

# 软件分析技术 HW1

石显杰

September 2023

**Q1** 停机问题的证明定义在没有输入的函数上，能否改成在带输入的函数上？注意这时  $\text{Halt}(p, i)$  函数接受两个参数，其中  $i$  是输入。

**Ans:**

即证明对带输入的停机问题也是不可判定问题。

可设置邪恶程序如下：

```
void evil(i){  
    if(halt(evil,i)) while(1);  
    else return;  
}
```

本质上可以归约到停机问题，仍然可以通过 evil 程序对 halt 的调用设置矛盾：若  $\text{halt}(\text{evil}, i)$  返回真，则  $\text{evil}(i)$  不停机，矛盾；反之亦然

**Q2:** 假设我们把符号分析的抽象域改成自然数、负、罣三个值，其中自然数表示所有正数和零，请写出加法和除法的计算规则，并给出一个式子，在该抽象域上得到的结果不如原始分析精确。

**Ans:**

在新的抽象域自然数、负、罣中，我们可以定义加法和除法的计算规则如下：

**加法规则：**

$$\text{自然数} + \text{自然数} = \text{自然数}$$

$$\text{自然数} + \text{负} = \text{罣}$$

$$\text{自然数} + \text{罣} = \text{罣}$$

$$\text{负} + \text{自然数} = \text{罣}$$

$$\text{负} + \text{负} = \text{负}$$

$$\text{负} + \text{罣} = \text{罣}$$

$$\text{罣} + \text{自然数} = \text{罣}$$

$$\text{罣} + \text{负} = \text{罣}$$

$$\text{罣} + \text{罣} = \text{罣}$$

**除法规则：**

$$\text{自然数} / \text{自然数} = \text{自然数 (如果除尽)}$$

$$\text{自然数} / \text{负} = \text{罣}$$

$$\text{自然数} / \text{罣} = \text{罣}$$

$$\text{负} / \text{自然数} = \text{罣 (除数可能为零)}$$

$$\text{负} / \text{负} = \text{自然数 (如果除尽)}$$

$$\text{负} / \text{罣} = \text{罣}$$

$$\text{罣} / \text{自然数} = \text{罣}$$

$$\text{罣} / \text{负} = \text{罣}$$

$$\text{罣} / \text{罣} = \text{罣}$$

现在让我们考虑一个式子：自然数 / 负。如果是原始分析，则是正 / 负 = 负以及零 / 负 = 零这两种精确的结果。根据除法规则，这将得到 罣。这是因为无法确定被除数是零还是正数，从而得出 罣，结果不精确，因为我们通过这种抽象方式损失了一些信息。