HOMEWORK:

1) Figure out what this method actually returns, and describe it, in detail, in terms of arg1 and

   arg2.

2) Rewrite this method so that it does exactly the same thing, but doesn't have a single local

   variable whose scope is the entire method (like "OtherMethod", but, you know, with the

   same functionality as this method).

EXTRA CREDIT:

1) Write your own "puzzle" method.  Make it as confusing per length as you can.

2) Share your "puzzle" method with the class (on the class forum), so that other people

   can try to figure out what it does, and how to rewrite it.

3) Bring your solution to class on Tuesday (what it does, and what a clean rewrite looks like).

**Solution**

1.  Analysis.  Test cases:      arg1 : arg1 is an element of {arg1< 17,  arg1 >=17};

arg2 : arg2 is an elelment of {arg2<=6, 6<arg2<=12,

12<arg2<=22, 22<arg2}

If arg2 is less than or equal to 6 and arg 1 is greater than or equal to 17, the method returns “”.

However, if arg1 is less than 17 and arg2 is less than or equal to 6, the the first if statement does not overwrite result, instead returning “”.

if arg2 is greater than 6, the function returns “arg1” + arg2 if this return value has length less than or equal to 5. Otherwise “Ooops” is returned.

2.

Public string SomeMethod(int arg1, int arg2)

{

if ((arg2 <= 6) )

{

return “”;

} else

{

return ((result = (“arg1” + arg2)).Length > 5)?   “Ooops…” : result;

}