CAB402 Programming Paradigms
Practical Exercises (Week 2)

Part A - Getting started with F# in Visual Studio

https://docs.microsoft.com/en-us/dotnet/fsharp/get-started/get-started-visual-studio

Part B

For each of the following examples, try to guess what the output will be before executing the snippets in the F# interactive window.

When viewing the results, examine both the values and their types.

Ask you tutor if you can’t understand what it is doing.

Exercise 1 (simple name binding)

```
let a = 42
a + 3
```

Exercise 2 (different value types)

```
let anInt = 10
let aFloat = 20.0
let aString = "I'm a string!"
```

Exercise 3 (simple functions)

```
let succ x = x + 1
succ 3
succ (succ 5)
```

Exercise 4 (partial function application)

```
let add x y = x + y
add 3 5
let a4 = add 4
a4 5
```

Exercise 5 (nested helper functions)

```
let quadruple x =
    let double x =
        x * 2
    double(double x)
quadruple 3
```

Exercise 6 (higher order functions)

```
let chrisTest test =
    test "Chris"
let isMe x =
    if x = "Chris" then
        "it is Chris!"
    else
        "it's someone else"
chrisTest isMe
```

Exercise 7 (lambda functions)

```
let add = (fun x y -> x + y)
add 2 2
```

Exercise 8 (inline lambda functions)

```
let twoTest test =
    test 2
twoTest (fun x -> x < 0)
```

Exercise 9 (printing messages)

```
printfn "hello world from Try F#!"
```

Exercise 10 (printing with parameters)

```
printfn "The answer is %d" 42
```

Part C – Indentation and the Offside rule

Which of the following examples follow the offside rule correctly?

Example 1

```
let area r =
    let pi = 3.14159
    2.0 * pi * r
area 45.0
```

Example 2

```
let area r =
    let pi = 3.14159
     2.0 * pi * r
area 45.0
```

Example 3

```
let area r =
    let pi = 3.14159
   2.0 * pi * r
area 45.0
```

Example 4

```
let area r =
    let pi = 3.14159
    2.0 * pi * r
pi
```

Example 5

```
let area
  r =
    let pi =
      3.14159
    2.0
     * pi * r
area 45.0
```

Part D – Writing your own functions

Be sure to thoroughly test your functions to ensure they work for a diverse range of input values.

Exercise 1

Write a function to calculate the absolute value of a number.

Exercise 2

Write a function to calculate the length of the hypotenuse of a right-angle triangle using Pythagoras theorem.

Exercise 3

Write a function to calculate if a year is a leap year.

Exercise 3

Write a function to calculate the number of days in a particular month.

Exercise 4 (Recursion)

Recursive functions are defined in F# as follows:

```
let rec fact x = if x < 1 then 1 else x * fact(x-1)

let rec fib n = if n < 2 then n else fib(n-1) + fib(n-2)
```

Write recursive function implementations for the following functions:

- http://en.wikipedia.org/wiki/Lucas_number
- http://en.wikipedia.org/wiki/Ackermann_function
- Is http://en.wikipedia.org/wiki/Prime_number ?
- http://en.wikipedia.org/wiki/Greatest_common_divisor
- http://en.wikipedia.org/wiki/Least_common_multiple