Small Basic

Controlling your code

What can we do in a computer program?

- Receive information (input)
- Output information (display text or graphics)
- Do arithmetic (+, -, *, / and a few others)
- Assign a value to a variable (store data – and find it again)
- Compare two pieces of information and select an action (‘selection’)
- Repeat actions until a condition is met (‘repetition’)

All the software that has been written ultimately depends on these 6 actions.

Quick Quiz - Review

For each line of code below, pick the action from the list above that best describes the statement. For example, the first line of code is an output statement. (For answers, see the end of these notes).

```smallbasic
TextWindow.WriteLine("Please enter your name")
myname=TextWindow.Read()
TextWindow.WriteLine("good morning " + myname + " What is your favourite number?")
numb= TextWindow.ReadNumber()
square = numb * numb
cube = square * numb
TextWindow.WriteLine("The square of " + numb + " is " + square)
TextWindow.WriteLine("The cube of " + numb + " is " + cube)
TextWindow.WriteLine("I'm so smart. Goodbye")
```

Selection (If ..then..else)

Most computer problems involve decisions – for example, you may check the data entered in a form and decide if meets certain criteria – if the data is valid we store it, otherwise we display an error message.

A selection statement evaluates a condition and then carries out one of two sets of actions. A condition is a comparison which results in either true or false.

An example is shown below:

```smallbasic
TextWindow.WriteLine("Enter the password")
pwd = TextWindow.Read()
If pwd="abracadbra" Then
   TextWindow.WriteLine("You can enter")
Else
   TextWindow.WriteLine("Go away")
EndIf
```
In this code, the user is asked to enter a password. The response is stored in a variable called `pwd` and one of two messages is displayed.

The **CONDITION** compares the contents of the variable `pwd` to a fixed string (“abracadabra”).

```plaintext
If pwd="abracadabra" Then
```

If the condition is TRUE, the first action is carried out, otherwise, the computer skips to the second action.

**CONDITIONS**

A condition MUST return TRUE or FALSE. Conditions use the comparison operators, listed below with some examples:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Greater than</td>
<td>a &gt; b</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
<td>Age &lt; 18</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
<td>counter &gt;= 10</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
<td>hours &lt;=40</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>Not equal to</td>
<td>password &lt;&gt; &quot;&quot; (this checks that the password is not ‘empty’)</td>
</tr>
<tr>
<td>=</td>
<td>Equal to</td>
<td>userPassword = pwd</td>
</tr>
</tbody>
</table>

Exactly the same kind of conditions are used to control a loop (i.e., repetition). For now, conditions will be used just with if...then statements.

**EXERCISES**

1. Write a program that asks the user the hour (24 hour clock). If the hour is after 5pm (17) then tell them to go home, otherwise ask them how their day at TAFE is going.

2. Write a program that asks the user to input two different numbers. Display a message if both numbers are the same.

3. The award wages for an employee require that they be paid time and a half for hours worked in excess of 40. Write a program that accepts the hours worked and the hourly pay rate as input then displays the gross wage.

4. Write a program that asks the user what the capital of Australia is? If the answer is incorrect display message and ask if they would like another try.

**COMPLEX CONDITIONS**

Conditions can be joined with ‘and’ or ‘or’ to allow for more complicated decisions, for example the code below allows someone to ride a roller coaster if they are over 12 or over a certain height:

```plaintext
If age>12 or height>1.5 Then
    TextWindow("Enjoy the roller coaster")
Else
    TextWindow("You can't ride")
EndIf
```
For more complex questions, statements can be ‘nested’, for example:

```csharp
If pwd="abracadbra" Then
    TextWindow.WriteLine("You can enter")
Else
    If pwd="" Then
        TextWindow.WriteLine("You have to enter something")
    Else
        TextWindow.WriteLine("Go away")
    EndIf
EndIf
```

Notice that the whole second ‘if’ statement starts and finishes inside the second half of the first ‘if’ statement.

5. Write a program that displays the message on a mobile speed camera. Use a variable to store the current speed limit (it may be 60, 100, 40, 110 or anything you decide when you test the program). Ask the user to input the car speed (this would be automatically entered from the radar but we’ll have to make do). If the car speed is under the limit display ‘good driver’. For speeds more than 10 km over, display ‘dangerous’ otherwise display ‘slow down’.

### Answers - Quick Quiz

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextWindow.WriteLine(&quot;Please enter your name&quot;)</td>
<td>Output information</td>
</tr>
<tr>
<td>myname=TextWindow.Read()</td>
<td>Receive information</td>
</tr>
<tr>
<td></td>
<td>Assign a value to a variable</td>
</tr>
<tr>
<td>TextWindow.WriteLine(&quot;good morning &quot; + myname + &quot; What is your favourite number?&quot;)</td>
<td>Output information</td>
</tr>
<tr>
<td>numb= TextWindow.ReadNumber()</td>
<td>Receive information</td>
</tr>
<tr>
<td></td>
<td>Assign a value to a variable</td>
</tr>
<tr>
<td>square = numb * numb</td>
<td>Do arithmetic</td>
</tr>
<tr>
<td></td>
<td>Assign a value to a variable</td>
</tr>
<tr>
<td>cube = square * numb</td>
<td>Do arithmetic</td>
</tr>
<tr>
<td></td>
<td>Assign a value to a variable</td>
</tr>
<tr>
<td>TextWindow.WriteLine(&quot;The square of &quot; + numb + &quot; is &quot; + square)</td>
<td>Output information</td>
</tr>
<tr>
<td>TextWindow.WriteLine(&quot;The cube of &quot; + numb + &quot; is &quot; + cube)</td>
<td>Output information</td>
</tr>
<tr>
<td>TextWindow.WriteLine(&quot;I'm so smart. Goodbye&quot;)</td>
<td>Output information</td>
</tr>
</tbody>
</table>

### SOLUTIONS

Sample code for solutions to these exercises is provided at the end of this topic.