该测试程序测试重点在于以下几个方面：

1. 数组
2. 表达式
3. 程序递归调用
4. 分支和循环
5. 类型间的隐式转换

样例1

测试输入：10 a 2.7

测试输出：10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 a b c d e f g h i j k l m n o p q r s t 0.000000 -2.592592 -1.592592 -0.592592 0.407408 1.407408 2.407408 3.407408 4.407408 5.407408 6.407408 7.407408 8.407408 9.407408 10.407408 11.407408 12.407408 13.407408 14.407408 15.407408 6765 a b c d e f g h i j k l m n o p q r s t Implicit Transformation Passed

其中10, 11…29是\_global\_int\_arr数组中的元素;a,b…t是\_global\_char\_arr数组中的元素;0.000000, -2.592592…15.407408是\_global\_float\_arr数组中的内容。这三个数组内容的计算都与在init函数中从标准输入中读入的\_global\_int，\_global\_char和\_global\_float有关。以上是对于数组的测试，其中涉及到了不同类型间的表达式计算和隐式转换内容。

6765是第十个斐波那契数，由float average();函数计算出\_global\_int\_arr的平均值后传入int fib(int k);函数计算而得。其中average函数返回值是float，而fib函数的参数是int类型，所以也考虑到了隐式转换的测试；fib有递归调用，所以也是一个针对函数递归的测试。

接下来的输出：a,b…t则是对if分支和while循环的测试；最后输出的的Implicit Transformation Passed判断的是不同类型间的条件判断。

样例2

测试输入：20 A 2.7

测试输出：20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 A B C D E F G H I J K L M N O P Q R S T 0.000000 -0.203704 0.796296 1.796296 2.796296 3.796296 4.796296 5.796296 6.796296 7.796296 8.796296 9.796296 10.796296 11.796296 12.796296 13.796296 14.796296 15.796296 16.796297 17.796297 832040 A B C D E F G H I J K L M N O P Q R S T Implicit Transformation Passed

样例3

测试输入：-10 c 1.

测试输出：-10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 c d e f g h i j k l m n o p q r s t u v 0.000000 10.900000 11.900000 12.900000 13.900000 14.900000 15.900000 16.900000 17.900000 18.900000 19.900000 20.900000 21.900000 22.900000 23.900000 24.900000 25.900000 26.900000 27.900000 28.900000 1 What to do now?