

PUNTUACIONES:

| | 1 | 2 | 3 | 4 | 5 | 6 | total |
|--|---|---|---|---|---|---|-------|
| | 2 | 1 | 2 | 1 | 1 | 3 | 10.0 |
| | | | | | | | |

APELLIDOS: _____

NOMBRE: _____

si } desevo que se publique mi calificación si fuera negativa
 no }

Días de asistencia a clase en este parcial: _____ de 12

Consideremos las siguientes declaraciones para representar y manipular los números naturales:

```

data N = O | S N deriving (Show, Eq)
_1 = S O; _2 = S _1; _3 = S _2
instance Num N where y + O   = y
                  y + S x = S (y + x)
pliega f z O    = z
pliega f z (S n) = f (pliega f z n)

```

1 Demuestra por inducción que se verifica $\forall n . n :: N \cdot _1 + n = S n$

2 Demostremos que el tipo de *pliega* es $pliega :: \dots$
 En efecto:

3 Define las funciones *doble* (calcula el doble de un natural) y la suma directamente a través de *pliega*:

| | |
|----------------------------------|------------------------------------|
| <i>doble</i> :: N → N | instance Num N where |
| <i>doble</i> = <i>pliega</i> ... | (+) y = <i>pliega</i> ... |

4 Completa las declaraciones del operador \leq :

```

instance Ord N where O ≤ y = True
                  S x ≤ S y = ...
                  ...

```

5 Define una función para calcular la lista de los menores del argumento:

| | |
|-----------------------------------|---------------------------|
| MAIN> <i>menores</i> _3 | <i>menores</i> :: N → [N] |
| [S(S O), S O, O] :: [N] | <i>menores</i> ... |

6 Sea ahora la función de iteración mas general:

```

itera :: (N → a → a) → a → N → a
itera g z O    = z
itera g z (S n) = g n (itera g z n)

```

Define las siguientes funciones a partir de ésta: $\begin{cases} menores = itera & \dots \\ pliega f = itera & \dots \end{cases}$