For all future communications, please use the following SUBJECT LINE:

CIS 27 Spring 2024 YourName : Needs/Questions

Without the above SUBJECT LINE, your emails will be deemed as SPAM/Phishing/Virus and will be deleted.

Turn in:

1. **Exercise #1** – Due on **Thursday, May 23, 2024 by 12:10 pm — Email Submission to class Gmail.**

   Final exam (and homework) formats/conventions/styles are explained in class or provided through Canvas. Students must follow the coding formats, conventions, and styles to obtain full credit for all assignments. More information was available through code demonstration in class — I do coding LIVE during class meetings and discussions, and I use ONLY Visual Studio IDE!

   a) For each exercise, a package must be generated to include the following items:

   - Copy of your source files (C programs → **`*.c, *.h`**)—your source files MUST HAVE A DRIVER NAMED as **`cis27Spring2024FinalExamYourName.c`**
   - Note! Replace "**YourName**" with your own full-name; i.e., **FirstnameLastName**.
   - Copy of output (copy and paste to the end of your program as **`PROGRAM_OUTPUT`** comment block)
   - Copy of **`Logic_Code_Output_Issues`** comment block (as a separate comment block) after the **`PROGRAM_OUTPUT`**.

   b) Emailing each package as follows,
   - One email message for each exercise.
   - The SUBJECT line of the message should have the following line:

     `cis27Spring2024FinalExamYourName.c`

   - Attaching ALL source files (only `*.c` and `*.h`) that was created in Part a) individually. Double check before email.

     `fractionYourName.h`
     `fractionYourName.c`

     `fractionUtilityYourName.h`
     `fractionUtilityYourName.c`

     `polyNodeYourName.h`
     `polyNodeYourName.c`

     `polyNodeUtilityYourName.h`
     `polyNodeUtilityYourName.c`

`polyTermYourName.h`
`polyTermYourName.c`

`cis27Spring2024UtilityYourName.h`
`cis27Spring2024UtilityYourName.c`

The above files are also from the setup code!

- Only email the above `*.h` and `*.c` files !
- Do not email `ZIP` files, image files, `*.sln` files. You will get ZERO if you email any of these files.

Note 0! "`YourName`" means `FirstnameLastname`—no abbreviation!

2. Reminder!
   - Getting the program to work *is not enough to earn full credit*. Your program must run correctly and follow all proper conventions and consistent styles as explained in class, and produce the exact textual information/display as shown/required (of course, with appropriate user input data/values for each run/selection!) to receive credit accordingly.
   - Also, your program must work with all reasonable data sets or patterns to start with.
   - Again, writing code is not just the code works. It also involves care, patience, coding idioms + forms, and other reminders. Please see the posted code written in class and the coding convention C file.
3. You will get ZERO points if your code does not compile! Please make sure that you compile your code frequently and correctly throughout the working session. Please check and run your submission again exactly as you just submitted.
4. Q.E.D.

****

More Notes!

- You are only allowed to use `printf()` and `scanf()` from the `stdio.h` header.
- You are not allowed to use any other syntax structures that are not introduced in class and class meetings — please confirm with the instructor if you have any doubt!
- You are not allowed to use functions written by someone else.
- You must write all functions yourself before using them in the required class work.
- Except for the ***member data, function arguments and local variables within member functions, and indices of `i`, `j`, `k`***, all other variables must have the initials of your first name and last name added to the end of the variable names.

For examples, assuming the name is "`Nice Effort`",

```
int usrInputNE;
int digitCountNE;
int absValueNE;
int tmpNE;
```

- All function names must have the initials of your Firstname and Lastname appended at the end.

  For examples, assuming the name is "`Nice Effort`",

  ```
  void displayClassInfoNE(void);
  void runMenuFinalExamNE(void);
  ```

- All **filenames** must have your complete **Firstname** and **Lastname** appended as required.
- The values of the elements from the created array, it any, MUST NOT be changed/modified at all during the analysis steps and process unless they are required to be updated or changed based on the specified tasks.
- Again, you must declare all local variables at the top of the function; one declaration per statement except for the indices that you may have.
- Please be consistent with the above format requirements.
- Please follow all of the code formats, idioms, and practices that have been presented in class as well as in code and coding convention C file.
- Penalties will be applied for bad coding and practices as discussed/quizzed/demonstrated throughout class.
- After copying the output to the driver **`cis27Spring2024FinalExamYourName.c`**, do not manually modify/change/insert text for the output of your code! You will get zero (0) for the work! I will run your code and know if you manually alter the pasted output as submitted in your C file!

Reminder!

- In your program, no GLOBAL DATA are allowed, and you must write all needed functions (no library functions values are allowed).
- Again, writing code is not just the code works. It also involves care, patience, coding idioms + forms, and other reminders. Please see the posted code written in class and the explanation on coding convention/style C file.

Plus More notes!

1. We write code to manipulate data (which are provided by the user) to produce the required outcome in the most efficient way!
2. Getting the program to work is not enough to earn full credit. Your program must run correctly and follow all proper convention and consistent styles as explained in class in order to receive credit accordingly.
3. Writing code is not just the code works. It also involves patience, care and code idioms + forms along with others. Please see the posted code that have been written in class as well as the coding convention C file — posted on Canvas.

And plus more notes!

1. You will get zero (0) points if your code does not compile! Please make sure that you compile your code frequently and properly throughout the working session. Please check and run your submission again exactly as you just submitted.
2. A `runMenuFinalExamNE()` function has a menu that MUST BE a combination of `do-while` and `switch`. Any other form of the menu will receive 0 for the whole final exam.
3. You will be penalized heavily if there are violations on code conventions as explained in class, homework submissions, quizzes, and document posted on Canvas!
4. Your code must work with all reasonable input data sets/patterns. At least, your code should be tested with the given data samples indicated/given from the exam/document, etc.
5. Pay attention to naming; that means the specified/required names, your own generated names, and filenames. Penalty points will apply if you do not follow the instructions.
6. You are only allowed to use `stdio.h` and `stdlib.h` for `printf()`, `scanf()`, `malloc()`, and `free()`. All other functions must be written by you!
7. All function names must have the initials of your firstname and lastname appended at the end.
8. Q.E.D.

****

Exercise Assignment

Exercise 1 – Due Thursday, May 23, 2024 by 12:10 pm — emailing to class EMAIL (Gmail)

(1) Write a C program with calls to functions to produce the output given below.

(2) First, the program should display the output to screen as

```
We write code to manipulate data (which are
provided by the user) to produce the
required outcome in the most efficient way!
```

You need to replace "**`Your Name`**" with your real when you submit your work/email.

The above result should come from a call to a function named as `displayClassInfoFL()`, where `FL` must be replaced by the initial of your first name and the initial of your last name. For examples, if your name is **`John Smith`** then `FL` should be `JS` throughout all of your work/code as mentioned; the function name `displayClassInfoFL()` will become `displayClassInfoJS()`.

(3) Then, the program will continue with calls to the required functions to display the remaining results. The sample output is given on the final exam day!