

# PublicBoard 测试用例

---

## PublicBoard Test Case

**项目名称：**PublicBoard

**摘 要：**本文档系统阐述 PublicBoard 信息共享平台项目软件各模块的白盒测试与黑盒测试。

**版权所有：**除特别允许只在项目组内部使用

**修改记录：**

日期	版本	说明	作者
2021-1-1	V1.0	PublicBoard 白盒黑盒测试用例	邹秉禧

# 目录

<b>1 用户登录模块.....</b>	<b>4</b>
<b>1.1 白盒测试 .....</b>	<b>4</b>
1.1.1 程序流程图: .....	4
1.1.2 流图.....	5
1.1.3 流图的环形复杂度.....	6
1.1.4 独立路径的基本组合 .....	6
1.1.5 条件组合覆盖测试用例.....	6
<b>1.2 黑盒测试 .....</b>	<b>8</b>
1.2.1 有效等价类划分 .....	8
1.2.2 有效等价类测试用例 .....	8
<b>2 用户注册模块.....</b>	<b>9</b>
<b>2.1 白盒测试 .....</b>	<b>9</b>
2.1.1 程序流程图.....	9
2.1.2 流图.....	10
2.1.3 流图的环形复杂度.....	11
2.1.4 独立路径的基本组合 .....	11
2.1.5 条件组合覆盖测试用例.....	11
<b>2.2 黑盒测试 .....</b>	<b>14</b>
2.2.1 有效等价类划分 .....	14
2.2.2 有效等价类测试用例 .....	14
<b>3 帖子点赞模块.....</b>	<b>16</b>
<b>3.1 白盒测试 .....</b>	<b>16</b>
3.1.1 程序流程图.....	16
3.1.2 流图.....	17
3.1.3 流图的环形复杂度.....	18
3.1.4 独立路径的基本组合 .....	18
3.1.5 条件组合覆盖测试用例.....	18
<b>3.2 黑盒测试 .....</b>	<b>20</b>
3.2.1 等价类划分 .....	20
3.2.2 有效等价类测试用例 .....	20
3.2.3 无效等价类测试用例 .....	22
<b>4 忘记密码模块.....</b>	<b>23</b>
<b>4.1 白盒测试 .....</b>	<b>23</b>
4.1.1 程序流程图.....	23
4.1.2 流图.....	24
4.1.3 流图的环形复杂度.....	25

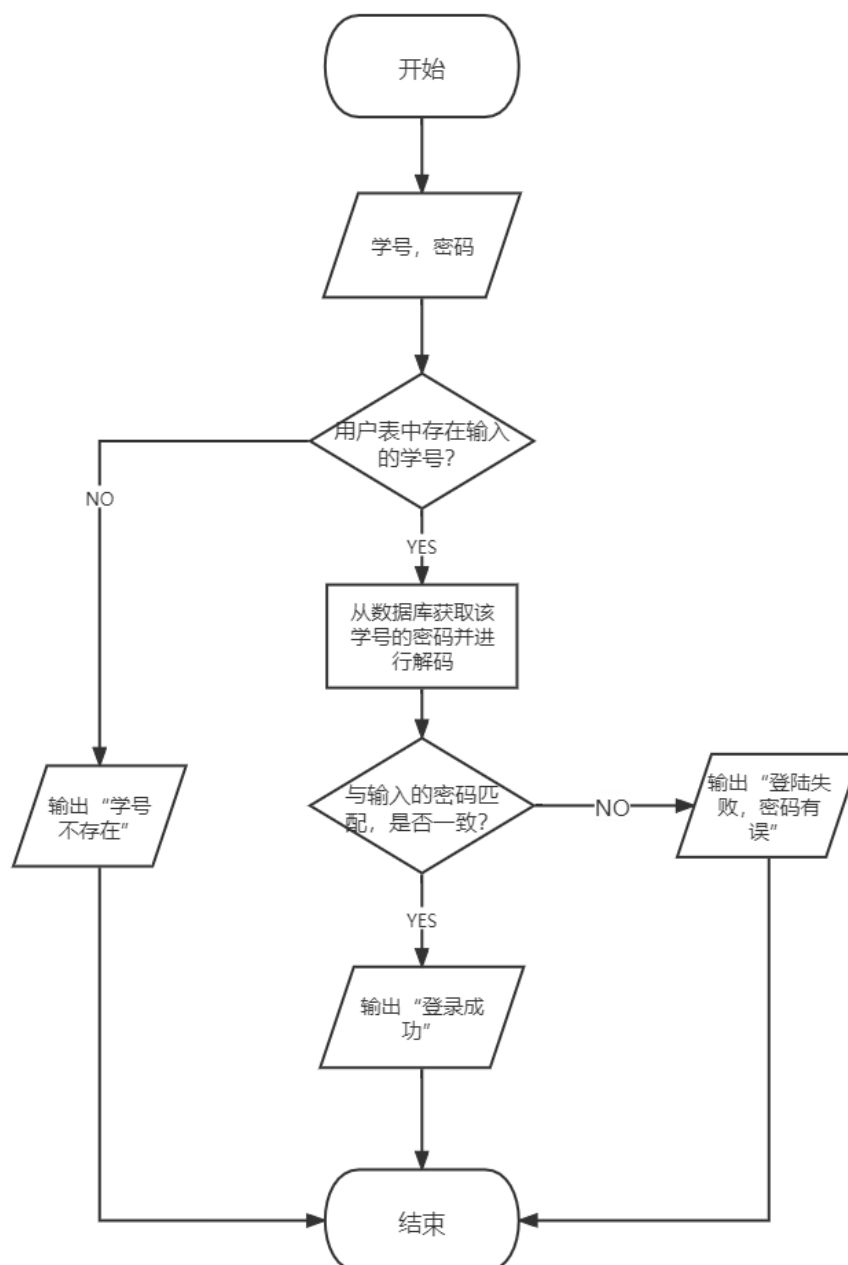
---

4.1.4 独立路径的基本组合 .....	25
4.1.5 条件组合覆盖测试用例.....	25
<b>4.2 黑盒测试 .....</b>	<b>28</b>
4.2.1 等价类划分 .....	28
4.2.2 有效等价类测试用例 .....	28
<b>5 修改邮箱模块.....</b>	<b>29</b>
<b>5.1 白盒测试 .....</b>	<b>29</b>
5.1.1 程序流程图: .....	29
5.1.2 流图.....	30
5.1.3 流图的环形复杂度.....	31
5.1.4 独立路径的基本组合 .....	31
5.1.5 条件组合覆盖测试用例.....	31
<b>5.2 黑盒测试 .....</b>	<b>34</b>
5.2.1 有效等价类划分 .....	34
5.2.2 有效等价类测试用例 .....	34
<b>6 修改密码模块.....</b>	<b>35</b>
<b>6.1 白盒测试 .....</b>	<b>35</b>
6.1.1 程序流程图.....	35
6.1.2 流图.....	36
6.1.3 流图的环形复杂度.....	37
6.1.4 独立路径的基本组合 .....	37
6.1.5 条件组合覆盖测试用例.....	37
<b>6.2 黑盒测试 .....</b>	<b>38</b>
6.2.1 有效等价类划分 .....	38
6.2.2 有效等价类测试用例 .....	38

