Pseudocode

Revision sheet

1 Syntax

Pseudocode doesn't use curly brackets, it uses indentation and end statements for multi-line code sections (for example `endif`).

1.1 Comments

Lines which are to be commented out begin with a double slash, `//`.

1.2 Variables

Variables are declared the first time they are assigned. Data types do not need to be declared. For example:
`x = 3` would assign 3 to variable called x.
`name = 'bob'` would assign ‘bob’ to variable called name.

1.2.1 Casting

Variables can be typecast using the functions shown below.
`str(3)` returns `'3'`.
`int('3')` returns `3`.
`float('3.14')` returns `3.14`.

1.3 Arrays

Arrays are zero based and are declared and assigned in the following way:

```
array names[5]
names[0]='Ben'
```

1.3.1 Two Dimensional Arrays

These are declared and assigned as followed.

```
array board[8,8]
board[0,0]='rook'
```

1.4 Input And Output

1.4.1 Input

This is done as follows

```
variableName = input('string to appear on screen')
```

1.4.2 Output

This is done as follows

```
print(string)
```

For example

```
print('hello world')
```

1.5 Iteration

1.5.1 Count Controlled

The example shows a for loop which will run 8 times (0 to 7 inclusive).

```
for i=0 to 7
    print('Hello world')
next i
```

1.5.2 Condition Controlled

The examples below both use the same situation, input validation. The first example uses a while loop.

```
while answer != 'computer'
    answer = input ('What is the password?')
endwhile
```

The second example uses a do while loop.

```
do
    answer = input('What is the password?')
until answer == 'computer'
```

1.6 Selection

In the examples in the following sections, the code is looking at the contents of the variable `entry` and outputting a string based off of its contents.

1.6.1 If

The example below uses `if`, `elseif` and `else`.

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

1.6.2 Switch

The examples below use a `switch` statement.

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

1.7 Operators

These are the same as they are in other programming languages.

1.8 String Handling

To get the length of a string, use the following syntax:
`stringname.length`
To get a substring, use the following syntax (nb. strings are zero-based - like arrays):
`stringname.subString(startingPosition, numberOfCharacters)`

1.9 Subroutines

1.9.1 Functions

```
function triple (number)
    return number * 3
endfunction
```

1.9.2 Procedure

```
procedure greeting(name)
    print('Hello' + name)
endprocedure
```

1.9.3 Calling From The Main Program

```
greeting('Hamish')
```

The code above would print `Hello Hamish` to the screen.

1.9.4 Parameters

By default, these are passed by value. If the question doesn't specify, assume it means pass by value.
`parName:byVal` - by value declaration.
`parName:byRef` - by reference declaration.

1.10 Files

The following program will print out the contents of `sample.txt`. The same code can be adapted to save the contents of the file to an array or string for example.

```
myFile = openRead('sample.txt')
while NOT myFile.endOfFile()
    print(myFile.readLine())
endwhile
myFile.close()
```

To write to a file, use the following syntax

```
myFile = openWrite('sample.txt')
myFile.writeLine('Hello World')
myFile.close()
```

1.11 Object Oriented Pseudocode

1.11.1 Class Declaration and Inheritance

The code below declares a class called `Pet` with an attribute `name` and a method called `new` - this is the classes constructor.

```
class Pet
    private name
    public procedure new(givenName)
        name = givenName
    endprocedure
endclass
```

Inheritance is denoted by the `inherits` keyword. Methods from the super class will be called with the keyword `super`. For example, `super.methodName(parameters)` The example below shows a subclass of `Pet` called `Dog` (notice the constructor for the superclass uses the `super.methodName` syntax).

```
class Dog inherits Pet
    private breed
    public procedure new(giveName, givenBreed)
        super.new(givenName)
        breed = givenBreed
    endprocedure
endclass
```

1.11.2 Methods and Attributes

Both methods and attributes can be assumed to be public unless otherwise stated. Where the access level is relevant to the question, it will always be explicit in the code (denoted by the keywords `public` and `private`). The example below shows a public procedure.

```
public procedure setAttempts(number)
    attempts=number
endprocedure
```

Methods will be called using the following syntax:

```
player.setAttempts(5)
```