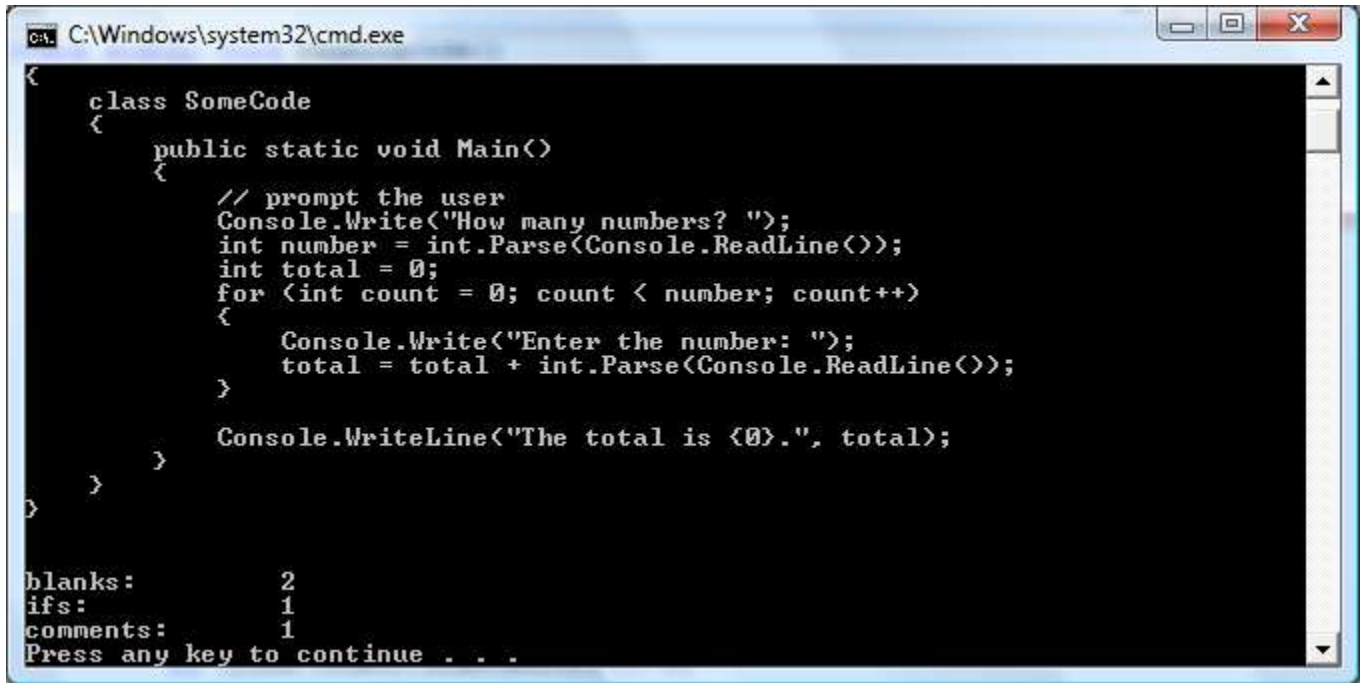


Assignment 3

1. **File I/O.** Write a program that reads in a .cs file. Print out the number of lines of comments, the number of for statements, and the number of blank lines. The [Trim](#) and [StartsWith](#) methods of the string class will be helpful.

Here's my output, where I also printed the file to the screen, which is useful for verifying the results:



```

C:\Windows\system32\cmd.exe
<
class SomeCode
{
    public static void Main()
    {
        // prompt the user
        Console.Write("How many numbers? ");
        int number = int.Parse(Console.ReadLine());
        int total = 0;
        for (int count = 0; count < number; count++)
        {
            Console.Write("Enter the number: ");
            total = total + int.Parse(Console.ReadLine());
        }
        Console.WriteLine("The total is {0}.", total);
    }
}

blanks:      2
ifs:         1
comments:    1
Press any key to continue . . .
  
```

2. **Arrays.** Rewrite the "Talk Like a Pirate" translator from assignment 1 to use arrays. There are *at least* two ways to do this.
 - Create two arrays. The first holds the English words, the second holds the pirate words. The 0th element of the English array corresponds to the 0th element of the pirate array, and so on for the 1st, 2nd, etc. elements. Use a for statement to iterate through the arrays and make replacements to the input string.
 - Create one 2-dimensional array. Use a for statement to iterate through the arrays and make replacements to the input string.
3. **Arrays and File I/O (Optional).** Make the "Talk Like a Pirate" translator extensible by reading in the phrases from a file. Hint: Put the English and pirate phrases one pair to a line, using a distinctive separator. For example:

Excuse me | arr

Sir | matey

Then use the [String.Split](#) method on each line to retrieve the phrases.