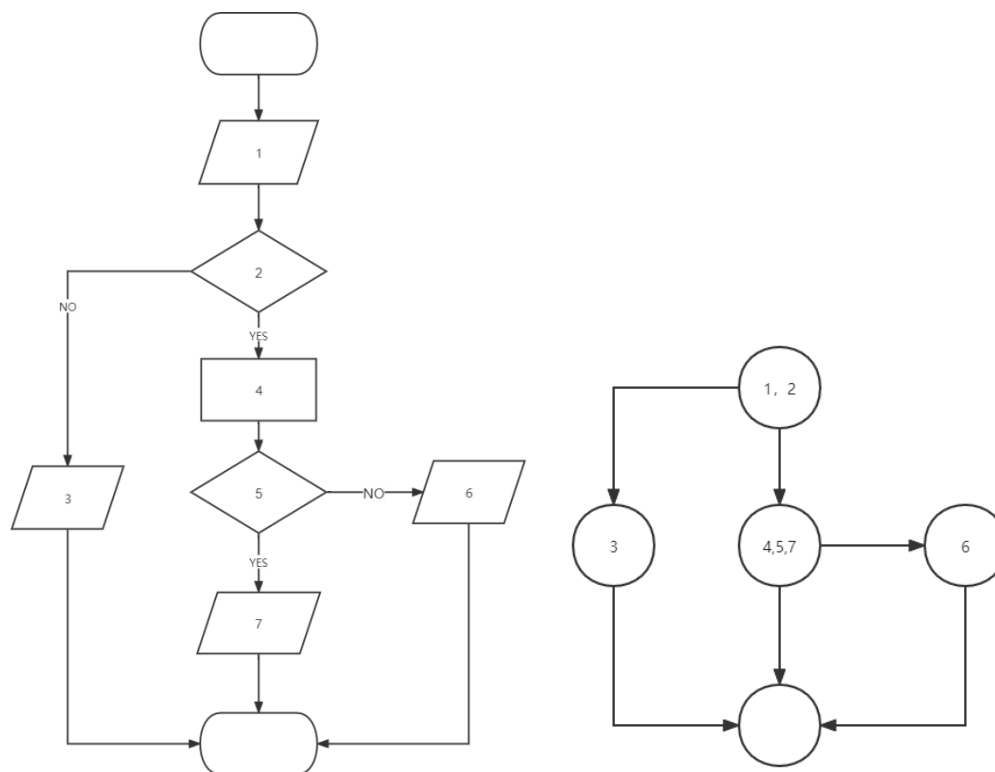
# 1 用户登录模块

## 1.1 白盒测试

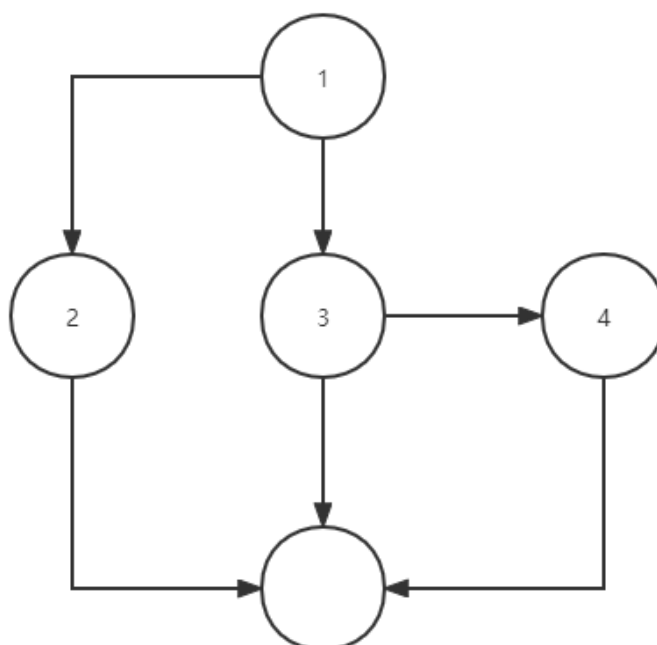
### 1.1.1 程序流程图：



### 1.1.2 流图



简化后流图如下：



### 1.1.3 流图的环形复杂度

流图中的判定节点数(1,2)，故环形复杂度为  $2+1=3$ 。

### 1.1.4 独立路径的基本组合

Path1: 1-3

Path2: 1-2

Path3: 1-3-4...


### 1.1.5 条件组合覆盖测试用例

$S = (\text{账号存在} \parallel \text{账号不存在}) \&\& (\text{密码正确} \parallel \text{密码不正确})$

	账号存在 $\parallel$ 账号不存在	密码正确 $\parallel$ 密码不正确	S
Path1	T	T	T
Path2	F	-	F
Path3	T	F	F


独立路径的实例：

(1) 测试登录成功


用例名称	测试登陆成功		
测试目的	测试学号存在且密码正确	覆盖路径	Path1
测试预置条件： 登录界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤： 1) 进入登录界面，输入用户的学号 2019010101 和密码 123456789  2) 点击  按钮，进行登录			
期望结果： 1) 用户正常登录，提示“登录成功”，并跳转到首页			
实际结果： 1) 用户用户正常登录，提示“登录成功”，并跳转到首页			

说明:

(2) 测试登录失败 1

用例名称	测试登录失败 1		
测试目的	测试学号不存在	覆盖路径	Path2
测试预置条件: 登录界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤: 1) 进入登录界面, 输入用户的学号 2019010101 和密码 123456789  2) 点击  按钮, 进行登录			
期望结果: 1) 用户登录失败, 提示“该学号不存在”, 并停留在登录界面			
实际结果: 1) 用户登录失败, 提示“该学号不存在”, 并停留在登录界面			
说明:			

(3) 测试登录失败 2

用例名称	测试登录失败 2		
测试目的	测试学号正确但密码错误	覆盖路径	Path3
测试预置条件: 登录界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤: 1) 进入登录界面, 输入用户的学号 2019010101 和密码 987654321  2) 点击  按钮, 进行登录			
期望结果: 1) 用户登录失败, 提示“登录失败, 密码有误”, 并停留在登录界面			
实际结果:			

1) 用户登录失败，提示“登录失败，密码有误”，并停留在登录界面
说明:


## 1.2 黑盒测试

### 1.2.1 有效等价类划分

输入等价类	有效等价类
学号	用户表中学号存在 (1)
密码	密码正确 (2)

### 1.2.2 有效等价类测试用例

#### 1) 测试登录成功

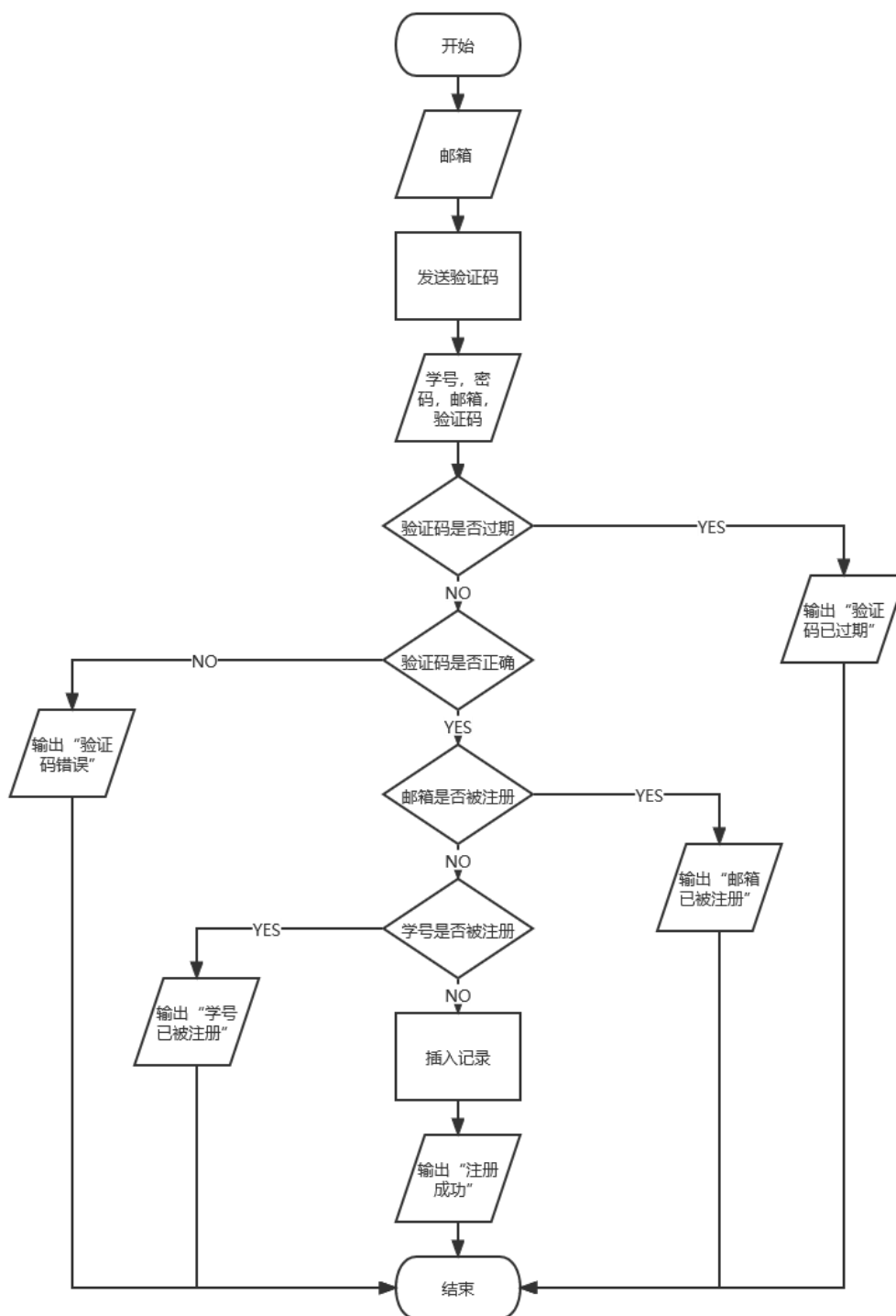
用例名称	测试登陆成功		
测试目的	测试用户表中学号存在且密码正确	覆盖范围	等价类(1)、(2)
测试预置条件: 登录界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤: 1) 进入登录界面，输入用户的学号 2019010101 和密码 123456789  2) 点击  按钮，进行登录			
期望结果: 1) 用户正常登录，提示“登录成功”，并跳转到首页			
实际结果: 1) 用户用户正常登录，提示“登录成功”，并跳转到首页			



