

1. 与无输入的程序同理。假设停机问题判定算法为  $\text{bool Halt}(p, i)$

邪恶程序为  $\text{void Evil}(i)$

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

对任意输入  $i$ , 若  $\text{Halt}(\text{Evil}, i)$  为真, 则  $\text{Evil}$  不停机;

若  $\text{Halt}(\text{Evil}, i)$  为假, 则  $\text{Evil}$  停机。均矛盾

故不存在停机问题判定算法。

2.

|     |     |   |   |
|-----|-----|---|---|
| +   | 自然数 | 负 | 假 |
| 自然数 | 自然数 | 假 | 假 |
| 负   | 假   | 负 | 假 |
| 假   | 假   | 假 | 假 |

|     |     |     |   |
|-----|-----|-----|---|
| /   | 自然数 | 负   | 假 |
| 自然数 | 假   | 假   | 假 |
| 负   | 假   | 自然数 | 假 |
| 假   | 假   | 假   | 假 |

若  $a, b, c$  均为负, 则  $a \div b \div c$  应为负, 但在该抽象域上分析为假