

Lab-1 总结

报告

1. 内容

报告不要求内容长，但要能体现你做的事情和你的思考。有的同学只是单纯的复述实验步骤，这样不能体现出你是否充分理解了本次实验。你可以在报告中写下你对某些部分是如何理解的，以及你在完成的过程中遇到了什么问题、是如何解决的，你有什么样的反思。

对细节的洞察力，和思考解决问题的能力是区分个人水平的重要标准。

2. 格式

格式尽可能工整，尤其是对代码块、标准输出等内容的组织要清晰。事实上，比起 Word，我们更推荐你使用 Markdown 来写实验报告。Markdown 是开源社区等平台广泛使用的文档格式之一，可以方便快捷地编写代码和公式等，建议不熟悉的同学了解一下。

3. 截图

部分同学的实验报告截图中，显示的命令是做实验时输错了的命令，建议同学们在实验报告的截图中尽可能不要留有这样输错的命令，尤其是那种易和正确命令相混淆的命令，这会让我们误会你是执行错误了。

4. 英文报告

英文只是锦上添花，我们关注的依然是你的内容。因此，如果使用英文极大限制了你的表达，无法满足我们在第一点提到的要求，只会弄巧成拙。我们依然鼓励你使用英文。而如果你的能力有限，部分内容使用中文也是可以接受的。

代码

- 部分同学对 `try...catch...` 语法的理解不正确，导致许多同学在 `try` 内对“in”的数量进行更新。放在这里执行结果正确，但逻辑上不甚合理。
- 部分同学对 Mininet 和 Switchyard 的关系理解不恰当。建议再看一看手册上的 [示意图](#)、Lab-2 的 PPT 以及 [讨论区](#)。
- 从 Lab-2 开始提交前要注意你的仓库要能通过 GitHub Action，即文件目录和命名要正确。具体参见 [手册](#)。
- 以后的实验要注意最终的提交能运行通过 testcases 里的 TestScenario 测试。例如 Lab-2 中如果你为了在 Mininet 中更准确地测试而改动了 Cache Table 的大小，一定记得提交到 GitHub 时恢复成测试用例里对应的大小。

反馈

我们通过 GitHub pull request 进行回复。回复后，默认情况你也会收到一个邮件。你可以选择邮件回复，但我们更建议你在 GitHub 的页面进行回复（在页面中可以使用上文提到的 Markdown 格式进行回复，格式更加丰富）。在你的项目仓库中，点击“Pull request”，选择“Feedback”即可看到我们对你实验的回复。需要说明的是，这只是我们给你的反馈，并不直接意味着评分高低。

其它

1. 没有特殊情况不要超过 Deadline，会根据迟交时间长短扣分。
2. 尽管 Lab 1 没什么代码量，但我们通过你们的报告和验收，依然能发现很多同学做的不认真，甚至可能存在不是自己做的情况，请相关同学自重。