

软件分析 第1次作业

September 2023

庄嘉毅

1

假设存在停机问题的判定算法 $\text{bool Halt}(\text{program}, \text{input})$, 则可以构造邪恶程序 evil :

```
void evil(int x) {  
    if (Halt(evil, x)) {  
        while (true) {}  
    } else {  
        return;  
    }  
}
```

考察 $\text{Halt}(\text{evil}, 0)$ 的值:

- 如果 $\text{Halt}(\text{evil}, 0)$ 为真, 则 $\text{evil}(0)$ 会陷入死循环, 即 $\text{Halt}(\text{evil}, 0)$ 为假, 矛盾。
- 如果 $\text{Halt}(\text{evil}, 0)$ 为假, 则 $\text{evil}(0)$ 会返回, 即 $\text{Halt}(\text{evil}, 0)$ 为真, 矛盾。

因此, 不存在停机问题的判定算法。

2

加法 ($a + b$):

$a \backslash b$	自然数	负	罣
自然数	自然数	罣	罣
负	罣	负	罣
罣	罣	罣	罣

除法 (a / b):

$a \backslash b$	自然数	负	罣
自然数	罣	罣	罣
负	罣	自然数	罣
罣	罣	罣	罣

表达式 $1/x$, 在原始分析下,

- 如果 x 为正数, 则 $1/x$ 为正数;

- 如果 x 为负数，则 $1/x$ 为负数；
- 如果 x 为零或 ∞ ，则 $1/x$ 为 ∞ 。

而在新分析下， $1/x$ 始终为 ∞ 。因此，新的分析不如原始分析精确。