1. Write a program that uses the **F( )** macro shown at the beginning of the chapter and demonstrates that it does not expand properly, as described in the text. Repair the macro and show that it works correctly.

2. Write a program that uses the **FLOOR ( )** macro shown at the beginning of the chapter. Show the conditions under which it does not work properly.

3. Create two identical functions, **f1 ( )** and **f2 ( )**. Inline **f1 ( )** and leave **f2 ( )** as an non-inline function. Use the Standard C Library function **clock( )** that is found in **<ctime>** to mark the starting point and ending points and compare the two functions to see which one is faster. You may need to make repeated calls to the functions inside your timing loop in order to get useful numbers.

4. Experiment with the size and complexity of the code inside the functions in Exercise 3 to see if you can find a break-even point where the inline function and the non-inline function take the same amount of time. If you have them available, try this with different compilers and note the differences.

5. Prove that inline functions default to internal linkage.

6. Create a class that contains an array of **char**. Add an inline constructor that uses the Standard C library function **memset ( )** to initialize the array to the constructor argument (default this to ‘ ’), and an inline member function called **print( )** to print out all the characters in the array.