.

○ playerScore > 50 ○ playerName = “Jim” ○ playerScore = 23 [1]

b) The nested IF statement checks which condition? Fill in one circle.

○ playerScore > 50 ○ playerName = “Jim” ○ playerScore = 23 [1]

c) What will the output be from the program? _______________ [1]

d) If playerScore was changed to the following scores, what would the output be from the program?

playerScore	Output
49	
50	
51	

[3]

e) If the first line of code were: playerName = “jim”
What would the output be from the program?

___________________________________ [1]

3. If a great score were greater than or equal to 50, what would the condition be in line 4? _________________________________ [1]

4. If a great score were greater than or equal to 50 and less than 100, what would the condition be? _________________________________ [1]

___
10

5. Look at the following code on the right and answer the questions about it below.

```
OUTPUT "What input device moves a mouse
pointer?"
INPUT answer
SELECT CASE answer OF
    "mouse":          OUTPUT "Correct"
    "keyboard":       OUTPUT "That’s for letters"
OTHERWISE             OUTPUT "Incorrect"
END SELECT
```

a) For each of the following inputs for the *answer* variable, write what the output would be from the program.

answer	Output
mouse	
Keyboard	
graphics tablet	

[3]

b) If you wanted to include “graphics tablet” as a “Correct” choice, what would the extra line of code need to be?

___________________________________ [1]

6. Look at the code on the right. For each of the inputs below what will the output(s) from the program be?

```
INPUT a
INPUT b
IF a > b AND
   (a < 20 OR b >= 13) THEN
   OUTPUT "A"
ELSE
   OUTPUT "B"
   IF b = a THEN
        OUTPUT "C"
   END IF
END IF
```

a	b	Output(s)
5	3	
-3	-5	
-5	-3	
5	5	
20	12	
22	13	

[6]

___
10