***Your submission** for this assignment **must be commented** and **must include both your name** and your **student number** as a comment at the top of **every source file you submit**. Each of your submitted files must use a **file name beginning 'comp3007_w23_#########_assignment_06'** (replacing the number signs with your own student number) and any **submissions that crash** (i.e., terminate with an error) on execution will automatically **receive a mark of 0**.*

Officially, the Due Date for this Assignment is:

***Friday, March 17th, 2023**, at **11:59pm EST**.*

Late Submissions are Accepted Without Penalty Until Sunday, March 19th, by 11:59pm EST.
Submissions received after that will not be accepted and will receive a mark of 0.

The objective of this assignment is allow you to practice with structural induction, for the purpose of proving the correctness of a program. With the following collection of function definitions…

```
foo :: [Int] -> Int
foo [] = 0
foo [x] = 1
foo (h:t) = 1 + (foo t)

bar :: [Int] -> [Int]
bar [] = []
bar (h:t) = (foo (h:t)) : bar t

qux :: [Int] -> (Int, Int)
qux [] = (0, 0)
qux x = (ham x, 1)

ham :: [Int] -> Int
ham (h:_) = h

xyz :: (Int, Int) -> Int
xyz (x, 1) = x * (x + 1)
xyz x = 0
```

…prove, using structural induction, that, for any list of integers `x` …

```
xyz (qux (bar x)) == (foo x) * (foo x + 1)
```

For this assignment:

- your submission must be a single pdf document that you created electronically (using a word processor like Microsoft Word or Google Docs), and please note that scans or digital photographs of handwritten submissions will not be accepted and will receive a mark of zero
- you must include your full name and student number at the top of your submission, and you must copy the source code included above and provide an "identifier" for every line of source code
- you must show *all* your work (i.e., one step per line) in proving both the base case and the inductive case, *separately* working the left-hand side and right-hand side of each expression, and you must *explicitly* state your inductive assumption when proving the inductive case
- you do not need to prove termination