COMP 7500 Project 2 – A Pipe-Based Wordcount Tool

The program's objective is to find the number of words in a text file by running different processes and sending data between them using pipes. I have defined 3 functions which are error(), cw() and main(). The objective of error() is to check whether the given file exist or if the user given input is correct. The function is passing an integer argument which if passed is '0', it prints a message saying the given input by the user is in incorrect format, if '1' is passed, it prints a message saying unable to print file. The objective of cw () is to count the number of words and there are test cases to check multiple spaces between words and count them as one between two words. main() contains the process of opening files and retrieving words from them using two pipes. One pipe is used to send data and the other pipe is used to receive data.

The parent process starts, and the contents of the file is read and that is sent to child process. Process 1 opens and reads the file. The data is read and fed into an array. Process 1 will pass the array through a pipe which is sent to a child process. Process 2 counts the array content and sends the count number back to parent (Process 1) which displays the count as the output. Limited amount of data is read as defined by BUFFER, this is done to handle big files since it divides the files into smaller portions.

Function Prototypes: -

Void error(int em)

int cw(char* array)

int main (int argc, char *argv[])

The possible errors which can terminate or ask the user for new input: -

1. The program is run by giving the command: -
   ./<program_name> <file_name> (./pwordcount input.txt)
   If the file name is not provided, an error message is printed out as: -
   Please enter a file name

   Usage ./pwordcount <file_name>

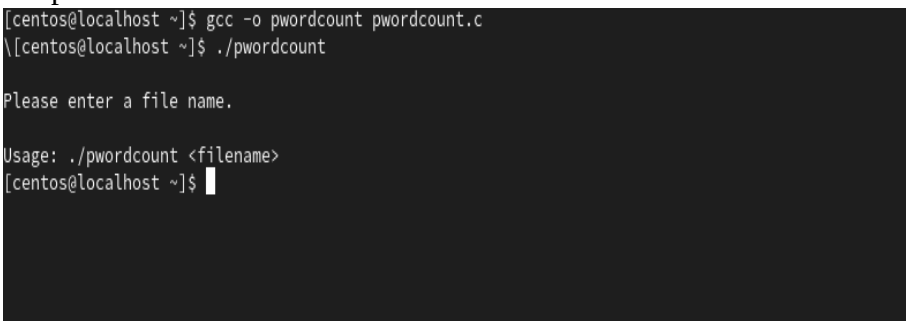
   If the file can't be accessed or if the file doesn't exist, then an error message is printed as follows: -

   Unable to open file
   Please check if file exists and you have read privilege.

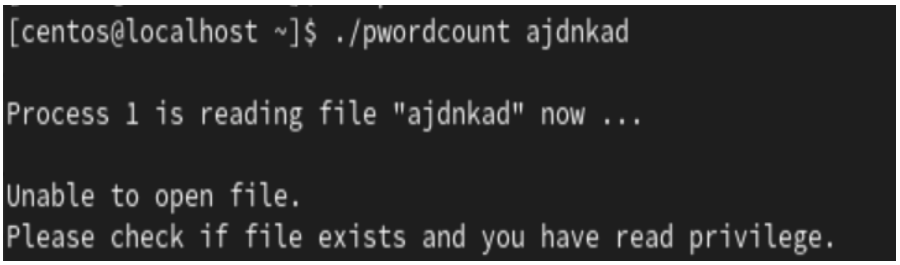
2. If the creation of pipe processing fails, an error message will be printed as follows: One of the Pipes Failed

