

edge的另一个端

在存储 edge 时，一般会将两端以 edge.u 和 edge.v 的形式存在 edge 中。

假设我知道了 edge 的一个端点 a，现在想要快速拿到了一个端点 b，但我不知道 a 对应的是 edge.u 还是 edge.v，可以用下面的判断语句：

```
1  # 假设 edge.u = 1, edge.v = 2。a=2
2
3  if a == edge.u:
4      b = edge.v
5  else:
6      b == edge.u
```

上面的方法需要使用 if-else，相对较慢，可以直接采用异或运行：

通过 `edge.u ^ edge.v ^ a` 这种方式来获取边的另一个端点，是利用异或运算（XOR）的一个特性： $a \oplus a = 0$ ，且 $a \oplus 0 = a$ 。

具体来说，假设一条边连接了两个顶点 `u` 和 `v`，你已经知道了其中一个端点 `a`，现在想得到另一个端点 `b`。通过 `edge.u ^ edge.v ^ a` 的运算可以去除已知的端点 `a`，得到未知的端点 `b`。