COMPUTER SCIENCE 1JC3
ASSIGNMENT 1 - 4 OCT 2016

name	
student number	

seagull

Instructor: Christopher Anand

Explain to a TA in tutorial or the drop-in centre by: 18 Oct for 5/5 marks, 21 Oct for 4/5 marks, 25 Oct for 3/5, 28 Oct for 2/5, 1 Nov for 1/5. Make any significant mistakes, you have to come back and explain it again.

In this assignment, you will show your assignment to a TA in your tutorial or lab, where you will have to explain your solution and answer a question about the solution you got.

You will use the following data structure in this assignment.

```
type Expr = Const Float
          | Var String
          | Add  Expr Expr
          | Mult Expr Expr
          | Cos Expr
```

Draw a tree corresponding to the expression $3((x+2)(x+0)+(y+2)(y+2))+\cos(xy+2x+(1+0)y)$.

Write a ELM expression to create this tree, i.e. something like `Add (Var "x") (Const 2)`.

Show how the following function would transform the tree, and be prepared to explain (a) what the function does, (b) why you think this is correct and incorrect and how you could prove it, (c) how you would extend the function given new requirements by your TA, and (d) underline *conquering* circle *dividing* and box *combining*.

```
trans e = case e of
            (Add a (Const 0))   -> trans a
            (Add a b)           -> Add (trans a) (trans b)
            (Mult a b)          -> Mult (trans a) (trans b)
            (Mult a (Const 1))  -> trans a
            (Cos b)             -> Cos (trans b)
            otherwise           -> e
```

You will be asked a question about this function:

```
pretty : Expr -> String
pretty e = case e of
             (Const d)    -> toString d                                          -- eg: "3.0"
             (Var x)      -> x                                                   -- eg: "x"
             (Add e1 e2)  -> "(" ++ (pretty e1) ++ " + " ++ (pretty e2) ++ ")" -- eg: "(x + z)"
             (Cos e1)     -> "cos(" ++ (pretty e1) ++ ")"                        -- eg: "cos(x)"
             (Mult e1 e2) -> "(" ++ (pretty e1) ++ " * " ++ (pretty e2) ++ ")" -- eg: "((x+z)*y)"
```

You will get to answer a surprise question.

The End.