COMP10120 Practical Set 7: The C Pre-processor and C Helper Functionality

Please read the questions carefully. Name each program based on your student number, the practical set number and question number. For this set (set7), question 1 should be named 1234567s7q1.c where your student number replaces 1234567. All questions that you are submitting can be zipped into a single file called 1234567s7.zip, where 1234567 is your student number and s7 refers to set 7. Please also include a readme.txt file which says which compiler you used to test your implementation. This zipped file can be submitted via Moodle for grading.

Part 1

1. Write a C Program that **defines and uses a macro** which calculates the minimum of 2 integers. Extend this program with another macro to calculate the minimum of 4 integers. This extended program **must** use the macro which calculates the minimum of 2 integers. Submit the extended program. The program should get the input from the user from the console and print the result to the console.

2. Write a C Program which copies the contents of one file to another file. The names of the files should be passed to the program as command line arguments. For example

   MyFileCopier.exe file1.c file2.c

3. Write a C Program which contains a function with the following function prototype:

```
void myPrinter(const char *fmt, ...);
```

   The function should print each argument (except the first argument) on a new line. The first argument should indicate the types of the other arguments. For example, the function could be called as follows:

   myPrinter("ddcf", 5, 10, 'a', 5.5');

   which indicates the arguments are integer, integer character and float.

Part 2: Portfolio Ideas

1. Write a C Program that **defines and uses 2 macros;** one macro should print the elements of an integer array and the other macro should print the elements in reverse order. The array can be defined in the program as follows:

```
int myArray[10] = { 2, 4, 6, 8, 10, 20, 30, 40, 60, 80};
```

2. Write a C Program that dynamically allocates an array of integers. The size of the array should be based on the number of elements in a file which records the number of visitors to a website (transactions1.txt). The program should place the contents of the file in the array elements and print the contents of the array. The C Program should then reallocate memory for the array to allow the contents of transactions2.txt to be included in the array. The program should again print the contents of the array. You could vary, so the program must

take this dynamic nature of the data into account. (**Hint:** You could pass over the file once to count the number of lines, create an array based on this number and then use rewind() to return the file marker to the start of the file and scan the file again and start recording the integers into the array).

This program must use a dynamically allocated array of integers.