Resolución parcial # 1

Inciso # 1: Demuestre que para todo natural de peano "n" se cumple la siguiente propiedad:

Succ 0 + n = Succ n

Propiedades de la suma:

1. n+0 = 0
2. 0+m = m
3. n + Succ a = Succ (n+a)
4. a + Succ b = Succ a + b

Podemos realizar lo siguiente:

= Succ 0 + n
= 0 + Succ n
= Succ (0+n)
= Succ n

Inciso # 2: Provea una definición inductiva para la propiedad "mayor que" (>) tal que:

$$a > b \begin{cases} Succ\ 0 & si\ a\ es\ mayor\ que\ b \\ 0 & de\ lo\ contrario \end{cases}$$

En otras palabras, la propiedad "mayor que" es equivalente a Succ 0 si el primer valor es mayor que el segundo o 0 de lo contrario. Puede utilizar el operador ">" en su definición de la misma manera que se utiliza "+" en la definición de suma.

Definiciones de la suma:

1. n+0 = 0
2. 0+m = m
3. n + Succ a = Succ (n+a)
4. a + Succ b = Succ a + b

Definición para mayor que
1. a = b + n (siendo “n” un número natural)
2. b = 0
3. a = 0 + n
4. a > 0 = Succ a
5. entonces a > b siempre que a = b + n
6. a > b siempre que b = 0 y a ≠ 0

Inciso # 3: Provea una definición de las propiedades "esPar" e "esImpar" tal que:

$$esImpar \begin{cases} Succ\ 0 & Si\ n\ es\ impar \\ 0 & de\ lo\ contrario \end{cases}$$

$$esPar\ n \begin{cases} Succ\ 0 & si\ n\ es\ un\ numero\ par \\ 0 & de\ lo\ contrario \end{cases}$$

Paso 1:

1. esPar n= Succ 0 + n (siendo n únicamente un número impar)
2. esImpar n = Succ 0 + n (Siendo n únicamente un número par)

Teniendo en cuenta que siempre que cualquier número es multiplicado por 2 tendremos como resultado un número par:

1. Succ (Succ 0) x n = n seria par
2. Succ (Succ 0) x n + Succ 0 = n seria impar (al sumar una unidad deja de ser par).

Siempre que "n" tenga el valor de un número par y además se le sume una unidad tendremos como resultado un número impar.
Si le sumamos el valor de 0 permanecerá como un número par.

Por lo tanto:

1. esPar n = Succ 0 + n (siendo n únicamente un número impar)
2. esImpar n = Succ 0 + n (Siendo n únicamente un número par)
3. Succ (Succ 0) x n = n seria par
4. Succ (Succ 0) x n + Succ 0= n seria impar (al sumar una unidad deja de ser par).

Serie #4: Utilice el lenguaje de programación Haskell para definir la propiedad "predecesor". Esta propiedad debe aceptar un numero de peano y producir el predecesor de este. En el caso de cero, utilizar cero como su predecesor.

```
{-# LANGUAGE NoImplicitPrelude #-}
module Main where

import Prelude (Show, undefined, appendFile, (-), Foldable (sum) )

data Natural = O | Succ Natural deriving Show

--De numeros naturales a numeros de peano
-- Anat 6 = Succ......
anat 0 = O
anat n = Succ (anat (n - 1))

--Suma
(+) :: Natural -> Natural -> Natural
O + m = m
n + O = n
n + (Succ a) = Succ (n + a)

--Predecesor
pred (Succ O) = O
pred (Succ a) = a
main = undefined
```