软件测试有许多种方法，其中黑盒测试是广泛使用的两类测试方法之一。

**“黑盒”测的是功能**

**黑盒测试也称功能测试或数据驱动测试。**它在已知产品应具有的功能的条件下，通过测试来检测每个功能是否都能正常使用。

在测试时，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，测试者在程序接口进行测试，它只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数据而产生正确的输出信息，并且保持外部信息（如数据库或文件）的完整性。

**“黑盒”法着眼于程序外部结构、不考虑内部逻辑结构、针对软件界面和软件功能进行测试。**“黑盒”法是穷举输入测试，只有把所有可能的输入都作为测试情况使用，才能以这种方法查出程序中所有的错误。

实际上测试情况有无穷多个，人们不仅要测试所有合法的输入，而且还要对那些不合法但是可能的输入进行测试。

**“黑盒”的两种基本方法**

黑盒测试有两种基本方法，即通过测试和失败测试。

在进行通过测试时，实际上是确认软件能做什么，而不会去考验其能力如何。软件测试员只运用最简单，最直观的测试案例。

在设计和执行测试案例时，总是先要进行通过测试。在进行破坏性试验之前，看一看软件基本功能是否能够实现。这一点很重要，否则在正常使用软件时就会奇怪地发现，为什么会有那么多的软件缺陷出现？

在确信了软件正确运行之后，就可以采取各种手段通过搞“垮”软件来找出缺陷。纯粹为了破坏软件而设计和执行的测试案例，被称为失败测试或迫使出错测试。

 **黑盒测试的设计方法**

黑盒测试是以用户的观点，从输入数据与输出数据的对应关系出发进行测试的，它不涉及到程序的内部结构。

很明显，如果外部特性本身有问题或规格说明的规定有误，用黑盒测试方法是发现不了的。

黑盒测试法注重于测试软件的功能需求，主要试图发现几类错误：功能不对或遗漏、界面错误、数据结构或外部数据库访问错误、性能错误、初始化和终止错误。

具体的黑盒测试方法包括等价类划分、因果图、正交实验设计法、边值分析、判定表驱动法、功能测试等。在使用时，自然要针对开发项目的特点对方法加以适当的选择。

**◆ 等价类划分**

等价类划分是一种典型的黑盒测试方法，用这一方法设计测试用例可以不用考虑程序的内部结构，只以对程序的要求和说明，即需求规格说明书为依据，仔细分析和推敲说明书的各项需求，特别是功能需求，把说明中对输入的要求和输出的要求区别开来并加以分解。

由于穷举测试的数量太大，以致于无法实际完成，促使我们在大量的可能数据中选取其中的一部分作为测试用例。

例如，在不了解等价分配技术的前提下，测试了1+1、1+2、1+3和1+4之后，还有必要测试1+5和1+6吗?能否放心地认为它们正确吗?那么1+999…（可以输入的最大数值）呢？这个测试用例是否与其他用例不同？是否属于另外一种类别？另外一个等价区间？这是软件测试员必须考虑到的问题。

等价类别或者等价区间是指测试相同目标或者暴露相同软件缺陷的一组测试案例。 1+999…和1+13有什么区别呢？

至于1+13，就像一个普通的加法，与1+5或者1+392没有什么两样，而1+999…则属于邻界的极端情况。假如输入最大允许数值，然后加1，就会出现问题——也许就是软件的缺陷。这个极端案例属于一个单独的区间，与常规数字的普通区间不同。

等价类划分的办法是把程序的输入域划分成若干部分，然后从每个部分中选取少数代表性数据当作测试用例。

每一类的代表性数据在测试中的作用等价于这一类中的其他值，也就是说，如果某一类中的一个例子发现了错误，这一等价类中的其他例子也能出现同样的错误。使用这一方法设计测试用例，首先必须在分析需求规格说明的基础上划分等价类，列出等价类表。

在考虑等价类划分时，先从程序的功能说明中找出每个输入条件，然后为每个输入条件划分两个或更多个等价类。

等价类可分两种情况：有效等价类和无效等价类。有效等价类是指对程序的规格说明是有意义的、合理的输人数据所构成的集合；无效等价类是指对程序的规格说明是不合理的或无意义的输人数据所构成的集合。

**◆ 边界值分析**

软件测试常用的一个方法是把测试工作按同样的形式划分。对数据进行软件测试，就是检查用户输入的信息、返回结果以及中间计算结果是否正确。

即使是最简单的程序，要处理的数据也可能数量极大。还记得在计算器上简单加法的全部可能性吗？再想一想字处理程序、导航系统和证券交易程序。使这些数据得以测试的技巧（如果称得上的话）是，根据下列主要原则进行等价分配，以合理的方式减少测试案列：边界条件、次边界条件、空值和无效数据。

边界值分析（Boundary Value Analysis，BVA）是一种补充等价划分的测试用例设计技术，它不是选择等价类的任意元素，而是选择等价类边界的测试用例。实践证明，在设计测试用例时，对边界附近的处理必须给予足够的重视，为检验边界附近的处理专门设计测试用例，常常可以取得良好的测试效果。BVA不仅重视输人条件边界，而且也从输出域导出测试用例。

**边界值设计测试遵循的五条原则：**

1、如果输入条件规定了取值范围，应以该范围的边界内及刚刚超范围边界外的值作为测试用例。如以a和b为边界，测试用例应当包含a和b及略大于a和略小于b的值；

2、若规定了值的个数，分别以最大、最小个数及稍小于最小、稍大于最大个数作为测试用例；

3、针对每个输出条件使用上述1、2条原则；

4、如果程序规格说明中提到的输入或输出域是个有序的集合（如顺序文件、表格等），就应注意选取有序集的第一个和最后一个元素作为测试用例；

5、分析规格说明，找出其他的可能边界条件。