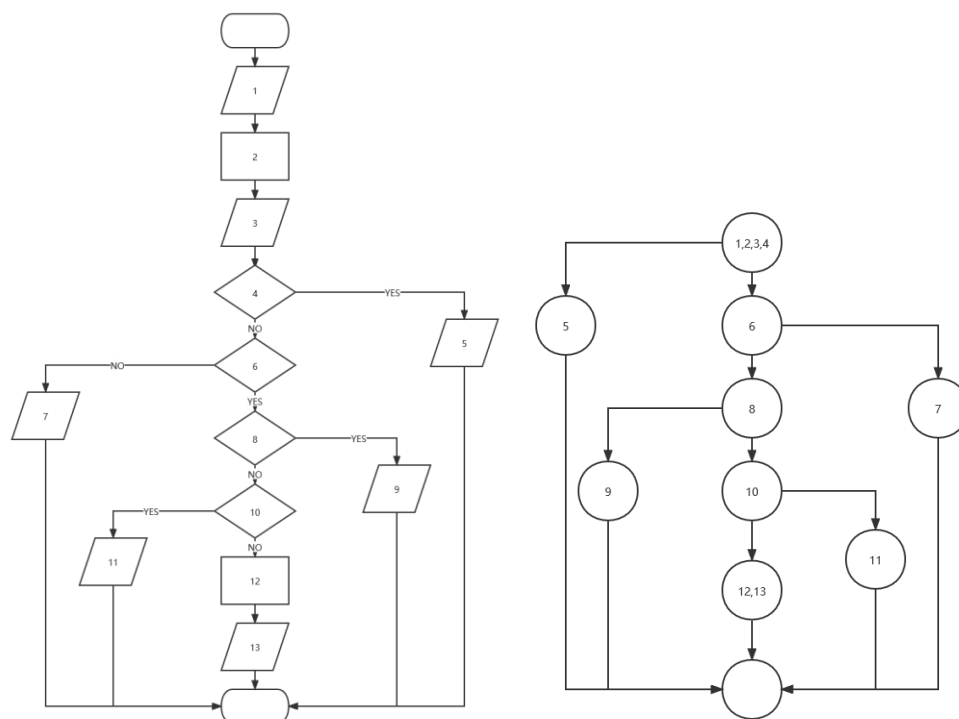
## 2 用户注册模块

### 2.1 白盒测试

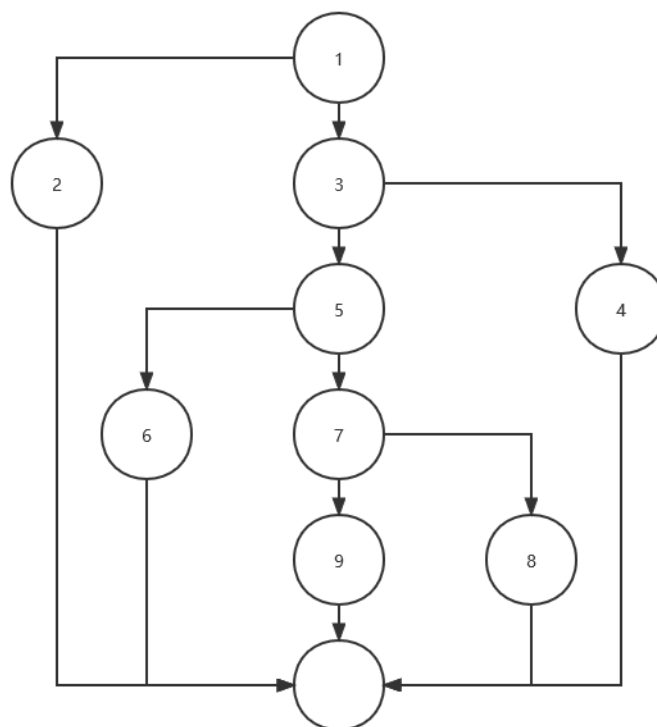
#### 2.1.1 程序流程图



## 2.1.2 流图



简化后流图如下：



### 2.1.3 流图的环形复杂度

流图中的判定节点数(1,3,5,7)，故环形复杂度为  $4+1=5$ 。

### 2.1.4 独立路径的基本组合

Path1: 1-2-9

Path2: 1-3-4-9

Path3: 1-3-5-6-9

Path4: 1-2-5-7-8-9

Path5: 1-3-5-7-9

### 2.1.5 条件组合覆盖测试用例

S = (验证码已过期 || 验证码未过期) && (验证码正确 || 验证码不正确) && (邮箱未被注册 || 邮箱已被注册) && (学号未被注册 || 学号已被注册)

	验证码已过期    验证码未过期	验证码正确    验证码不正确	邮箱未被注册    邮箱已被注册	学号未被注册    学号已被注册	S
Path1	T	-	-	-	F
Path2	F	F	-	-	F
Path3	F	T	F	-	F
Path4	F	T	T	F	F
Path5	F	T	T	T	T

独立路径的实例:

(1)测试注册成功

用例名称	测试注册成功		
测试目的	测试注册成功	覆盖路径	Path5
测试预置条件: 注册界面正常运行。数据库中不存在用户学号 2019152010, 且不存在用户邮箱 1564068065@qq.com, 向指定邮箱发送验证码, 获取验证码。			
测试步骤: 1) 进入登录界面, 输入用户的学号 2019152010 和密码 123456789 和邮箱 1564068065@qq.com 和验证码 2) 点击“注册”按钮, 进行注册			

期望结果:

1) 用户正常登录, 提示“注册成功”, 并跳转到登录界面

实际结果:

1) 用户正常登录, 提示“注册成功”, 并跳转到登录界面

说明:

## (2)测试注册失败 1

用例名称	测试注册失败 1		
测试目的	测试验证过期	覆盖路径	Path1
测试预置条件: 注册界面正常运行。数据库中不存在用户学号 2019152010, 且不存在用户邮箱 1564068065@qq. com 向指定邮箱发送验证码, 且等待 5 分钟后验证码过期。			
测试步骤: 1) 进入注册界面, 输入用户的学号 2019152010 和密码 123456789 和邮箱 1564068065@qq. com 和验证码 2) 点击“注册”按钮, 进行注册			
期望结果: 1) 用户注册失败, 提示“验证码已过期”, 并停留在注册界面			
实际结果: 1) 用户注册失败, 提示“验证码已过期”, 并停留在注册界面			
说明:			

## (3)测试注册失败 2

用例名称	测试注册失败 2		
测试目的	测试验证码错误	覆盖路径	Path2
测试预置条件: 注册界面正常运行。数据库中不存在用户学号 2019152010, 且不存在用户邮箱 1564068065@qq. com 向指定邮箱发送验证码, 获取验证码。			
测试步骤: 1) 进入注册界面, 输入用户的学号 2019152010 和密码 123456789 和邮箱 1564068065@qq. com 和验证码 2) 点击“注册”按钮, 进行注册			

期望结果: 1) 用户注册失败, 提示“验证码错误”, 并停留在注册界面
实际结果: 1) 用户注册失败, 提示“验证码错误”, 并停留在注册界面
说明:

## (4)测试注册失败 3

用例名称	测试注册失败 3		
测试目的	测试账号已被注册	覆盖路径	Path3
测试预置条件: 注册界面正常运行。数据库中已存在用户学号 2019152010, 向指定邮箱发送验证码。			
测试步骤: 1) 进入注册界面, 输入用户的学号 2019152010 和密码 123456789 和邮箱 1564068065@qq. com 和验证码 2) 点击“注册”按钮, 进行注册			
期望结果: 1) 用户注册失败, 提示“该账号已注册”, 并停留在注册界面			
实际结果: 1) 用户注册失败, 提示“该账号已注册”, 并停留在注册界面			
说明:			

## (5)测试登录失败 4

用例名称	测试登录失败 4		
测试目的	测试邮箱已被注册	覆盖路径	Path4
测试预置条件: 注册界面正常运行。数据库中已存在用户邮箱 1564068065@qq. com。向指定邮箱发送验证码, 获取验证码。			
测试步骤: 1) 进入注册界面, 输入用户的学号 2019152010 和密码 123456789 和邮箱 1564068065@qq. com 和验证码 2) 点击“注册”按钮, 进行注册			

期望结果:

- 1) 用户注册失败, 提示“该邮箱已注册”, 并停留在注册界面

实际结果:

- 1) 用户注册失败, 提示“该邮箱已注册”, 并停留在注册界面

说明:

## 2.2 黑盒测试

### 2.2.1 有效等价类划分

输入等价类	有效等价类
学号和密码	学号未存在 (1)
邮箱	邮箱未存在 (2)
验证码	验证码未过期 (3)
	验证码正确 (4)

### 2.2.2 有效等价类测试用例

#### 1) 测试注册成功

用例名称	测试注册成功		
测试目的	测试注册成功	覆盖范围	等价类 (1) (2) (3) (4)
测试预置条件: 注册界面正常运行。数据库中不存在用户学号 2019152010, 且不存在用户邮箱 1564068065@qq.com, 向指定邮箱发送验证码, 获取验证码。			

测试步骤:

- 1) 进入登录界面, 输入用户的学号 2019152010 和密码 123456789 和邮箱 1564068065@qq. com 和验证码
- 2) 点击“注册”按钮, 进行注册

期望结果:

