

## Writing Pseudocode



When writing pseudocode you must follow a certain format. Below is the format you need to write your algorithm in.

### Variables

Variables are assigned using the = operator.

```
x=3  
name="Bob"
```

A variable is declared the first time a value is assigned. It assumes the data type of the value it is given.

Variables declared inside a function or procedure are local to that subroutine.

Variables in the main program can be made global with the keyword global.

```
global userid = 123
```

### Casting

Variables can be typecast using the int str and float functions.

```
str(3) returns "3"  
int ("3.14") returns 3  
float ("3.14") returns 3.14
```

### Outputting to Screen

```
print(string/variable)  
OUTPUT string/variable
```

### Example

```
print("hello")  
OUTPUT "Hello"
```

### Taking Input from User

```
variable=input(prompt to user)  
INPUT variable
```

### Example

```
name=input("Please enter your name")  
INPUT name
```

### Selection

Selection will be carried out with if/else

```
if entry=="a" then  
    print("You selected A")  
elseif entry=="b" then  
    print("You selected B")  
else  
    print("Unrecognised selection")  
endif
```

### switch/case

```
switch entry:  
    case "A":  
        print("You selected A")  
    case "B":  
        print("You selected B")  
    default:  
        print("Unrecognised  
        selection")  
endswitch
```

### Iteration – Count Controlled

```
for i=0 to 7  
    print("Hello")  
next i  
Will print hello 8 times (0-7 inclusive).
```

### Iteration – Condition Controlled

```
while answer!="computer"  
    answer=input("What is the  
    password?")  
endwhile  
  
do  
    answer=input("What is the  
    password?")  
until answer=="computer"
```

<p><b>String Handling</b></p> <p>To get the length of a string: stringname.length</p> <p>To get a substring: stringname.subString(startingPosition, numberOfCharacters)</p> <p><b>NB</b> The string will start with the 0th character.</p>	<p><b>Example</b></p> <pre>someText="Computer Science" print(someText.length) print(someText.substring(3,3))</pre> <p>Will display 16 put</p>
--	---

<p><b>Arrays</b></p> <p>Arrays will be 0 based and declared with the keyword array.</p> <pre>array names[5] names[0]="Ahmad" names[1]="Ben" names[2]="Catherine" names[3]="Dana" names[4]="Elijah" print(names[3])</pre> <p>Example of 2D array: Array board[8,8] board[0,0]="rook"</p>	<p><b>Reading to and Writing from Files</b></p> <p>To open a file to read from openRead is used and readLine to return a line of text from the file.</p> <p>The following program makes x the first line of sample.txt</p> <pre>myFile = openRead("sample.txt") x = myFile.readLine() myFile.close()</pre> <p>endOfFile() is used to determine the end of the file. The following program will print out the contents of sample.txt</p> <pre>myFile = openRead("sample.txt") while NOT myFile.endOfFile() print(myFile.readLine()) endwhile myFile.close()</pre> <p>To open a file to write to openWrite is used and writeLine to add a line of text to the file. In the program below hello world is made the contents of sample.txt (any previous contents are overwritten).</p> <pre>myFile = openWrite("sample.txt") myFile.writeLine("Hello World") myFile.close()</pre>
---	---

<p><b>Subroutines</b></p> <pre>function triple(number)     return number*3 endfunction</pre> <p><b>Called from main program</b></p> <pre>y=triple(7) procedure greeting(name)     print("hello"+name) endprocedure</pre> <p><b>Called from main program</b></p> <pre>greeting("Hamish")</pre>	<p><b>Unless stated values passed to subroutines can be assumed to be passed by value. If this is relevant to the question byVal and byRef will be used. In the case below x is passed by value and y is passed by reference.</b></p> <pre>procedure foobar(x:byVal, y:byRef)     ...     ... endprocedure</pre>
---	--

<b>Logical Operators</b>	
AND	while x<=5 AND flag==false both conditions must be met
OR	while x<=5 OR flag==false one condition must be met
NOT	while x>5 NOT x=3 one condition not another is met
<b>Comparison Operators</b>	
==	Equal to
!=	Not equal to
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
<b>Arithmetic Operators</b>	
+	Addition e.g. x=6+5 gives 11
-	Subtraction e.g. x=6-5 gives 1
*	Multiplication e.g. x=12*2 gives 24
/	Division e.g. x=12/2 gives 6
^	Exponentiation e.g. 3^4 gives 81