

## ECE106 - Homework Assignment 2

All questions need to be answered by filling in the supplied `a2.cc` file. Please ensure that the file works correctly with the supplied `a2main.cc` and `a2.h` files. Note: if you are wondering about the approximate amount of effort involved, for my solutions all the function bodies are between 4 and 18 lines long and the entire file is between 100 and 110 lines long. You do not have to restrict yourself to these limits and can have longer or shorter answers.

Please submit the assignment by using the command:

```
submitece106s 6 a2.cc
```

1. Write the body of a function that returns the geometric mean of all the elements in a one dimensional array. Hint: use the `pow()` function.

The function declaration is:

```
double geomean(double a[], int n);
```

Note: if the array is NULL or `n` is less than or equal to zero, the function is to return 0.0.

2. If  $x$  is  $((c1 + 26 \times c2) \text{ modulo } 3) + 1$ , where  $c1$  and  $c2$  are the same as in the first homework assignment, what does the following code output if it is in `main()`?

```
double a[4]={ 1.5, 2.1, 4.2, 9.9 };
cout << setprecision(3);
cout << geomean(a,x) << endl;
```

3. Write the body of a function that transposes a square matrix, which is represented by a two dimensional array. Note that `N` is defined somewhere at the beginning of the file (and is assumed to be 4).

The function declaration is:

```
void transpose(double a[][N]);
```

Note: if the array is NULL the function is to perform no action.

4. If  $x$  is  $((c1 + 26 \times c2) \text{ modulo } 3) + 1$ , where  $c1$  and  $c2$  are the same as in the first homework assignment, what does the following code output if it is in `main()`?

Note: assume that `#define N 4` is at the beginning of the file.

```

double a[N][N]={
    {1.111,2.338,3.0*x,4.0},
    {1.111-x,2.338,3.0+x,4.0},
    {1.111,2.338-x,3.0,4.0+3.0*x},
    {0.111-x,1.338,4.0+x,2.0},
};
transpose(a);
cout << setprecision(2);
for (int i=0;i<N;i++) {
    cout<<a[i][0]<<" "<<a[i][1]<<" "<<a[i][2]<<" "<<a[i][3]<<endl;
}

```

5. Write the body of a function that reads a one dimensional array of imaginary numbers from a file, returns this array, and writes the size of the array to a memory location that is specified by one of the parameters.

Assume that the file consists of an integer that contains the number of elements followed by that many elements. Each element consists of the real and imaginary value. For example, the following is the contents of a file with a two element array:

```
2 1.0 1.0 2.1 1.9
```

The structure and function declarations are:

```

typedef struct complex_st complex;
struct complex_st {
    double real;
    double imaginary;
};
complex* readIn(const char* name, int* size);

```

The following is additional information about how the function is to work.

Assume that opening the file will succeed.

If the number of elements is less than or equal to 0, or an error occurs while reading this number, then readIn is to set memory pointed to by `size` to 0 and return NULL. Otherwise `size` is set to the number of elements in the array.

If an error occurs reading one of the elements, any values that have not been read need to be 0.0.

The only memory that is to be allocated is for the array.

6. Write the body of a function that reads in two one dimensional arrays of imaginary numbers, multiplies their elements together and prints out the resulting array to a third file (with a new line character after the array). The data structures are the same as for question 5.

For example, if the input files contain `2 1 1 2 2` and `2 2 3 4 1` then the output is `2 -1 5 6 10`. Recall that  $(a + ib) \times (c + id) = (a \times c - b \times d) + i(a \times d + b \times c)$ .

The function declaration is

```
void process(const char* in1, const char* in2, const char* outname);
```

The function needs three helper functions: one to read in one array (this function is in question 5), one to multiply the values, and one to write an array. Also, the function that multiplies the values needs a helper function that multiplies two values together.

Therefore the following additional functions need to be implemented:

```
void writeOut(const char* name, complex a[], int size);
complex* multiplyArrays(complex a1[], complex a2[], int size);
complex multiplyComplex(complex c1, complex c2);
```

The following is additional information.

Assume that opening the files will succeed.

Reading the files is the same as for question 5.

All memory that is dynamically allocated needs to be deallocated before `process()` returns.

If the either of the input arrays are `NULL` or have zero elements or have an unequal number of elements, then the output should be an array with zero elements (i.e. the output will be a single number: 0).

7. Please read the document at the following url:

```
http://education.calumet.purdue.edu/vockell/edpsybook/Edpsy1/edpsy1\_intro.htm
```

This is the beginning of an online book that I've found to be an interesting read. The book's url is

```
http://education.calumet.purdue.edu/vockell/edpsybook/
```

Which two principles did you find most interesting and why? Please limit yourself to at most 75 words (if you found more than two most interesting principles, arbitrarily reduce the set to two).