

1. Is P necessary for utrb?

What is the weakest failure detector for utrb? Exercise 9.1 Prove that P is the weakest failure detector to implement utrb Hint: Try to implement P assuming utrb (and reliable channels) This would mean that P and utrb are equivalent in a system with reliable channels

在这个练习中，我们被要求证明P（完美故障检测器）是实现均匀终止可靠广播

（Uniform Terminating Reliable Broadcast，简称utrb）的最弱故障检测器。我们可以通过假设utrb和可靠通道的存在来实现P，从而证明P是必要的，并且是实现utrb的最弱故障检测器。以下是证明过程的中文解释：

解决方案：

- 我们假设每个进程 p_i 可以使用无限数量的utrb实例，其中 p_i 是发送者 src 。
- 每个进程 p_i 在一个无限循环中不断地通过utrb广播消息。
- 当任何进程 p 通过utrb接收到特殊符号 Δ 时，它触发了一个 $P, Crash \mid p$ 事件，表明进程 p 崩溃了。

```
while (true) do
  k++
  trigger < utrb, Broadcast | k >
end loop

upon event < utrb, Deliver | p,  $\Delta$  > do
  trigger < P, Crash | p >
```

证明逻辑：

1. **utrb实现P：** 通过utrb实例不断地广播消息，并在接收到 Δ 时报告进程崩溃，我们可以使用utrb来模拟完美故障检测器P的行为。每当utrb决定 Δ （表示源进程崩溃），我们可以将其解释为故障检测器P检测到某个进程的崩溃。
2. **P是最弱的故障检测器：** 如果我们能够只使用utrb和可靠通道来实现P，这意味着没有比P更弱的故障检测器可以用来实现utrb。这是因为utrb本身足以实现P的功能。

综上所述，P是实现utrb所需的最弱故障检测器。这个证明表明，如果我们有utrb和可靠通道，我们就可以实现P，而不需要任何比P更强的故障检测功能。

2. P is clearly sufficient for gm Is it necessary? Exercise 9.2 Can you implement P using gm?

3. 这个练习考察的是完美故障检测器（P）是否是实现群组成员关系（Group Membership, gm）的必要条件。具体问题是，我们是否可以使用gm来实现P。以下是针对这个问题的解决方案和解释：

解决方案：

通过gm实现P的伪代码如下：

```
upon event < P, Init > do
    correct :=  $\Pi$ 

upon event < gm, View | (id, M) >
    forall p  $\in$  correct \ M do
        trigger < P, Crash | p >
    correct := M
```

解释:

1. **初始化:** 在 `P, Init` 事件中, 我们初始化 `correct` 变量, 它表示当前系统中认为是正确的 (未崩溃的) 所有进程的集合。
2. **处理视图更改:** 当 `gm, View` 事件被触发时, 表示 `gm` 模块已经安装了一个新的视图。新视图中包含了当前被认为是活跃的进程集合 `M`。
3. **更新故障检测:** 对于那些在上一个视图中被认为是正确的但不在新视图 `M` 中的进程, 我们触发 `P, Crash` 事件。这意味着这些进程被认为已经崩溃了。
4. **更新正确的进程集合:** 最后, 更新 `correct` 变量, 使其仅包含新视图 `M` 中的进程。