

1. 可以. 构造函数 `void Evilx(i){`
 if (halt(Evilx,i)) return;
 else while(1);
 }

则 halt(p,i) 就无法判断 Evilx(i) 的停机问题

2. 加法: $a \oplus b = \begin{cases} \text{自然数} & a=\text{自然数}, b=\text{自然数} \\ \text{负} & a=\text{负}, b=\text{负} \\ \text{异} & \text{其他情况} \end{cases}$

除法: $a \oslash b = \begin{cases} \text{自然数} & a=\text{负}, b=\text{负} \\ \text{异} & \text{其他情况} \end{cases}$

当我们在分析 0/2 时, 原始域可得到 $\{\text{零}\} \oslash \{\text{正}\} = \{\text{零}\}$

而当前抽象域只能得到 $\{\text{自然数}\} \oslash \{\text{自然数}\} = \{\text{异}\}$, 不如原始分析精确