

--RESOLUCIÓN propuesta - Guía de ejercitación nro. 8 - apartados B y C  
--Autor: Nico Colacioppo

--comentarios

--Definición de una función:  
--cabecera  
--implementación

--Función 1) cuadrado de un número.  
--nombre\_funcion::TipoDatoEntrada -> TipoDatoSalida  
cuadrado :: Int -> Int  
--nombre\_funcion argumentos = implementación  
cuadrado x = x ^ 2

--Función 2) menor entre dos, parcializada  
menor::Int -> Int -> Int  
menor x y = if x < y then x else y

--Función 3) mayor entre dos, no parcializada  
mayor :: (Int, Int) -> Int  
mayor(x, y) = if x > y then x else y

--Función 4) determinar si es par o no, un número.  
es\_par :: Int -> Bool  
es\_par x = mod x 2 == 0

--Función 5) devuelve el incremento de 1 respecto a un nro. entero recibido  
(++) :: Int -> Int  
(++) x = x + 1

--Función 7) aplicando if then else anidados  
evaluar\_entero :: Int -> String  
evaluar\_entero x = if x > 0 then "mayor que cero" else (if x < 0 then "menor que cero" else "igual que cero")

--función 7) aplicando guardas  
evaluar\_entero\_v2 :: Int -> String  
evaluar\_entero\_v2 x | x > 0 = "mayor que cero"  
                  | x < 0 = "menor que cero"  
                  | x == 0 = "igual que cero"

--función 7) aplicando guardas, con otherwise  
evaluar\_entero\_v3 :: Int -> String  
evaluar\_entero\_v3 x | x > 0 = "mayor que cero"  
                  | x < 0 = "menor que cero"  
                  | otherwise = "igual que cero"

--función 8)  
evaluar\_polinomio::Int->String  
evaluar\_polinomio x | y > 0 = "mayor que cero"  
                  | y < 0 = "menor que cero"  
                  | otherwise = "igual que cero"  
                  where y = x^2 + 5\*x + 0

dia :: Int -> String  
dia x = case x of  
    0 -> "domingo"  
    1 -> "lunes"  
    2 -> "martes"  
    3 -> "miércoles"  
    4 -> "jueves"  
    5 -> "viernes"  
    6 -> "sabado"  
    otherwise -> "no especificado"

--hay\_nota\_menor\_a\_4 n1 n2 n3 = n1 < 4 || n2 < 4 || n3 < 4

--hay\_nota\_menor\_a\_7 n1 n2 n3 = n1 < 7 || n2 < 7 || n3 < 7

hay\_nota\_menor\_a n1 n2 n3 ref = n1 < ref || n2 < ref || n3 < ref

promedio n1 n2 n3 = (n1 + n2 + n3) / 3

estado\_academico n1 n2 n3 | hay\_nota\_menor\_a n1 n2 n3 4 = "Libre"  
                          | hay\_nota\_menor\_a n1 n2 n3 7 || p < 8 = "Regular"  
                          | otherwise = "Promoción"  
                          where p = promedio n1 n2 n3