3. If the creation of fork process fails, an error message will be printed as follows: Fork Failed!

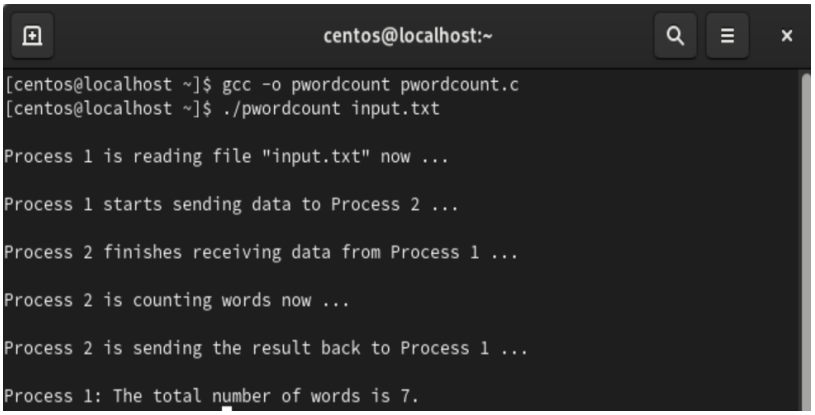
Output Screenshot are as follows: -



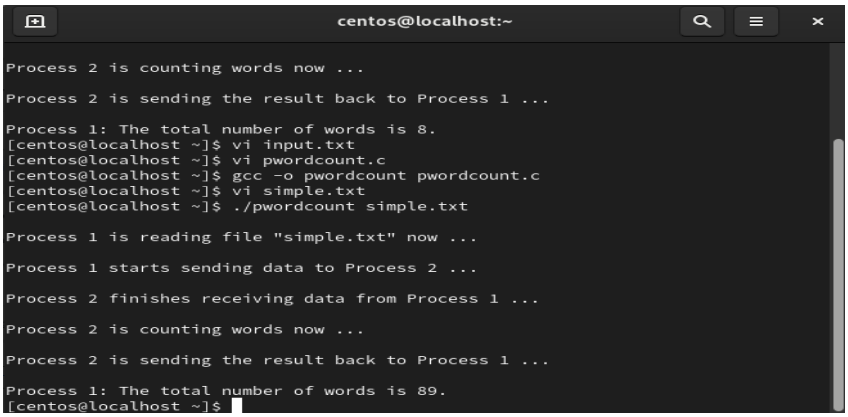
When the user runs the code without giving filename as input.



When the user runs the code with non-existing file.

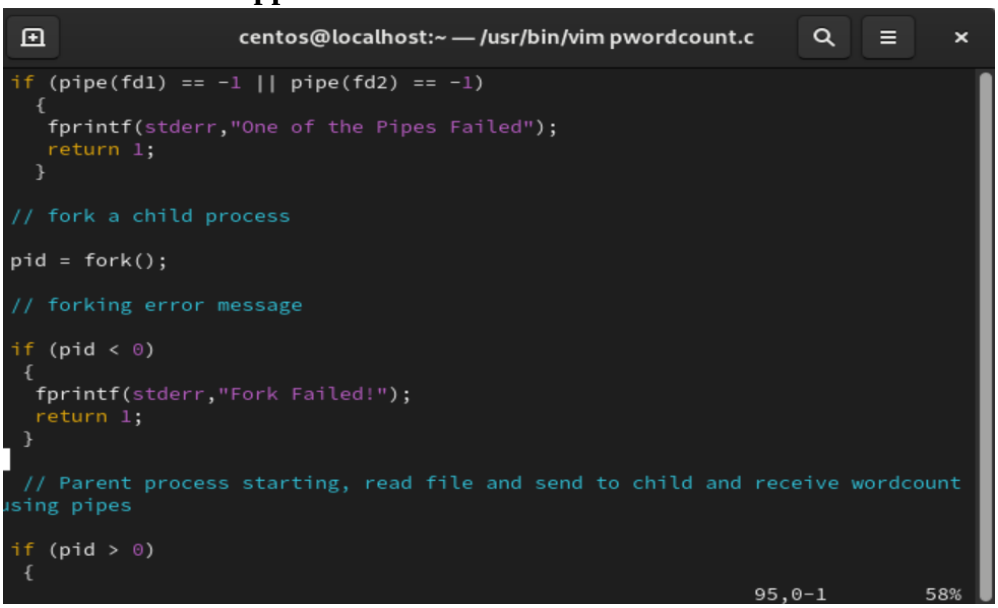


When the file is passed with the text “this is the second project of COMP7500”.

