CPS 171 Answers to Exam Review Questions for Test 2 that are not in the book

Chapter 5
3. `MadamNowI'mAdam`
5. a. `Eligible to serve.`
    b. `Too short and too light to serve.`
11. To fix the dangling `else` problem, insert braces as follows:

```
if (typeA || typeB)
{
    if (typeA && typeB)
        cout << "Type AB";
}
else
```

Switch
3. True
5.

```
switch (n)
{
    case 3 : alpha++;
             break;
    case 7 : beta++;
             break;
    case 10: gamma++;
             break;    // Optional
}
```

Loops
10. Nothing is printed. While and For are pretest loops. Here, the loop condition is immediately `false` and the body is skipped.
7.

```
sum = 0;
for (count = 1; count <= 1000; count++)
    sum = sum + count;
```

8.

```
m = 93;
while (m >= 5)
{
    cout << m << ' ' << m * m << endl;
    m--;
}
```

9.

```
k = 9;
do
{
    cout << k << ' ' << 3 * k << endl;
    k++;
} while (k <= 21);
```

Nested Loops
5. True
8. True
12. True
11. 4 3 2 1 4
   3 2 1 3
   2 1 2
   1 1

12.
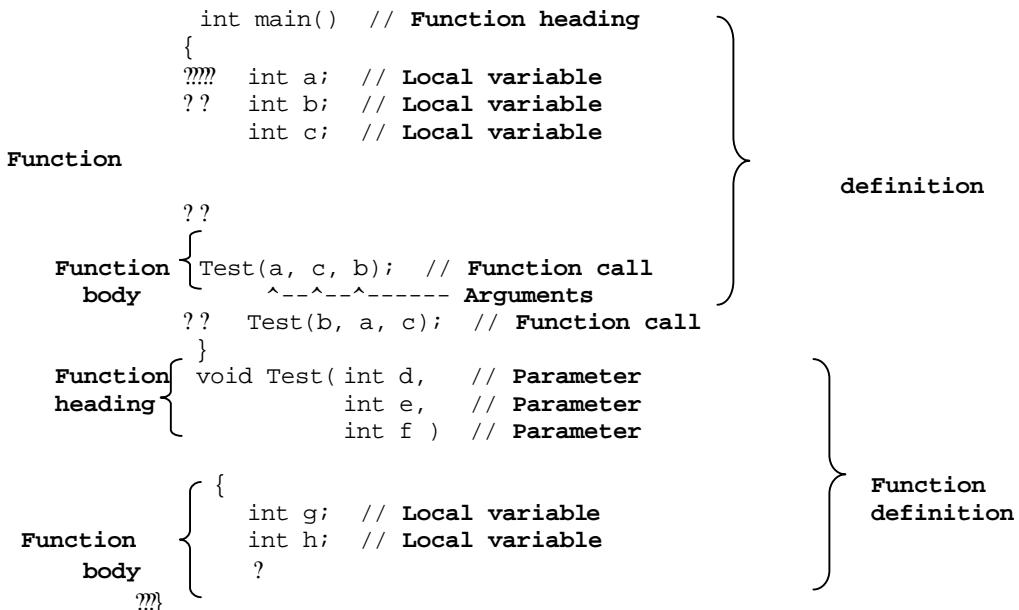
```
********* *********
*******   *******
*******     *******
******       ******
*****         *****
****           ****
***             ***
**               **
*                 *
```

Chapter 7

1.

Function call	The mechanism that transfers control to the body of a function
Parameter list	A mechanism by which functions communicate with each other; the list of variables and/or expressions following the name of a function that determines how the function and the calling code communicate with each other
Parameterless function	A function that has no parameters; a function whose parameter list is empty
Parameter	A variable declared in a function heading
Argument	A variable or expression listed in a call to a function
Local variable	A variable declared within a block and not accessible outside of that block

2.

```
        void Test( int, int, int );  // Function prototype

            int main()  // Function heading
           {
       ????  int a;  // Local variable
       ??    int b;  // Local variable
             int c;  // Local variable
   Function
                                                                    definition
           ??
   Function ┌ Test(a, c, b);  // Function call
     body   └      ^--^--^------ Arguments
           ??   Test(b, a, c);  // Function call
           }
   Function ┌ void Test(int d,   // Parameter
   heading  ┤           int e,   // Parameter
            └           int f )  // Parameter

            ┌ {
            │    int g;  // Local variable              Function
   Function ┤    int h;  // Local variable              definition
     body   │    ?
      ??}   └
```

3.

	First Call to Test			Second Call to Test	
	Formal	*Actual*		*Formal*	*Actual*
1.	d	a	1.	d	b
2.	e	c	2.	e	a
3.	f	b	3.	f	c

5. True

6. True

13.

```
   1      cout << "Exercise ";
   2      DoThis(number1, number2);
   5      cout << number1 << ' ' << number2 << endl;

   3      cin >> value3 >> value1;
   4      value2 = value1 + 10;
```

14. In the `main` function just before execution of the `return` statement, `number1` contains 15, `number2` contains 25, and `value3` does not exist (it is local to the `DoThis` function).