

(1)

假设存在停机问题判定算法 `bool Halt(p, input)`，其能够返回在 `p` 以 `input` 为输入时是否会停机。

考虑某个程序：

```
void Evil(input) {
    if(!Halt(Evil, input)) return ;
    else while(1);
}
```

在执行任意一个 `Evil(input)` 时：

如果程序会停止：`Halt(Evia1, input)` 返回值是 1，会进入 `while(1)` 部分，程序就不会停止，矛盾。

如果程序不会停止：`Halt(Evia1, input)` 返回值是 0，会进入 `return` 部分，程序就会停止，矛盾。

故不存在这样的的 `Halt` 函数。

(2)

加法	自然数	负	𐀀
自然数	自然数	𐀀	𐀀
负	𐀀	负	𐀀
𐀀	𐀀	𐀀	𐀀

除法	自然数	负	𐀀
自然数	自然数	负	𐀀
负	𐀀	自然数	𐀀
𐀀	𐀀	𐀀	𐀀

比如 a/b ，其中 a 是正数（本抽象域为自然数）， b 是负数，在本抽象域分析的结果是𐀀，在原抽象域结果能分析出来是负数，不如原始分析精确。