- 1) 用户正常登录, 提示“注册成功”, 并跳转到登录界面

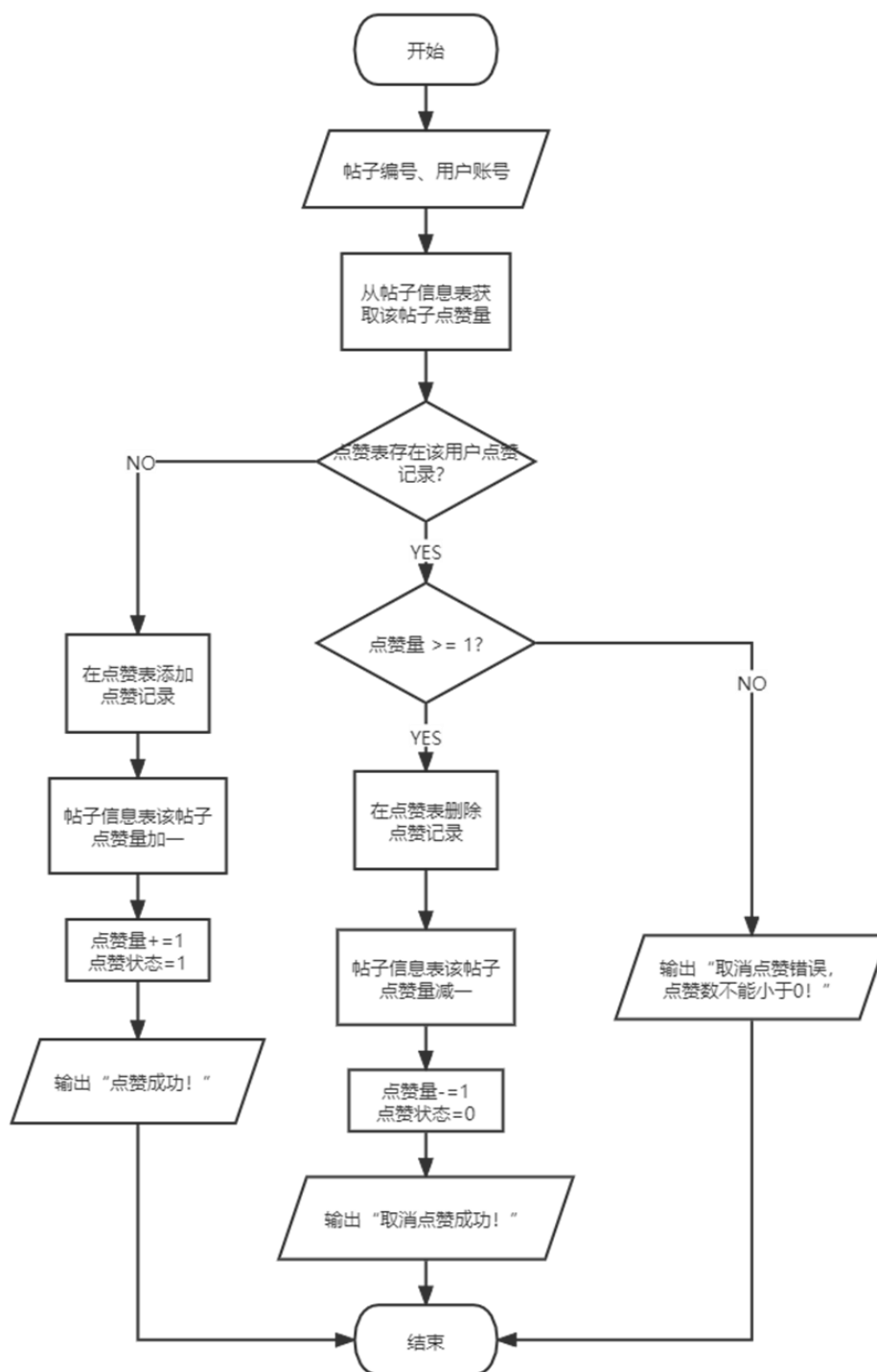
实际结果:

- 1) 用户正常登录, 提示“注册成功”, 并跳转到登录界面

## 3 帖子点赞模块

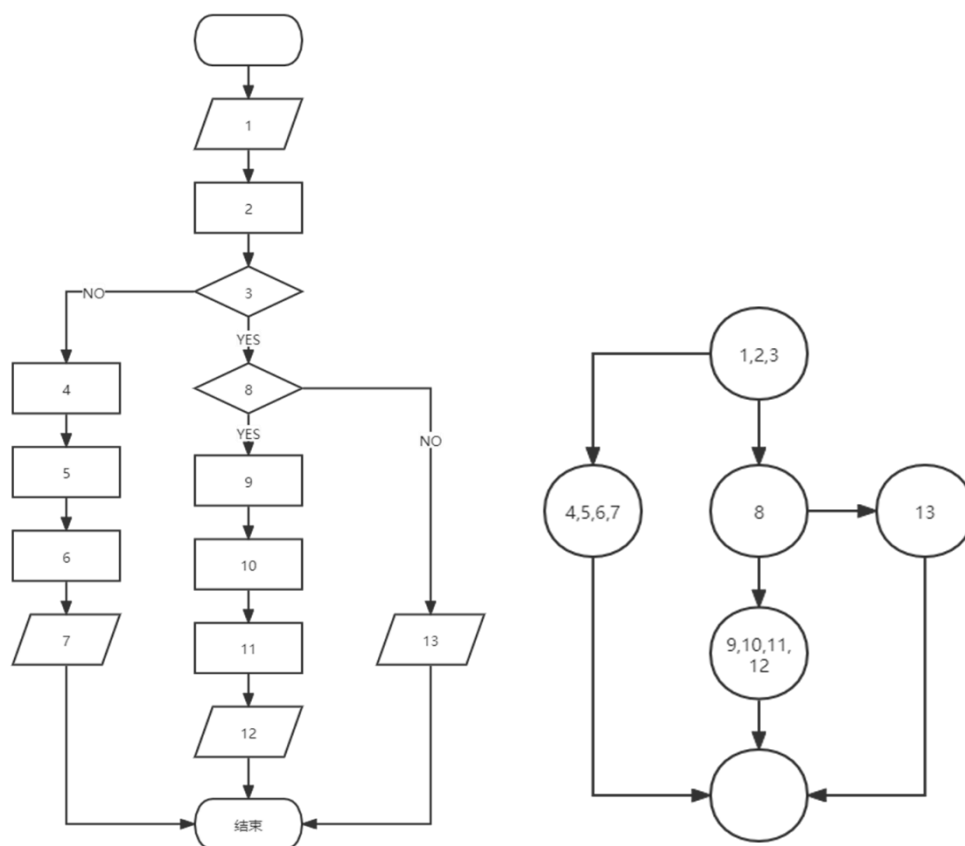
### 3.1 白盒测试

#### 3.1.1 程序流程图

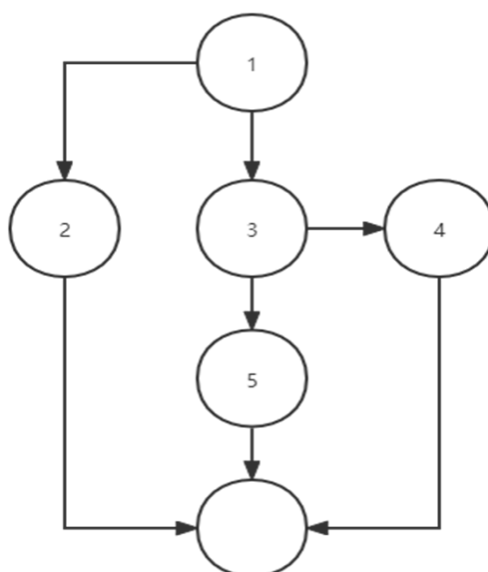




### 3.1.2 流图



简化后流图如下：



### 3.1.3 流图的环形复杂度

流图中的判定节点数(1, 3)，故环形复杂度为  $2+1=3$ 。

### 3.1.4 独立路径的基本组合

Path1: 1-2

Path2: 1-3-5

Path3: 1-3-4


### 3.1.5 条件组合覆盖测试用例

$S = (\text{点赞表存在该用户点赞记录} \parallel \text{点赞表不存在该用户点赞记录}) \&\& (\text{点赞量大于等于 } 1 \parallel \text{点赞小于 } 1)$


	点赞表存在该用户点赞记录 $\parallel$ 点赞表不存在该用户点赞记录	点赞量大于等于 1 $\parallel$ 点赞小于 1	S
Path1	F	-	F
Path2	T	T	T
Path3	T	F	F

独立路径的实例：

(1) 测试点赞成功



用例名称	测试点赞成功		
测试目的	测试用户未点赞，对帖子进行点赞成功	覆盖路径	Path1
测试预置条件： 任一帖子详情页面正常运行。数据库中存在用户账号 2019010101。用户未对该帖子进行点赞。点赞表不存在用户账号 2019010101 对该帖子的点赞记录。			
测试步骤： 1) 登陆账号，在主页点击任一帖子标题进入帖子详情界面 2) 点击  点赞按钮，对该篇帖子点赞			
期望结果： 1) 提示“点赞成功！”，白色爱心变为红色爱心，点赞量加一			

实际结果:

- 1) 提示“点赞成功!”,  白色爱心变为红色爱心, 并且点赞量由 1 变为 2

说明:

## (2) 测试取消点赞成功

用例名称	测试取消点赞成功		
测试目的	测试用户已点赞, 对帖子取消点赞成功	覆盖路径	Path2
测试预置条件: 任一帖子详情页面正常运行。数据库中存在用户账号 2019010101。用户已经对该帖子进行点赞。点赞表存在用户账号 2019010101 对该帖子的点赞记录。			
测试步骤: 1) 登陆账号, 在主页点击任一帖子标题进入帖子详情界面  2) 点击  点赞按钮, 对该篇帖子取消点赞			
期望结果: 1) 提示“取消点赞成功!”, 红色爱心变为白色爱心, 点赞量减一			
实际结果:  1) 提示“取消点赞成功!”,  红色爱心变为白色爱心, 并且点赞量由 2 变为 1			
说明:			

## (3) 测试取消点赞失败

用例名称	测试取消点赞失败		
测试目的	测试用户已点赞, 对帖子取消点赞失败的异常情况	覆盖路径	Path3
测试预置条件: 任一帖子详情页面正常运行。数据库中存在用户账号 2019010101。用户已经对该帖子进行点赞。点赞表存在用户账号 2019010101 对该帖子的点赞记录。			

