**A Simple Paragraph Justifier**

The mission of this assignment is to realize a simple paragraph justifier. As shown in my coding demonstration, your program will keep reading lines of text image using getline(). After reading a paragraph, it adjusts all the lines into an instructed width, so that it can show the words evenly spread out and fit in a straight edge at both margins (similar to "align full" option in a Microsoft word document).

This can be relatively easily done as we are dealing with a mono-space font set (not a proportional font set) shown on a command-line-type console

Assume we have five six letter words.  
Just the total of the letters of the words amounts to 30.  
If those six words are to be printed in a field width of 40.  
We have 10 spaces to distribute.  
Also assume that following the five six letter words is a word that has more than 9 characters. That is, the next word cannot be placed in the tail of that we are filling out. The demands of that word exceed the extra spaces.  
  
With tail justification:   
2 spaces are inserted between word1 and word2,  
2 spaces are inserted between word2 and word3,  
2 spaces are inserted between word3 and word4,  
and 4 spaces are inserted at the tail between word4 and word5.  
  
With even justification, fewer spaces are leftover for the slot between the last two words.  
More spaces are used earlier in the line.  
So the ten spaces will be distributed as follows:  
3 spaces are inserted between word1 and word2,  
3 spaces are inserted between word2 and word3,  
2 spaces are inserted between word3 and word4,  
and 2 spaces are inserted between word4 and word5.

The requirements of this assignment are:

1. program keeps reading lines of text until reading an empty line
2. program then reads a width for the read paragraph
3. program then justifies the paragraph based on tail adjustment
4. program then shows the result in a bounding box
5. program allows to adjust the paragraph by going back to 2
6. program ends when it reads 0 as a new width

**Restriction:  Do not use stringstream**. Instead parse each line read into words. A word should be defined as a sequence of non-white space characters  delimited by white-space before and after it. And, remember that getline does not read the new-line,  So here is a **hint**: Use the length of the line read to determine when your parse has reached the end of the line. Also the functionality provided **<cctype>** will be helpful.

**Hint**: For the solution of this problem you will have to read all of your words into a vector of strings.

Extra points (2 points) will be considered for even adjustment implementation.

Here is a hint to realize this mission - lines of words, say vector<string> words, is a straightforward conversion as implemented in the previous assignment and posted lecture example. Now, user types some width for justification. Let this width be *W*. The mission is to fill *i*-words in one line of this *W*. Note *i* > 0, i.e., every line must have at least one word. The logic will be:

1. let { w1, w2, ..., wi } be a collection of words. Then, w1.length() + w2.length() + ... + wi.length() is the total length of this collection
2. since we need at least one space between these *i* words, we need *i - 1* spaces at least, and therefore, we need *Wmin* = w1.length() + w2.length() + ... + wi.length() + (i - 1) characters, which must be less than or equal to *W*
3. your first loop must identify this *i* and*Wmin* by going through vector<string> words you created
4. in the second loop, you simply create a line by adding w1 through wi-1 by placing a space between the two consecutive words
5. before placing the last word wi, you need to place *W* - *Wmin* spaces because this number is the excess spaces to fill in for justification (tail adjustment)!

There is no miracle or hidden trick to deal with these operations.The left-hand side of the next example shows tail adjustment while the right-hand side shows even adjustment:

|  |  |
| --- | --- |
| Enter text, empty return will quit the input > Every photo, every edit, every album now lives  > in your iCloud Photo Library, easily viewable  > and consistent on all your devices.  > Automatically. The all-new Photos app makes  > it simpler than ever to find and rediscover    > your favorite photos. And you can make every  > shot look even better immediately after you've  > taken it with powerful new editing tools. > > Enter the width of text: 25 |-------------------------| |Every photo, every  edit,| |every album now lives  in| |your     iCloud     Photo| |Library, easily  viewable| |and  consistent  on   all| |your             devices.| |Automatically.        The| |all-new Photos app  makes| |it simpler than ever   to| |find and rediscover  your| |favorite photos. And  you| |can make every shot  look| |even  better  immediately| |after  you've  taken   it| |with powerful new editing| |tools.                   | |-------------------------|  Enter the width of text: 30 |------------------------------| |very photo, every edit,  every| |album now lives in your iCloud| |Photo Library, easily viewable| |and  consistent  on  all  your| |devices.  Automatically.   The| |all-new Photos app makes    it| |simpler than ever to find  and| |rediscover    your    favorite| |photos. And you can make every| |shot    look    even    better| |immediately after you've taken| |it with powerful new   editing| |tools.                        | |------------------------------| Enter the width of text: 0 | Enter text, empty return will quit the input >  Every photo, every edit, every album now lives > in your iCloud Photo Library, easily viewable  >  and consistent on all your devices.  > Automatically. The all-new Photos app makes  > it simpler than ever to find and rediscover    >  your favorite photos. And you can make every  > shot look even better immediately after you've  > taken it with powerful new editing tools. > > Enter the width of text: 25 |-------------------------| |Every  photo, every edit,| |every  album now lives in| |your     iCloud     Photo| |Library,  easily viewable| |and   consistent  on  all| |your             devices.| |Automatically.        The| |all-new  Photos app makes| |it  simpler  than ever to| |find  and rediscover your| |favorite  photos. And you| |can  make every shot look| |even  better  immediately| |after   you've  taken  it| |with powerful new editing| |tools.                   | |-------------------------|  Enter the width of text: 30 |------------------------------| |Every photo, every edit, every| |album now lives in your iCloud| |Photo Library, easily viewable| |and  consistent  on  all  your| |devices.   Automatically.  The| |all-new  Photos  app  makes it| |simpler  than ever to find and| |rediscover    your    favorite| |photos. And you can make every| |shot    look    even    better| |immediately after you've taken| |it  with  powerful new editing| |tools.                        | |------------------------------| Enter the width of text: 0 |