**2.**

**黑盒测试：针对课本P293,求一元二次方程根的程序，仅仅设计针对A、B、C的“正数、零、负数、非数值、舍入误差(精度）”的测试用例，充分吗？ 还需要哪些测试用例？参考P302**

仅考虑A、B、C的“正数、零、负数、非数值、舍入误差(精度）”的测试用例不够充分，还应考虑如下的几种情况

1. 当A为0时，该方程不满足一元二次方程的定义。
2. 当判别式小于0时，该方程无实数解，需要根据具体要求输出无解或输出复数解
3. 当A、B、C较大时，运算过程中可能产生溢出

**白盒测试：针对课本P304图8-7的的所有路径测试有多少条？所有路径都测试通过了，能保证程序不会出错？**

一共有4条

1-2-3-4-5-6-7

1-2-4-5-6-7

1-2-3-4-5-6-1

1-2-4-5-6-1

所有路径都测试通过了，也不能保证程序不会出错。因为一方面上面的测试路径不能够包含所有可能出现的情况，在其他没有被测试的路径中可能会有错误；另一方面，即使所有路径都能够测试通过也可能会存在当特殊情况发生时的问题。

**3.**

**课本P324 习题7, 给出所描述的6种方法对构件进行集成测试的顺序。**

**（1）自底向上方法：**

①测试L,G,H,M,N,J,K,E

②测试（F,L）和（I,M,N）

③测试（B,F,L,G）和（C,H）和（D,I,J,K,M,N）

④测试所有全部构件合在一起的整个程序（A,B,C,D,E,F,G,H,I,J,K,L,M,N）

**（2）自顶向下方法：**

①测试A

②测试（A,B,C,D,E）

③测试（A,B,C,D,E,F,G,H,I,J,K）

④测试所有全部构件合在一起的整个程序（A,B,C,D,E,F,G,H,I,J,K,L,M,N）

**（3）改进的自顶向下方法：**

①测试A

②测试B,C,D,E

③测试（A,B,C,D,E）

④测试F,G,H,I,J,K

⑤测试（A,B,C,D,E,F,G,H,I,J,K）

⑥测试L,M,N

⑦测试所有全部构件合在一起的整个程序（A,B,C,D,E,F,G,H,I,J,K,L,M,N）

**（4）一次性集成方法：**

①测试A,B,C,D,E,F,G,H,I,J,K,L,M,N

②测试所有全部构件合在一起的整个程序（A,B,C,D,E,F,G,H,I,J,K,L,M,N）

**（5）三明治方法：**

①测试A,L,M,N

②测试（A,B），（A,C），（A,D），（A,E），（F,L），（I,M,N）

③测试（B,F,G），（C,H），（D,I,J,K）

④测试所有全部构件合在一起的整个程序（A,B,C,D,E,F,G,H,I,J,K,L,M,N）

**（6）改进的三明治方法：**

①测试A,B,C,D,E,F,I,L,M,N

②测试G,H,J,K，（A,B），（A,C），（A,D），（A,E），（F,L），（I,M,N）

③测试（B,F,G），（C,H），（D,I,J,K）

④测试所有全部构件合在一起的整个程序（A,B,C,D,E,F,G,H,I,J,K,L,M,N）