```
using namespace std;
#include <iostream>
  void calcletter(double test, char & letter);
                                                              This function call passes test1 and letter1 to
                                                              the function calcletter, where it will be
 int main ()
                                                              known as test and letter.
  { double test1,test2,test3,average;
   char letter1, letter2, letter3, avgletter;
   cout << "Enter test 1 ";
                                                              This function call passes test2 and letter2 to
   cin >> test1;
                                                              the function calcletter, where it will be
   calcletter (test1,letter1);
                                                              known as test and letter.
   cout << "Enter test 2 " :
   cin >> test2:
                                                              This function call passes test3 and letter3 to
   calcletter (test2,letter2);
                                                              the function calcletter, where it will be
   cout << "Enter test 3 " :
                                                              known as test and letter.
   cin >> test3;
   calcletter (test3,letter3);
   average=(test1+test2+test3)/3.0;
   calcletter (average, avgletter); 	
                                                              This function call passes average and
                                                              avgletter to the function calcletter, where it
   cout << "Test \tTest\tLetter grade"<< endl<<endl;
                                                              will be known as test and letter.
   cout << "Test 1 \t"<< test1<<"\t "<<letter1<< endl;
   cout << "Test 2 \t"<< test2<<"\t "<<letter2<< endl;
   cout \ll "Test 3 \ \t" \ll test 3 \ll "\t" \ll endl \ll endl;
   cout << "Average \t"<< average<<"\t "<<avgletter<< endl;
   return 0;
  }
//---
  void calcletter(double test, char & letter)
                                                This function will only know that it has an in parameter known
                                                as test which is of type double, and has an out parameter letter
  \{ if (test >= 90) \}
                                                of type character. This technique allows programmers to avoid
     letter = 'A';
                                                typing the same code for test1, test2, test 3 and average.
   else if (test \geq 80)
     letter = 'B':
   else if (test \geq 70)
                             In parameter refers to the fact that the identifier has some value coming to the
     letter = 'C';
                             function which is meaningful and needed by the function. The function can make
   else if (test \geq 60)
                             changes to in parameters, but these changes only exist in the function and are not
     letter = 'D';
                             kept after the function completes it task.
   else
     letter = 'F':
                             Out parameter refers to the fact that an identifier usually returns a changed value
   return; }
                             to the calling location. The & is needed for this to occur in C++. The identifier
                             letter in this example will return a letter grade to the calling location. The
                             identifier letter had no meaningful value coming into the function.
```