***Lab # 11 Working with Strings in C***

**Reading Task 1:**

**Working with Arrays:**

Strings are one-dimensional array of characters terminated by a null character '\0'. Thus, a null-terminated string contains the characters that comprise the string followed by a null.

The following declaration and initialization create a string consisting of the word "Hello". To hold the null character at the end of the array, the size of the character array containing the string is one more than the number of characters in the word "Hello."

**In-Lab Task 1:**

Write a C Program that does the following:

1. Declares a C-String called ‘m1’ and initializes it with text “Programming is great fun!”.

2. Uses C-function puts() to print this string.

3. Asks the user to enter a String named ‘m2’ (Hint: Use gets() function for this.)

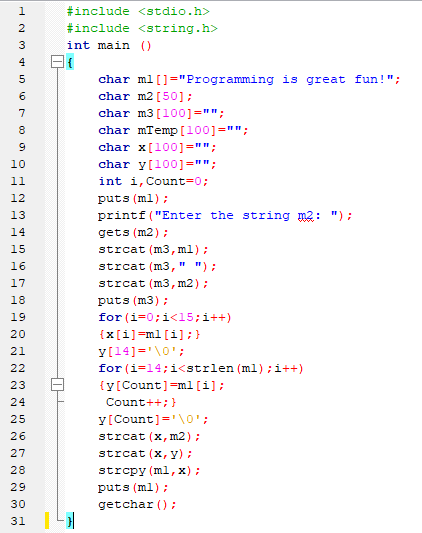
4. Concatenates the two strings and stores the result in ‘m3’.

For example if the user enters m2 as “Not Really!”, m3 should be “Programming is great fun! Not really!”

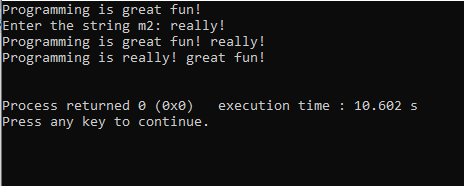
5. Inserts the user entered array (m2) into m1 after “Programming is ...”

For the above example, the resultant String would become “Programming is Not really! great fun!”

**Program:**

****

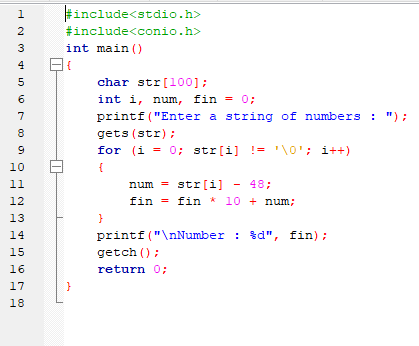
**Output:**

****

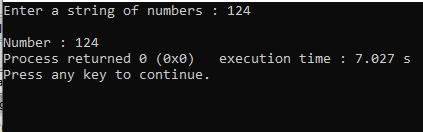
**In-Lab Task 2 a:**

Write a program that converts a string like "124" to an integer 124.

**Program:**

****

**Output:**

****

**In-Lab Task 2 b:**

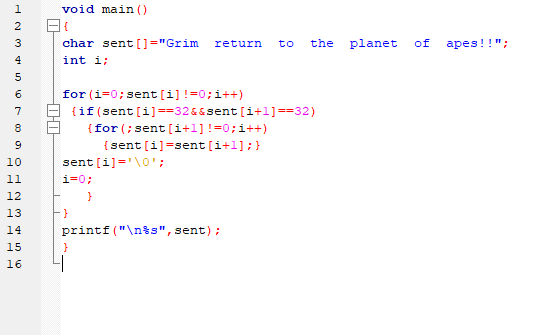
Write a program that replaces two or more consecutive blanks in a string by a single blank. For example, if the input is

“Grim return to the planet of apes!!”

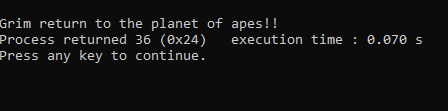
the output should be

“Grim return to the planet of apes!!”

**Program:**

****

**Output:**

****

**Critical Analysis:**