

Revisión de ejercicios de Haskell: *changing the point of view*

Los siguientes archivo de Hugs.hs , ¿están correcto? ¿funcionan?

Y si funcionan, ¿qué es lo que hace cada uno? ¿que función está definiendo cada uno?

Por ejemplo, si escribimos dentro de Haskel, Main> pote 3 4 , ¿cuál esperamos que sea la respuesta?

Se espera que el estudiante pueda responder estas preguntas antes de probarlo efectivamente con la computadora. Pero está aprendiendo ensayo y error

```
1)  pote::Int->Int->Int
    pote a b | b==0 =1
             | a==1 =1
             | otherwise = a*pote a (b-1)
```

Si hay dudas, probar con pote 2 3, pote 2 4, pote 2 5 y quizás se puedan dar cuenta.....

```
2)  pote::Int->Int->Int
    pote a b
    | b==0 =1
    | a==1 =1
    | otherwise = a*pote a (b-1)
```

3) Ahora que ya conocemos un poco mas de esta función, ¿como la tendríamos que modificar si queremos calcular 11^{13} ? Ayuda: calcule 10^8 , 10^9 , 10^{10} .

```
4)  jota :: Int -> String
    jota a | mod 7 a == 0 = "es"
           | otherwise    = "no lo es"
```

¿Qué es lo que hace jota ?

¿que pequeña modificación hay que hacer aquí para que esta función jota sea mucho mas útil?

```
5)  ceibo :: Int -> (Int , Int)
    ceibo a | mod a 8 ==0 =(div a 8,0)
            | mod a 3 ==0 =(0,div a 3)
            | mod a 8 == 1 = (div a 8,3)
```

La función ceibo esta bastante bien..... aunque está aún sin terminar. Para algunos números funciona, pero para otros no. Probar con : ceibo 64, ceibo 65,

Esta función, cuando este terminada, será la solución a un ejercicio del repartido anterior. ¿Cuál?