

Check6 实验报告

王昕浩 221502005

1. Structure and Design

使用以 map 为索引的树形结构来储存 ip 前缀。以前缀的共有部分作为比较的数值。

需要注意该比较有非自反性，传递性，非对称性，但其衍生的等于关系没有传递性，因此其不满足 c++ 对 map 的 key 所要求的比较要求。但因为我在其中设计的树形结构规避了这一点，所以该程序是形式上正确的，且实际上正确。

使 Router 类继承自 RouterMapped 类作为根节点。

使用递归方法 insert 和 find 以查询 IP 前缀，当查询不到时返回一个 *interface_num* = -1 的 RouterMapped 对象。

2. Implementation Challenges

本实验无较大难点。

需注意，在之前的 network 的 send_datagram 的实现里，需要先将 datagram 插入缓存再传输信息。否则，对 check5 不会有影响，但会使 check6 无法通过，因为 check6 中是在 transmit 函数中引用 send_datagram 和 recv_datagram 的，使得 recv_datagram 被调用时 datagram 还未插入缓存。

需要在 ttl 自减之后调用 compute_checksum() 以保证信息合法，需要在 ttl 自减前判断是否可以合法自减。

3. Remaining Bugs

没有剩余的 bug。

4. Experimental results and performance

完成截图如下：

```
● wexther@wxh:~/minnow$ cmake --build build -- format
[100%] Formatting source code...
[100%] Built target format
● wexther@wxh:~/minnow$ cmake --build build -- check6
Test project /home/wexther/minnow/build
  Start 1: compile with bug-checkers
1/3 Test #1: compile with bug-checkers ..... Passed    0.24 sec
  Start 35: net_interface
2/3 Test #35: net_interface ..... Passed    0.02 sec
  Start 36: router
3/3 Test #36: router ..... Passed    0.02 sec

100% tests passed, 0 tests failed out of 3

Total Test time (real) =  0.29 sec
Built target check6
● wexther@wxh:~/minnow$ cmake --build build -- tidy
Built target tidy_src__wrapping_integers.cc
Built target tidy_src__byte_stream_helpers.cc
Built target tidy_src__byte_stream.cc
Built target tidy_src__tcp_receiver.cc
Built target tidy_src__reassembler.cc
Built target tidy_src__tcp_minnow_socket.cc
Built target tidy_src__tcp_sender.cc
Built target tidy_src__network_interface.cc
Built target tidy_src__router.cc
Built target tidy
○ wexther@wxh:~/minnow$
```