测试步骤:
1) 登陆账号, 在主页点击任一帖子标题进入帖子详情界面
2) 点击  点赞按钮, 对该篇帖子取消点赞
期望结果:
1) 提示“取消点赞错误, 点赞数不能小于 0!”
实际结果:
1) 提示“取消点赞错误, 点赞数不能小于 0!”
说明:

## 3.2 黑盒测试

### 3.2.1 等价类划分

输入等价类	有效等价类	无效等价类
是否点赞成功	点赞表不存在该用户账号对该帖子编号点赞记录 (1)	点赞表存在该用户账号对该帖子编号点赞记录 (4)
是否取消点赞成功	点赞表存在该用户账号对该帖子编号点赞记录 (2)	点赞表不存在该用户账号对该帖子编号点赞记录 (5)
点赞量	点赞表存在该用户账号对该帖子编号点赞记录, 点赞量大于等于 1 (3)	点赞表存在该用户账号对该帖子编号点赞记录, 点赞量小于 1 (6)

### 3.2.2 有效等价类测试用例

#### 1) 测试点赞成功


用例名称	测试点赞成功		
测试目的	测试用户未点赞, 对帖子进行点赞成功	覆盖范围	等价类 (1) (3)

## 测试预置条件:

任一帖子详情页面正常运行。数据库中存在用户账号 2019010101。用户未对该帖子进行点赞。点赞表不存在用户账号 2019010101 对该帖子的点赞记录。

## 测试步骤:


1) 登陆账号, 在主页点击任一帖子标题进入帖子详情界面

2) 点击  点赞按钮, 对该篇帖子点赞

## 期望结果:



1) 提示“点赞成功!”, 白色爱心变为红色爱心, 点赞量加一

## 实际结果:

1) 提示“点赞成功!”,  白色爱心变为红色爱心, 并且点赞量由 1 变为 2

## 说明:


## 2) 测试取消点赞成功

用例名称	测试取消点赞成功		
测试目的	测试用户已点赞, 对帖子取消点赞成功	覆盖范围	等价类 (2) (3)
测试预置条件: 任一帖子详情页面正常运行。数据库中存在用户账号 2019010101。用户已经对该帖子进行点赞。点赞表存在用户账号 2019010101 对该帖子的点赞记录。			
测试步骤: 1) 登陆账号, 在主页点击任一帖子标题进入帖子详情界面  2) 点击  点赞按钮, 对该篇帖子取消点赞			
期望结果: 1) 提示“取消点赞成功!”, 红色爱心变为白色爱心, 点赞量减一			
实际结果:  1) 提示“取消点赞成功!”,  红色爱心变为白色爱心, 并且点赞量由 2 变为 1			

说明:

### 3.2.3 无效等价类测试用例

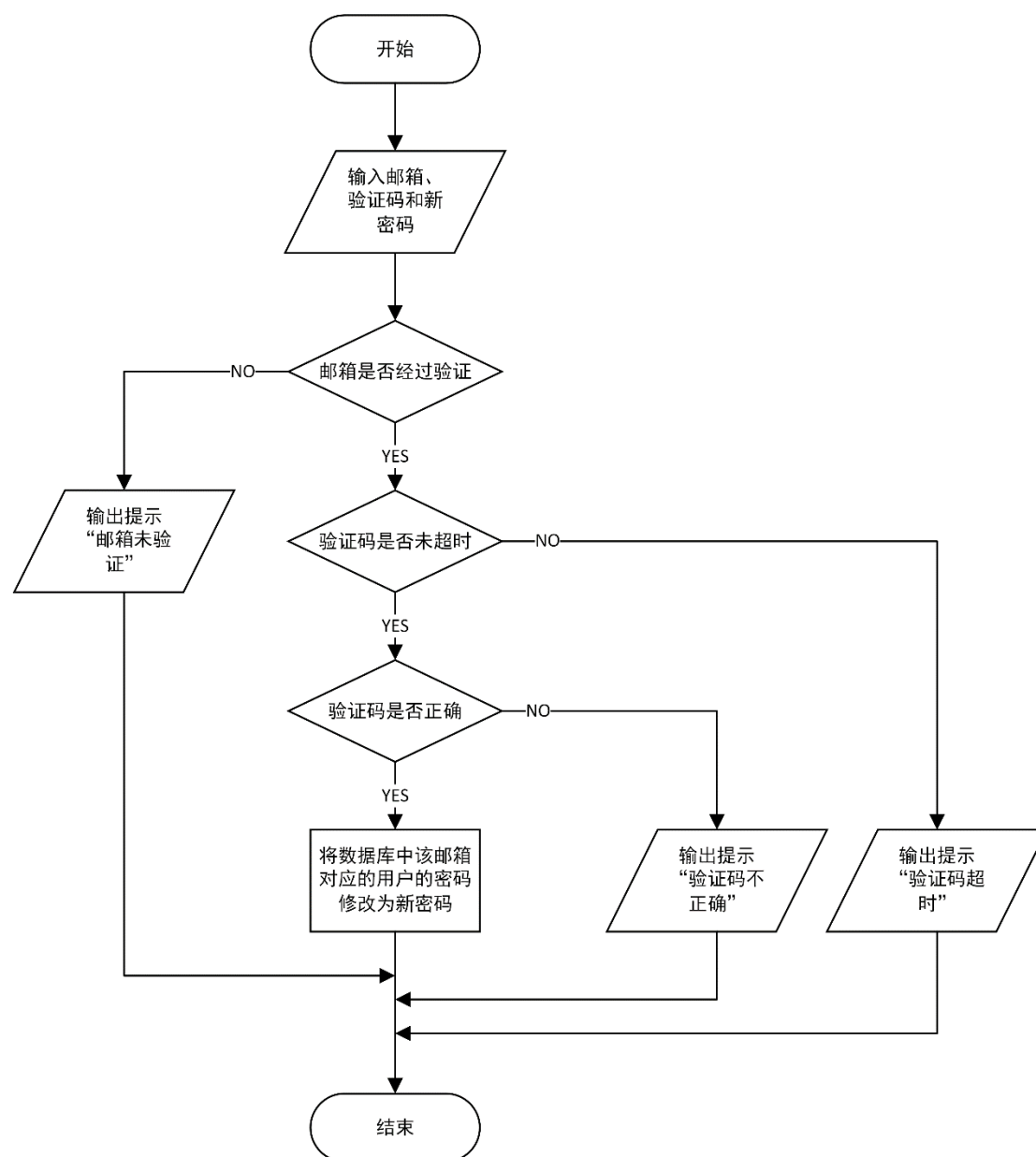
#### 1) 测试取消点赞失败

用例名称	测试取消点赞失败		
测试目的	测试用户已点赞，对帖子进行取消点赞失败	覆盖范围	等价类 (2) (6)
测试预置条件： 任一帖子详情页面正常运行。数据库中存在用户账号 2019010101。用户已经对该帖子进行点赞。点赞表存在用户账号 2019010101 对该帖子的点赞记录。			
测试步骤： 1) 登陆账号，在主页点击任一帖子标题进入帖子详情界面  2) 点击  点赞按钮，对该篇帖子取消点赞			
期望结果： 1) 提示“取消点赞错误，点赞数不能小于 0！”			
实际结果： 1) 提示“取消点赞错误，点赞数不能小于 0！”			
说明:			

