COEN 70: Formal Specification and Advanced Data Structures

Lab 1: Getting familiar with C++

1) Display the following pattern using C++ `string` and `setw()`

```
0123456789
 9876543210
  0123456789
   9876543210
    0123456789
     9876543210
      0123456789
       9876543210
        0123456789
         9876543210
```

2) Count the number of alphanumeric characters and non-alphanumeric characters from the standard input (`cin`). Spaces (‘ ‘) should not be counted toward either type of character. Here are some example outputs:

```
“Hello, World!” has 10 alphanumeric characters and 2 non-
alphanumeric characters.
"Formal Specification and Advanced Data Structures” has 44
alphanumeric characters and 0 non-alphanumeric characters.
```

3) Convert to uppercase and display all the words from an input file with length of at least 10 characters. All punctuation marks are removed and do not contribute to the words' length. The name of the file to be read should be read in as a common line argument. Here is the output with the Gettysburg Address as the input:

```
PROPOSITION
BATTLEFIELD
ALTOGETHER
CONSECRATE
CONSECRATED
UNFINISHED
GOVERNMENT
```

4) Fix the program so that the object B has the intended string (`Hello World`).

```
struct object {
      string* s;
};
object A, B;
A.s = new string;
*A.s = "Hello World"; // A has "Hello World"
B = A;                // B has "Hello World"
*A.s = "Goodbye";     // A has "Goodbye"
cout << *A.s << endl;
cout << *B.s << endl;
```