```
2 > 5
'Output': False
2 >= 5
'Output': False
2 != 5
'Output': True
2 == 5
'Output': False
```

The logical operators include Boolean NOT (not), Boolean AND (and), and Boolean OR (or). We can also carry out identity and membership tests using

- is, is not (identity)
- in, not in (membership)

```
a = [1, 2, 3]
2 in a
'Output': True
2 not in a
'Output': False
2 is a
'Output': False
2 is not a
'Output': True
```

The print() Statement

The print() statement is a simple way to show the output of data values to the console. Variables can be concatenated using the comma. Space is implicitly added after the comma.

```
a = 'I'
b = 'Love'
c = 'You'
print(a, b, c)
'Output': I Love You
```

Using the Formatter

Formatters add a placeholder for inputting a data value into a string output using the curly brace {}. The format method from the str class is invoked to receive the value as a parameter. The number of parameters in the format method should match the number of placeholders in the string representation. Other format specifiers can be added with the placeholder curly brackets.

```
print("{} {} {}".format(a, b, c))
'Output': I Love You
# re-ordering the output
print("{2} {1} {0}".format(a, b, c))
'Output': You Love I
```

Control Structures

Programs need to make decisions which result in executing a particular set of instructions or a specific block of code repeatedly. With control structures, we would have the ability to write programs that can make logical decisions and execute an instruction set until a terminating condition occurs.

The if/elif (else-if) Statements

The if/elif (else-if) statement executes a set of instructions if the tested condition evaluates to true. The else statement specifies the code that should execute if none of the previous conditions evaluate to true. It can be visualized by the flowchart in Figure 9-2.