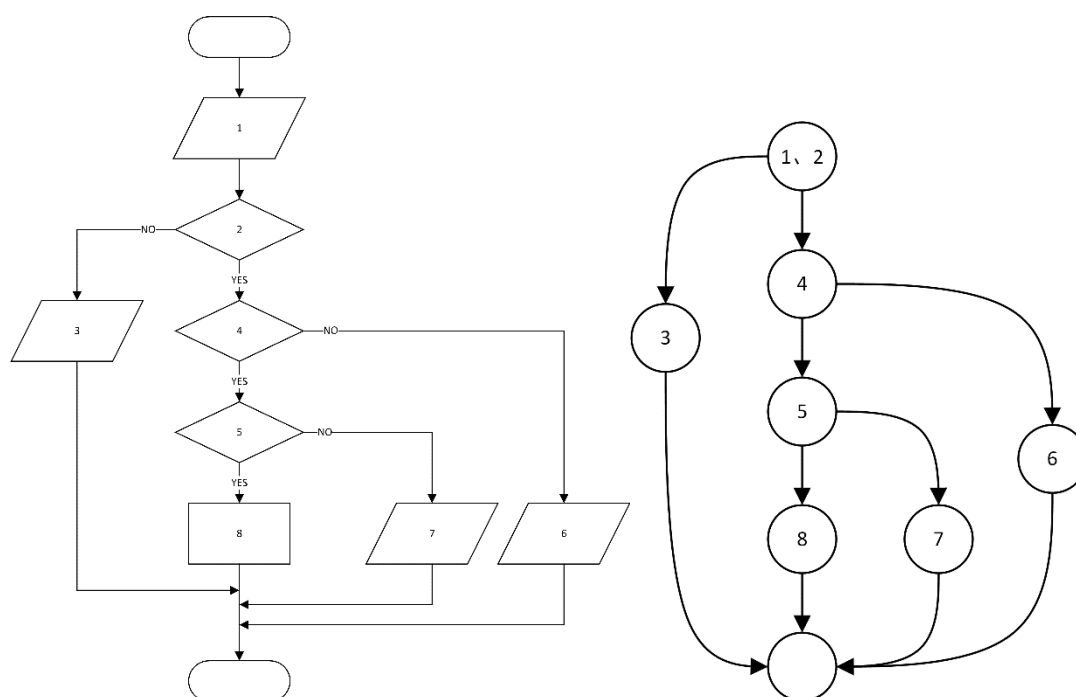
## 4 忘记密码模块

### 4.1 白盒测试

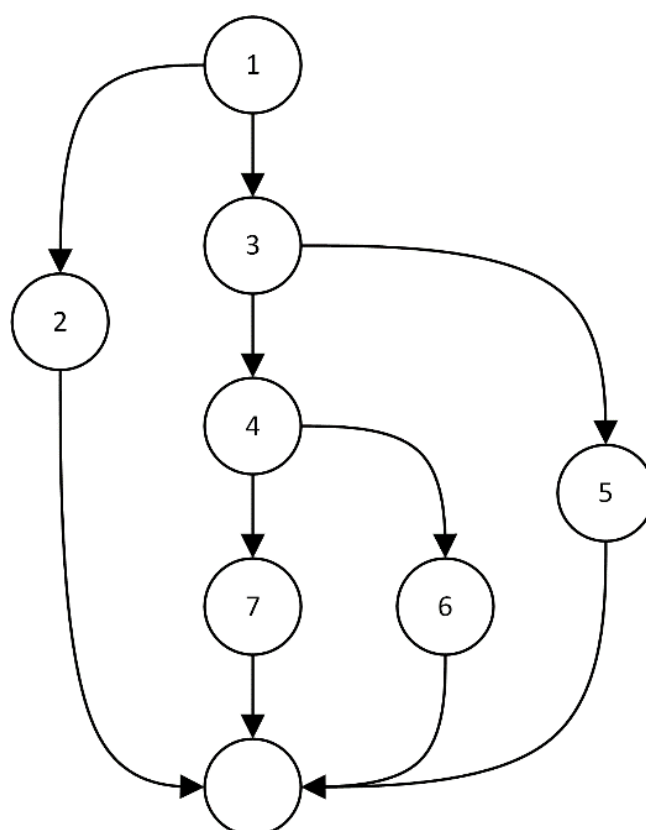
#### 4.1.1 程序流程图



### 4.1.2 流图



简化后流图如下：





### 4.1.3 流图的环形复杂度

流图中的判定节点数(1,3,4)，故环形复杂度为  $3+1=4$ 。

### 4.1.4 独立路径的基本组合

Path1: 1-2

Path2: 1-3-5

Path3: 1-3-4-6

Path4: 1-3-4-7

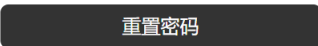
### 4.1.5 条件组合覆盖测试用例

$S = (\text{邮箱经过验证} \parallel \text{邮箱未经过验证}) \&\& (\text{验证码未超时} \parallel \text{验证码已超时}) \&\& (\text{验证码正确} \parallel \text{验证码错误})$

	邮箱经过验证 $\parallel$ 邮箱未经过验证	验证码未超时 $\parallel$ 验证码已超时	验证码正确 $\parallel$ 验证码错误	S
Path1	F	-	-	F
Path2	T	F	-	F
Path3	T	T	F	F
Path4	T	T	T	T

独立路径的实例：

#### (1) 测试重置密码失败 1

用例名称	测试重置密码失败 1		
测试目的	测试邮箱未经过验证	覆盖路径	Path1
测试预置条件： 忘记密码界面正常运行。使用的邮箱为已注册邮箱。			
测试步骤： 1) 进入忘记密码界面，输入已注册邮箱：1378113965@qq.com，输入验证码：123456，输入新密码：a123456789，输入确认密码：a123456789  2) 点击  按钮，进行密码重置			

期望结果:



1) 用户重置密码失败, 提示“请先验证邮箱”, 并停留忘记密码界面。

实际结果:

1) 用户重置密码失败, 提示“请先验证邮箱”, 并停留忘记密码界面。

说明:

## (2) 测试重置密码失败 2

用例名称	测试重置密码失败 2		
测试目的	测试验证码超时	覆盖路径	Path2
测试预置条件: 忘记密码界面正常运行。使用的邮箱为已注册邮箱。			
测试步骤:  1) 进入忘记密码界面, 输入已注册邮箱: 1378113965@qq.com, 点击  按钮, 等待 5min 后, 输入邮箱收到的正确验证码, 输入新密码: a123456789, 输入确认密码: a123456789  2) 点击  按钮, 进行密码重置			
期望结果: 1) 用户重置密码失败, 提示“验证码已过期”, 并停留忘记密码界面			
实际结果: 1) 用户重置密码失败, 提示“验证码已过期”, 并停留忘记密码界面			
说明:			

## (3) 测试重置密码失败 3

用例名称	测试重置密码失败 3		
测试目的	测试验证码错误	覆盖路径	Path3
测试预置条件: 忘记密码界面正常运行。使用的邮箱为已注册邮箱。			

测试步骤:

- 1) 进入忘记密码界面, 输入已注册邮箱: 1378113965@qq.com, 点击

点击发送验证码

按钮, 在 5min 内, 输入与邮箱收到的不一致的验证码, 输入新密码: a123456789, 输入确认密码: a123456789

- 2) 点击  按钮, 进行密码重置

期望结果:


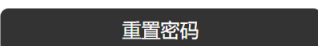
- 1) 用户重置密码失败, 提示“验证码错误”, 并停留忘记密码界面

实际结果:

- 1) 用户重置密码失败, 提示“验证码错误”, 并停留忘记密码界面

说明:

#### (4) 测试重置密码成功

用例名称	测试重置密码成功		
测试目的	测试重置密码成功	覆盖路径	Path4
测试预置条件: 忘记密码界面正常运行。使用的邮箱为已注册邮箱。			
测试步骤:  1) 进入忘记密码界面, 输入已注册邮箱: 1378113965@qq.com, 点击  按钮, 在 5min 内, 输入邮箱收到的正确验证码, 输入新密码: a123456789, 输入确认密码: a123456789  2) 点击  按钮, 进行密码重置			
期望结果: 1) 用户重置密码成功, 提示“重置密码成功”, 并跳转到登录界面			
实际结果: 1) 用户重置密码成功, 提示“重置密码成功”, 并跳转到登录界面			
说明:			



## 4.2 黑盒测试

### 4.2.1 等价类划分

输入等价类	有效等价类
邮箱	邮箱已验证（1）
验证码	验证码正确（2）
	验证码未超时（3）

### 4.2.2 有效等价类测试用例

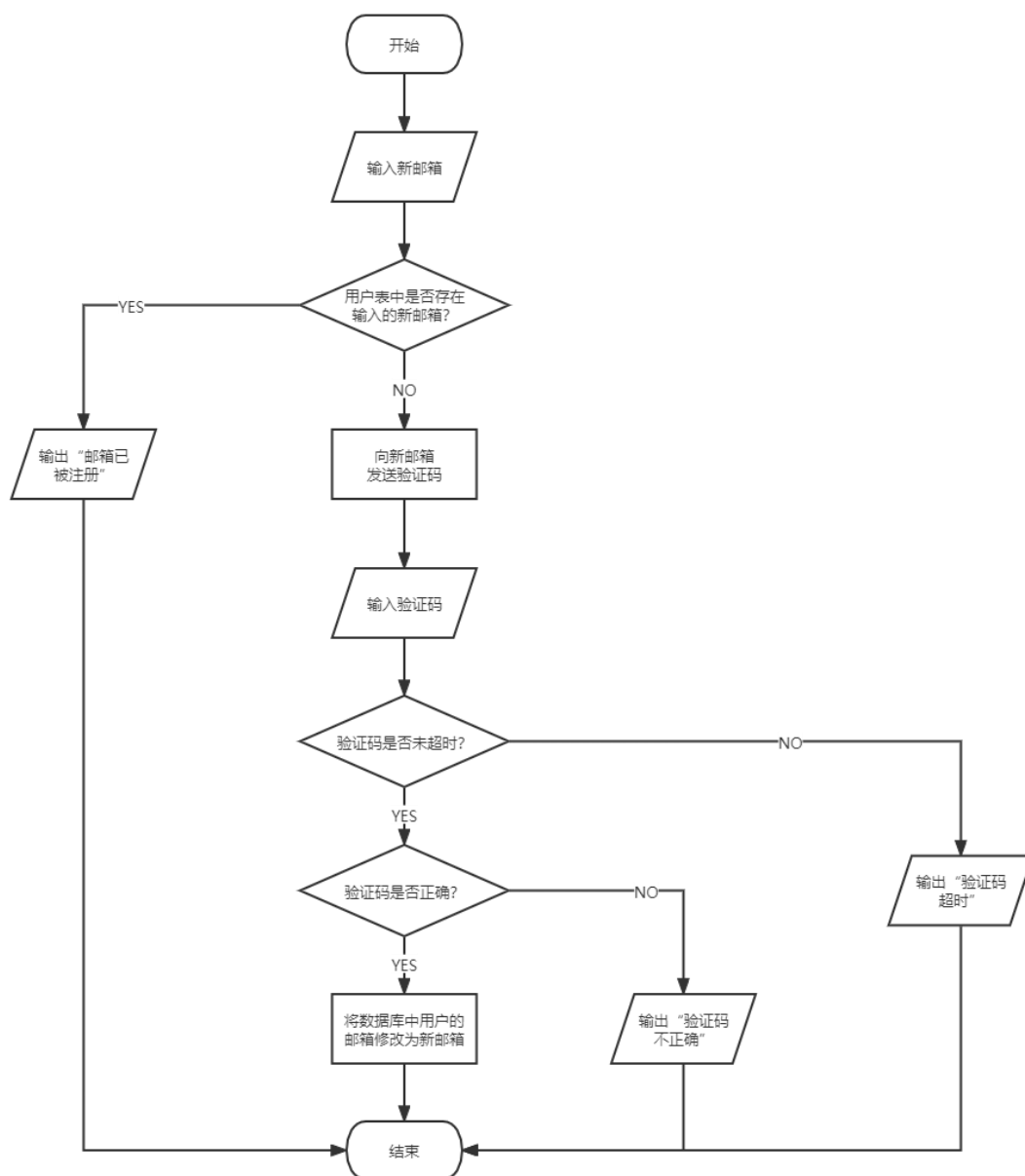
#### (1) 测试重置密码成功

用例名称	测试重置密码成功		
测试目的	测试重置密码成功	覆盖范围	等价类(1)、(2)、(3)
测试预置条件： 忘记密码界面正常运行。使用的邮箱为已注册邮箱。			
测试步骤：  1) 进入忘记密码界面，输入已注册邮箱：1378113965@qq.com，点击  按钮，在 5min 内，输入邮箱收到的正确验证码，输入新密码： a123456789，输入确认密码：a123456789  2) 点击  按钮，进行密码重置			
期望结果： 1) 用户重置密码成功，提示“重置密码成功”，并跳转到登录界面			
实际结果： 1) 用户重置密码成功，提示“重置密码成功”，并跳转到登录界面			
说明：			

