# v模型

开发和测试阶段划分比较清晰；

需求分析、概要设计、详细设计、编码、单元测试（独立的模块测试）、集成测试（模块联调）、系统测试（整体流程）、验收测试（验证是否满足需求）

优点：包含了底层测试（单元测试）和高层测试（系统测试）；阶段划分清晰，方便工作的整体把控

缺点：测试阶段比较靠后，之前的的问题已经产生，修改不方便；v模型就是瀑布模型的变种，如果需求发生变化，必然要返工！

# w模型（双v模型）

开发一个v，测试一个v，开发和测试并行；

1. 开发v（需求分析、概要设计、详细设计、编码、集成、实施、交付）
2. 测试v（系统测试设计、集成测试设计、单元测试设计、单元测试、集成测试、系统测试、验收测试）

优点：开发伴随着测试并行，需求和设计一样要进行测试；尽早的介入测试，会更早的发现问题，降低修复成本；阶段依然明显，方便整体流程把控

缺点：代码依然在测试之前，不方便代码的测试工作；如果没有文档，根本无法进行w模型；对于人员要求较高！

# 三、软件测试分类—按阶段分类

1、单元测试（模块测试）

2、集成测试（组装测试）

3、系统测试（功能、性能、兼容性）

# 四、分类—是否查看源代码

1、白盒测试（查看源代码结构）

2、黑盒测试（不看源代码）

（1）功能测试：逻辑、界面、易用性、安装、兼容性

（2）性能测试：一般性能、稳定性、负载、压力

# 五、分类—是否运行程序

1、静态测试（看文档、看代码、不运行程序）

2、动态测试（实际运行代码）

# 六、按照自动化分类

1、人工测试 2、自动化测试

七、其它分类  
 1、回归测试

2、冒烟测试

3、随机测试（测试重点模块或之前出过问题的模块）

4、验收测试：

（1）内测版本（alpha），内部人员测试，或者有很少一部分用户；此阶段要解决严重的问题

（2）公测版本（beta），所有用户都可以免费使用，通过用户的反馈修复软件的细节

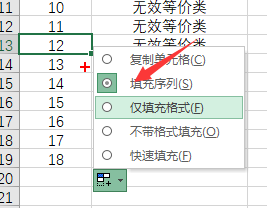
（3）准正式版（gamma），跟正式版几乎一样

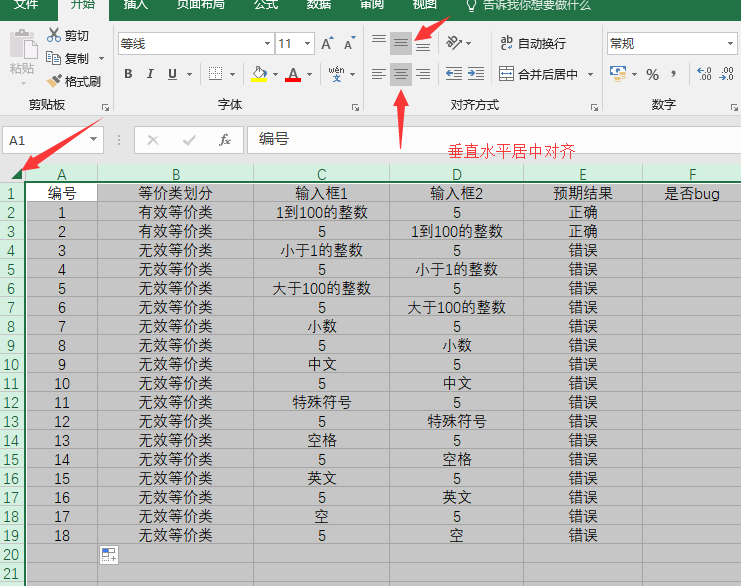
# 八、等价类划分法

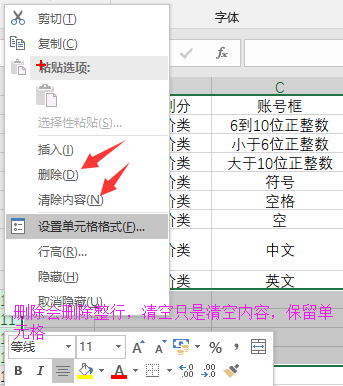
把无法穷举的数据分类书写

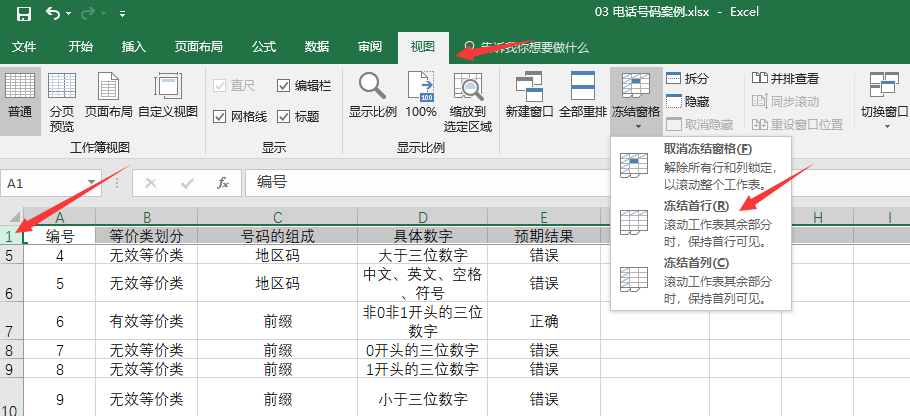
步骤:

1. 按照需求写出有效等价类
2. 根据需求取反，写出一部分无效等价类
3. 找到特殊情况的无效等价类（中文、英文、空格、空、符号、小数； 根据需求来选择）









## 九、等价类划分总结

1、长度

2、组合

3、类型

4、是否为空

5、是否区分大小写（根据需求来定，一般大小写都会区分）

6、空格问题（程序中文字的最前和最后部分的空格，程序员喜欢清除，文字中间的空格会保留）