Project 7 - Loops

- In Project 7, you will utilize loops to process an input file on a character by character basis
  - For each line count the number of characters that are
    - Letters
    - Digits
    - Other (any other character that is not a letter or digit)
  - After counting characters on each line, output the totals for each type of character on that line
  - After reading all lines, a grand total for each type of character read is output.

Project 7 Constraints

- Use only material from Chapters 1-7 of your textbook
- ***Make your program's output match that of the provided sample solution – output information in columns as shown.***
- ***Echo print all inputs read using cin***

Sample Solution

- Run the sample solution to see what the input and output looks like.

/home/work/cpe211/Executables/Project_07/Project_07_solution

- Running the sample solution will show you the vertical spacing required in the output

- The comparison script is available

/home/work/cpe211data/Project_07/CompareSolution.bash Project_07.cpp

Project 7 information

- Opening the input file
  - If the first attempt to open a file fails, enter a loop that performs the following actions until a file name is successfully opened.
    - Print out an error message
    - Reset the input file stream

      input_file_stream.clear();
    - Prompt for, read and open another attempt
- After the input file is opened, use a priming read to determine if the file is empty

Project 7 information Continued

- Counting characters on a line

  `Abcd 1234 *())`

  The above line contains 4 letters, 4 digits and 7 other characters: 2 spaces, *()) and the new line character at the end of the line

- To count the characters use the functions **isdigit(char)** and **isalpha(char)** which are part of the **cctype** header file. – both functions return a non-zero positive integer (true) value if the argument is a digit or a letter respectively.

Project 7 information continued

- A character that is not a letter or a digit is counted as other – this includes the new line character.
- Output the character counts for each type on a line by line basis.
- The dashed line contain 50 dashes
- Output the totals for the entire file
- Use field widths of 15 for the first column and 10 for the next 4 columns

Nested while loops

- One way to write the program is a single loop that processes all characters in the file
- Another way to write the program uses 2 nested while loops to process the input file
  - Outer loop tests on the status of the file stream. It continues to loop until the input stream goes into the fail state.
    - Use a priming read for this loop
    - Need another read at the end of this loop

Nested while loops Continued

  - Inner loop processes the characters on each line of the input file
    - The priming read for the outer loop acts as the priming read for this loop as well
    - Terminates when new line character is read, and the new line character is not processed in the body of this loop
    - Make sure that the new line character is counted
    - Each character on a line is read inside this loop

Output

- Totals line has the phrase output in a field width of 15 and the 4 values are output in a field width of 10
- Percent line has the phrase output in a field width of 15 and the 3 values in a field width of 10
- On percent calculation, watch out for integer division