

1. Simbolos utiles

\mathbb{Z}	<code>\ent</code>
\mathbb{R}	<code>\float</code>
Bool	<code>\bool</code>
True	<code>\True</code>
False	<code>\False</code>
\rightarrow	<code>\Then</code>
\leftrightarrow	<code>\Iff</code>
\longrightarrow	<code>\implica</code>
\wedge	<code>\$_\land\$</code>
\vee	<code>\$_\lor\$</code>
\neg	<code>\$_\lnot\$</code>
<code>if condicion then verdadero else falso</code>	<code>\IfThenElse{condicion}{verdadero}{falso}</code>
<code>[1..10]</code>	<code>\rango{1}{10}</code>
<code>(1..10)</code>	<code>\rangoac{1}{10}</code>
<code>[1..10)</code>	<code>\rangoca{1}{10}</code>
<code>(1..10)</code>	<code>\rangooa{1}{10}</code>
Float	<code>\rea</code>
Char	<code>\cha</code>
<code>prm(x)</code>	<code>\prm{x}</code>
<code>sgd(x)</code>	<code>\sgd{x}</code>
mcd	<code>\$_\mcd\$</code>
<code>[x]</code>	<code>\TLista{x}</code>
<code>[]</code>	<code>\lvacia</code>
<code>cons(xs)</code>	<code>\cons{xs}</code>
<code>indice(xs)</code>	<code>\indice{xs}</code>
<code>conc(xs)</code>	<code>\conc{xs}</code>
<code>cab(xs)</code>	<code>\cab{xs}</code>
<code>cola(x)</code>	<code>\cola{x}</code>
<code>sub(xs)</code>	<code>\sub{xs}</code>
<code>en(xs)</code>	<code>\en{xs}</code>
<code>++</code>	<code>\masmas</code>
<code>acum(expresion selector, condicion)</code>	<code>\acumselec{expresion}{selector}{condicion}</code>
<code>x ← [1..10]</code>	<code>\selector {x}{\rango{1}{10}}</code>

2. Definición de Problemas

```
problema sumar (a: $\mathbb{Z}$ , b: $\mathbb{Z}$ ) = res :  $\mathbb{Z}$  {
    requiere a  $\neq$  0;
    asegura res == a + b;
}
```

```
\begin{problema}{sumar}{a:\ent , b:\ent}{\ent}
\requiere{a \neq 0}
\asegura{\res == a+b}
\end{problema}
```

```
problema dividir (a: $\mathbb{Z}$ , b: $\mathbb{Z}$ ) = resultado :  $\mathbb{Z}$  {
    requiere puedoDividir : b  $\neq$  0;
    requiere elResultadoEsEntero : a mod b == 0;
    asegura divide : resultado == a/b;
}
```

```
\begin{problema}[resultado]{dividir}{a:\ent , b:\ent}{\ent}
\requiere[puedoDividir]{b \neq 0}
\requiere[elResultadoEsEntero]{a \textsf{mod} b == 0}
\asegura[divide]{\res == a/b}
\end{problema}
```

```

problema potencia (b:\mathbb{Z},e:\mathbb{Z}) {
  modifica b;
  asegura b == potencia(pre(b),e);
  aux potencia (b:\mathbb{Z},e:\mathbb{Z}) : \mathbb{Z} = b^e;
}

```

```

\begin{problema}{potencia}{b:\ent , e:\ent }{}
\modifica{b}
\asegura{b == potencia(\pre{b},e)}
\aux{potencia}{b:\ent , e:\ent }{\ent }{b^e}
\end{problema}

```

Otros comandos útiles (cuando no estamos en environment problema):

problema Sumar ((a:\mathbb{Z}, b:\mathbb{Z})) = result : \mathbb{Z}	\encabezadoDeProblema{result}{Sumar}{(a:\ent , b:\ent)}{\ent }
requiere positivo: a > 0	\requiereil{positivo}{a>0}
asegura divide: res = b/a	\asegurail{divide}{res = b/a}
modifica b	\modificail{b}
pre(b)	\pre{b}
aux potencia	\auxnom{potencia}

3. Definición de Tipos

```

tipo Edad = \mathbb{Z};
tipo Colores = rojo, amarillo, azul;

```

```

\sinonimo{Edad}{\ent }
\enum{Colores}{rojo , amarillo , azul}

```

```

tipo Complejo {
  observador real (c : Complejo) : \mathbb{Z};
  observador imag (c : Complejo) : \mathbb{Z};
  observador normaCuad (c: Complejo) : \mathbb{Z};
  La norma del Complejo al cuadrado
  observador angulo (c: Complejo) : \mathbb{Z};
  requiere existeAngulo : real(c) \neq 0 \wedge imag(c) \neq 0;
  invariante normaConsistente : normaCuad(c) == norma2(real(c), imag(c));
  aux norma2 (x:\mathbb{Z},y:\mathbb{Z}) : \mathbb{Z} = x^2 + y^2;
}

```

```

\begin{tipo}{Complejo}
\observador{real}{c : Complejo}{\ent }
\observador{imag}{c : Complejo}{\ent }
\observador{normaCuad}{c: Complejo}{\ent }
\explicacion{La norma del Complejo al cuadrado}
\observador{angulo}{c: Complejo}{\ent }
\requiere[existeAngulo]{ real(c) \neq 0 \wedge imag(c) \neq 0}
\invariante[normaConsistente]{ normaCuad(c) == norma2(real(c), imag(c))}
\aux{norma2}{x:\ent , y:\ent }{\ent }{x^2 + y^2}
\end{tipo}

```