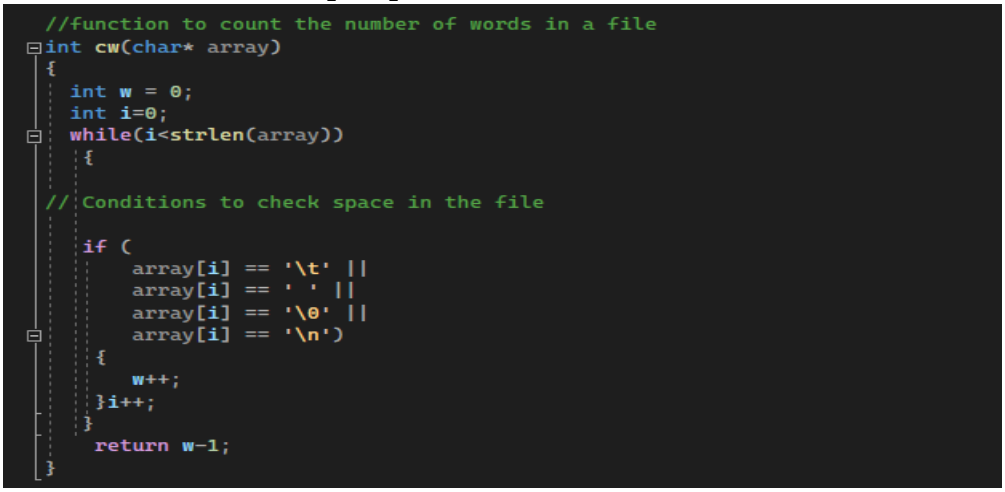


Sample Output

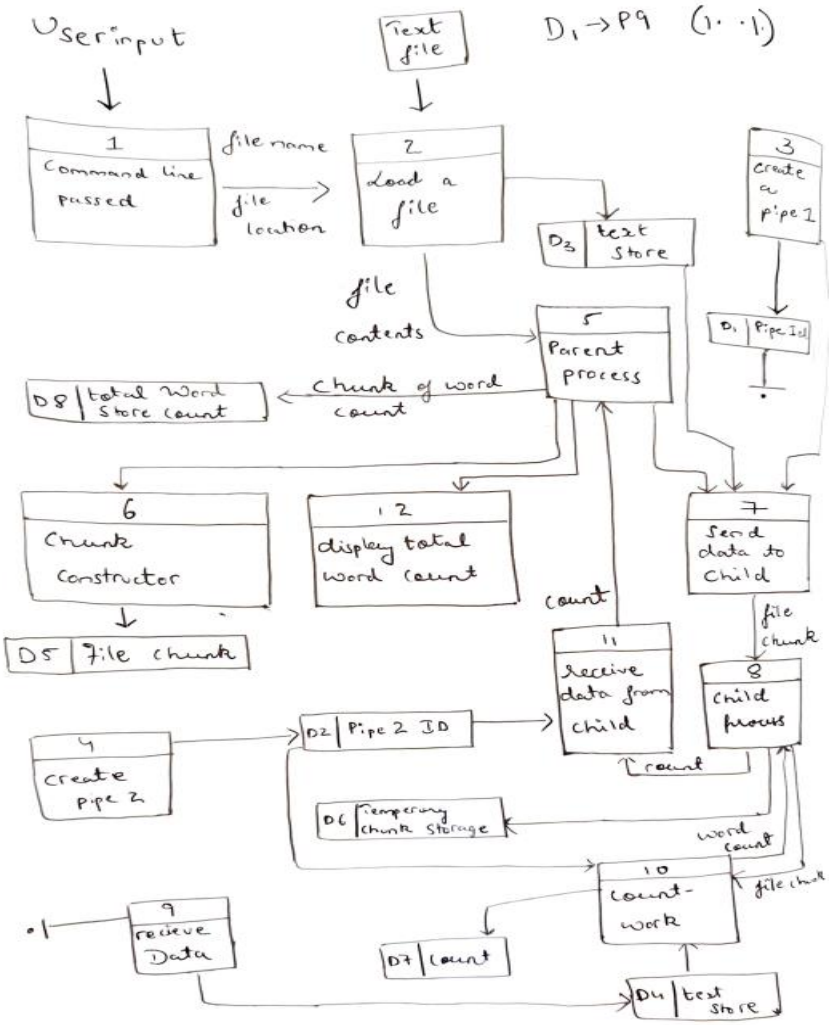
Main Function snippet: -



Function to check multiple spaces in file and count the number of words: -



Dataflow diagram: -



P2, P3, P4, P5, P6, P7, P12 are parent processes.
P8, P9, P10, P11 are child processes.