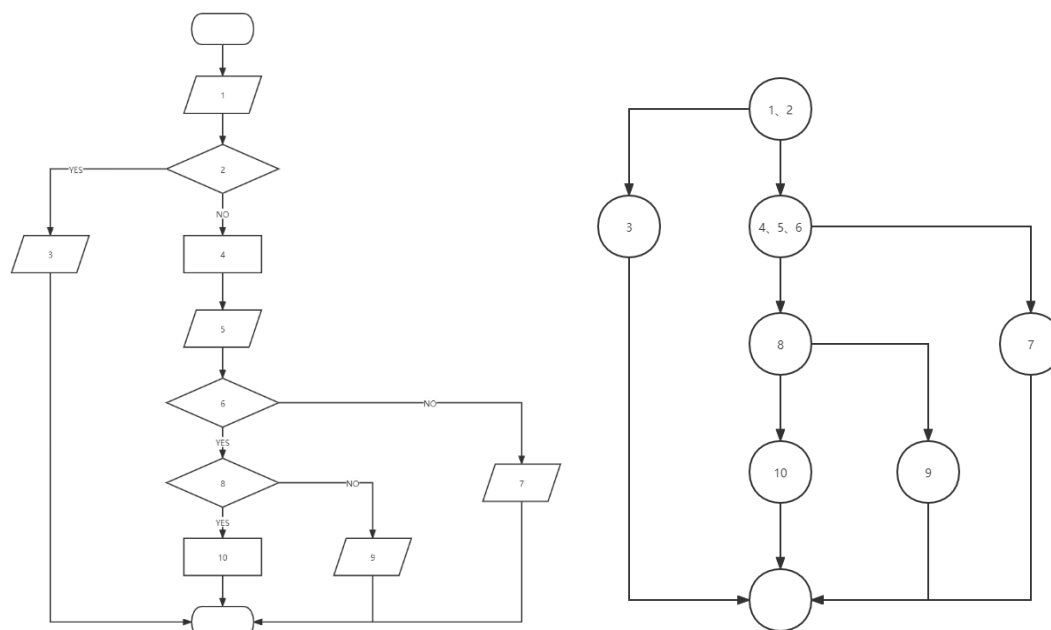
## 5 修改邮箱模块

### 5.1 白盒测试

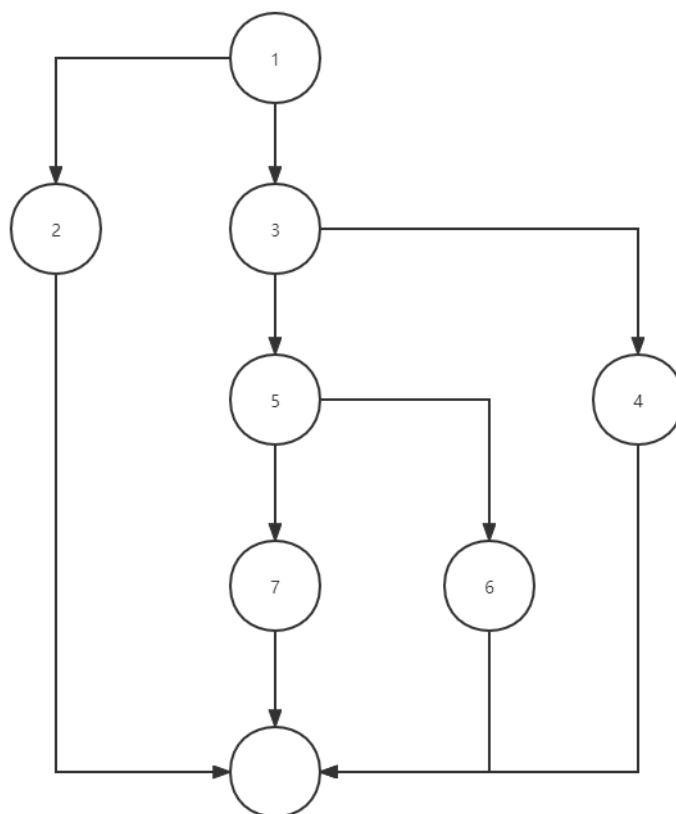
#### 5.1.1 程序流程图：



## 5.1.2 流图



简化后流图如下：



5.1.3 流图的环形复杂度

流图中的区域数等于环形复杂度，区域数为 4，故环形复杂度为 4。

5.1.4 独立路径的基本组合

Path1: 1-2

Path2: 1-3-4

Path3: 1-3-5-6

Path4: 1-3-5-7


5.1.5 条件组合覆盖测试用例

S = (邮箱不存在 || 邮箱存在) && (验证码未超时 || 验证码超时)&& (验证码正确 || 验证码错误)

	邮箱不存在    邮箱存在	验证码未超时    验证码超时	验证码正确    验证码错误	S
Path1	F	-	-	F
Path2	T	F	-	F
Path3	T	T	F	F
Path4	T	T	T	T



独立路径的实例：

(1) 测试修改邮箱失败 1

用例名称	测试修改邮箱失败 1		
测试目的	测试邮箱已被注册	覆盖路径	Path1
测试预置条件： 修改邮箱界面正常运行。数据库中存在用户邮箱为 2649070722@qq. com			
测试步骤： 1) 进入修改邮箱界面，输入用户的邮箱 2649070722@qq. com  2) 点击  按钮，向该邮箱发送验证码			

期望结果: 1) 修改邮箱失败, 提示“邮箱已被注册”, 并停留在修改邮箱界面
实际结果: 1) 修改邮箱失败, 提示“邮箱已被注册”, 并停留在修改邮箱界面
说明:



## (2) 测试修改邮箱失败 2

用例名称	测试修改邮箱失败 2		
测试目的	测试邮箱未注册但验证码超时	覆盖路径	Path2
测试预置条件: 修改邮箱界面正常运行。数据库中不存在用户邮箱为 3381977492@qq.com			
测试步骤: 1) 进入修改邮箱界面, 输入用户的邮箱 3381977492@qq.com  2) 点击  按钮, 向该邮箱发送验证码 3) 等待 5 分钟后输入邮箱收到的六位验证码  4) 点击  按钮, 确认修改邮箱			
期望结果: 1) 修改邮箱失败, 提示“验证码超时”, 并停留在修改邮箱界面			
实际结果: 1) 修改邮箱失败, 提示“验证码超时”, 并停留在修改邮箱界面			
说明:			



## (3) 测试修改邮箱失败 3

用例名称	测试修改邮箱失败 3		
测试目的	测试邮箱未注册且验证码未超时, 但验证码错误	覆盖路径	Path3
测试预置条件: 修改邮箱界面正常运行。数据库中不存在用户邮箱为 3381977492@qq.com			



测试步骤: 1) 进入修改邮箱界面，输入用户的邮箱 3381977492@qq.com  2) 点击  按钮，向该邮箱发送验证码 3) 在 5 分钟内输入与邮箱收到的六位验证码不同的六位字符  4) 点击  按钮，确认修改邮箱
期望结果: 1) 修改邮箱失败，提示“验证码错误”，并停留在修改邮箱界面
实际结果: 1) 修改邮箱失败，提示“验证码错误”，并停留在修改邮箱界面
说明:

(4) 测试修改邮箱成功

用例名称	测试修改邮箱成功		
测试目的	测试邮箱未注册且验证码正确未超时	覆盖路径	Path4
测试预置条件: 修改邮箱界面正常运行。数据库中不存在用户邮箱为 3381977492@qq.com			
测试步骤: 1) 进入修改邮箱界面，输入用户的邮箱 3381977492@qq.com  2) 点击  按钮，向该邮箱发送验证码 3) 在 5 分钟内输入邮箱收到的六位验证码  4) 点击  按钮，确认修改邮箱			
期望结果: 1) 修改邮箱成功，提示“修改邮箱成功”，返回个人信息界面且邮箱已更新			
实际结果: 1) 修改邮箱成功，提示“修改邮箱成功”，返回个人信息界面且邮箱已更新			
说明:			



