

# UNM

## Computer Science

The University of New Mexico

CS 341 — Computer Architecture and Organization: Assignment 1

**A low key intro**

Andree Jacobson

**Due: 1/27/2016 @ 11:59:59pm**

### Goal

Please note, this homework is a re-cap of the knowledge you should have from prerequisite classes and will assess your "C" programming skills and problem solving approach. Your individual effort is required on this assignment.

In order to receive full credit, all submitted programming assignments requiring C code to be submitted must:

- compile without warnings or errors on the CS lab machines when compiled with the following flags to gcc:  
`-Wall -ansi -pedantic`
- be properly commented using clear and concise language (i.e., no slang, instant messaging syntax, etc, etc...) Keep it professional as if your job depended on it! Also:
  - Each file starts with a comment which lists the author of the file (name and email), the date the file was created or submitted, and the purpose of the file.
  - Preferably commented in a style that can be parsed by doxygen.
- follow a commenting and indentation standard (pick one and stick to it)
- be accompanied by a makefile that compiles all code into the required binaries when invoked with the bare `make` command in the same directory as the make file, and... leaves only required files after typing `make clean`.  
(I realize the possibility that makefiles may not have been covered in the prereq courses, and I'll make exceptions for the first programs if this is the case.)

### Instructions

`pwl.c` Write a program that when passed an ANSI text file on the command line and invoked like so (in the directory where the binary is located): `./pwl ./myTextFile.txt` produces as output the following statistics about the text file:

- For the file: Absolute path, Number of paragraphs
- Per paragraph: Number of lines, Number of words, Number of characters

in the following format (see next page):

```
File      : /home/andree/cs341/assignment1/myTextFile.txt
Paragraphs : 5
```

```
Paragraph #1: 10 214 873
Paragraph #2: 5 34 143
Paragraph #3: 52 350 2356
Paragraph #4: 15 435 567
Paragraph #5: 200 500 3523
```

### **Notes:**

- A paragraph is defined as a section of text which is terminated by two `\n` (i.e., newline characters) with no other characters between them.
- Remember to use appropriate dynamic data types (i.e., use `malloc` rather than statically sized arrays)
- properly declare and initialize all your variables
- use procedural decomposition (i.e., don't put all your code in the "main" function, and
- avoid global variables like the plague!

I.e., use good programming form and style!

### **Discussion**

You may discuss the homework within the class (there is a discussion board on "Learn" that you can use, and the TA and instructor monitors it also. The rules are simple:

- Ask clarifying questions only
- Do not copy / paste any source code in the forum, and
- Do not reveal how to do something obvious that directly answers a question / problem in the homeworks.
- Feel free to share good online references you are using as a tool to solve the assignment, that you feel could benefit others. (Such links, like tutorials, How-To's, etc. will be gathered and listed in a references section on "Learn".

### **Turning in your homework**

Instructions for this will be made available shortly.