**SE185: Problem Solving in Software Engineering**

**Quiz # 10 (150 points)**

|  |  |
| --- | --- |
| Name: Adam Jennissen | Name: |

Answer the following questions and make a pdf file that includes the **source code, sample inputs, and outputs**. You must submit the **pdf file and all of the .c files** on Canvas for full credit. Do not forget to add your group partner name on the pdf file and the source codes.

1. **(50 points)** Write a complete C program that writes the following numbers to a file called “**numbers.txt**”.

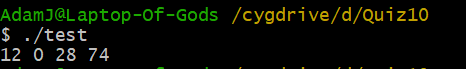
|  |
| --- |
| **12 7 3 0 28 74 37 299** |

* Then have your program read the file and store the even numbers into an **array** called **evenNums**. The array’s memory does not need to be dynamically allocated.
* Lastly, print out the array of even numbers separated by spaces.
* Your program must use **fopen()** and **fclose()** at least once and check **if your file is successfully opened** each time you use **fopen()**. [**fprintf, fscanf,** and **feof** may be useful functions]. You may begin with the following variable declarations:

|  |
| --- |
| FILE \*fp = fopen("numbers.txt" , "w");  int \*evenNums = malloc(8 \* sizeof(int)); // or int evenNums[8];  int curr;  int count = 0;  int i = 0; |

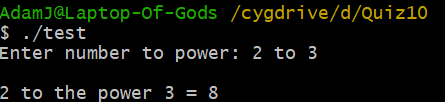
**Hint:** In order to read the file contents, the file needs to be closed after being in writing mode (“w”), then reopened in reading mode (“r”). Reading and writing mode (“r+”) can also be used, but the file must exist beforehand.

**Inputs and outputs format:**



1. **(50 points)** Write a complete C program which includes a recursive function to calculate the power of a number (e.g., xn) given by the user. (like the pow() function in the math.h library).

**Inputs and outputs format:**

****

1. **(50 points)** Write a complete C program which includes a recursive function to calculate the length of any string given by the user. (like the strlen() function in the math.h library).

**Inputs and outputs format:**