## 5.2 黑盒测试

### 5.2.1 有效等价类划分

输入等价类	有效等价类
邮箱	用户表中邮箱不存在（1）
验证码	验证码未超时（2）
	验证码正确（3）

### 5.2.2 有效等价类测试用例

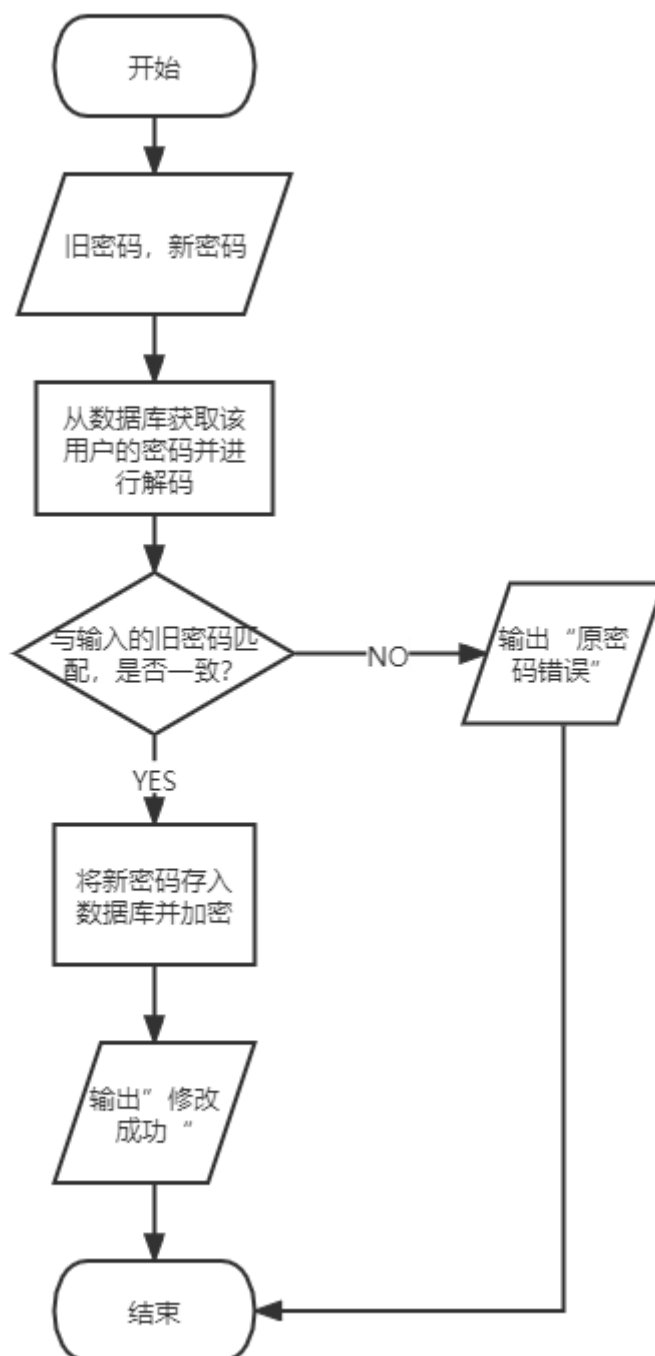
#### 1) 测试修改邮箱成功

用例名称	测试修改邮箱成功		
测试目的	测试邮箱未注册且验证码正确未超时	覆盖路径	等价类(1)、(2)、(3)
测试预置条件： 修改邮箱界面正常运行。数据库中不存在用户邮箱为 3381977492@qq. com			
测试步骤： 1) 进入修改邮箱界面，输入用户的邮箱 3381977492@qq. com  2) 点击  按钮，向该邮箱发送验证码 3) 在 5 分钟内输入邮箱收到的六位验证码  4) 点击  按钮，确认修改邮箱			
期望结果： 1) 修改邮箱成功，提示“修改邮箱成功”，返回个人信息界面且邮箱已更新			
实际结果： 1) 修改邮箱成功，提示“修改邮箱成功”，返回个人信息界面且邮箱已更新			

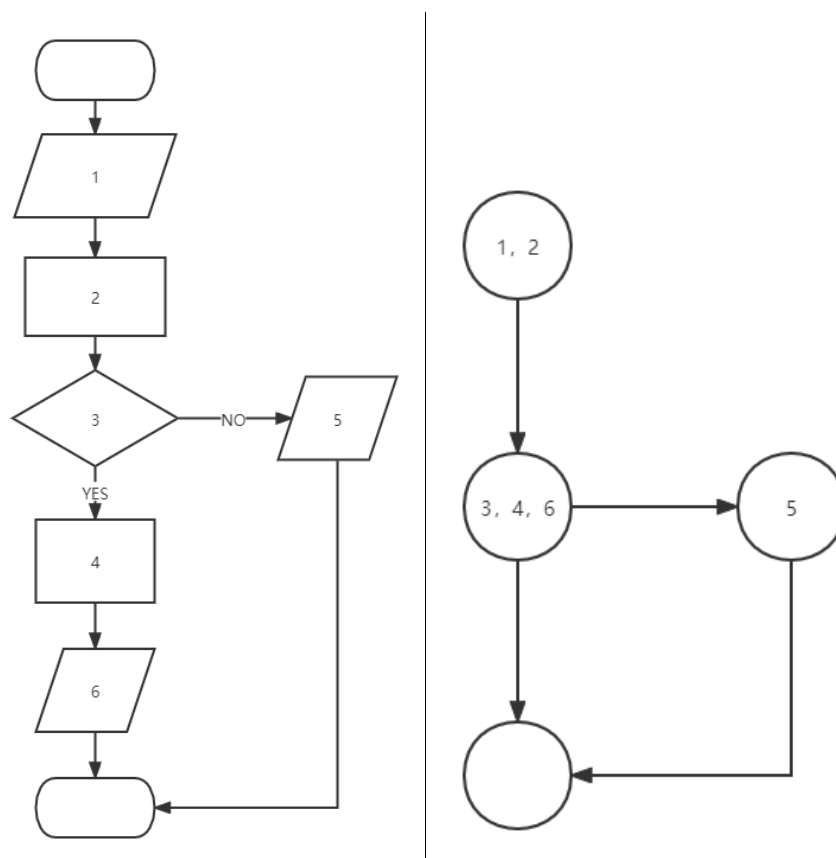
## 6 修改密码模块

### 6.1 白盒测试

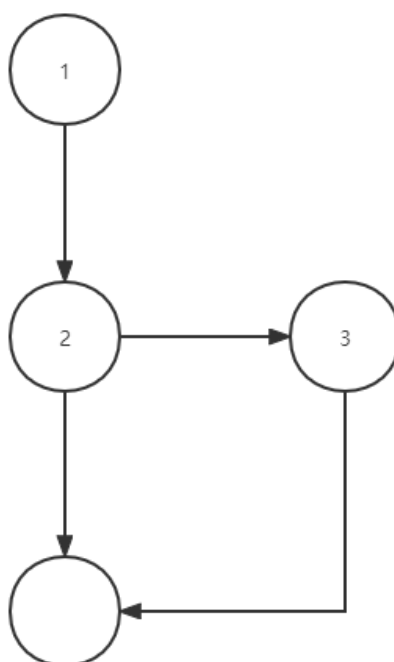
#### 6.1.1 程序流程图



### 6.1.2 流图



简化后流图如下：



### 6.1.3 流图的环形复杂度

流图中的判定节点数 1，故环形复杂度为  $1+1=2$ 。

### 6.1.4 独立路径的基本组合

Path1: 1-2

Path2: 1-2-3


### 6.1.5 条件组合覆盖测试用例

$S = (\text{原密码正确} \parallel \text{原密码不正确})$


	原密码正确 $\parallel$ 原密码不正确	S
Path1	T	T
Path2	F	F

独立路径的实例：

(1) 测试修改密码成功

用例名称	测试修改密码		
测试目的	测试修改成功	覆盖路径	Path1
测试预置条件： 修改密码界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤： 1) 进入修改密码界面，输入原密码 123456789 和新密码 111111111，并确认密码 111111111  2) 点击  按钮，进行修改密码			
期望结果： 1) 用户正常修改密码，提示“修改成功”			
实际结果： 1) 用户正常修改密码，提示“修改成功”			
说明：			

## (2) 测试修改密码失败

用例名称	测试修改密码失败		
测试目的	测试原密码错误，修改失败	覆盖路径	Path2
测试预置条件： 修改密码界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤： 1) 进入修改密码界面，输入原密码 111111112 和新密码 111111111，并确认密码 111111111  2) 点击  按钮，进行修改密码			
期望结果： 1) 用户修改密码失败，提示“原密码错误”，并停留在修改密码界面			
实际结果： 1) 用户修改密码失败，提示“原密码错误”，并停留在修改密码界面			
说明：			


## 6.2 黑盒测试

### 6.2.1 有效等价类划分

输入等价类	有效等价类
原密码	用户表中原密码正确（1）

### 6.2.2 有效等价类测试用例

测试修改密码成功

用例名称	测试修改密码成功		
测试目的	测试用户表中旧密码正确	覆盖范围	等价类(1)
测试预置条件： 修改密码界面正常运行。数据库中存在用户学号 2019010101, 且密码为 123456789			
测试步骤： 1) 进入修改密码界面，输入原密码 123456789 和新密码 111111111，并确认密码 111111111  2) 点击  按钮，进行修改密码			
期望结果： 1) 用户正常修改密码，提示“修改成功”			
实际结果： 1) 用户正常修改密码，提示“修改成功”			