Computational Physics Midterm Exam.

Weonjong Lee

2020-10-27

When you hand in the homework, you should gather all your files into a single tarball file as follows.

- Use an unix command tar -czf <file name>.tar.gz <file 1> <file 2> $\cdots$.
- For undergraduate students, put a copy of a tarball <file name>.tar.gz into a directory:

  /physics/upload/comp2023/<user-ID>.
- For graduate students, put a copy of a tarball <file name>.tar.gz into a directory:

  /physics/upload/acomp2023/<user-ID>.
- You must use the GNU `make` command and `Makefile` to compile the code.
- You must make a README file which describes how to run your code. The README file should include your name and student ID.
- You must use `gnuplot` to make a plot into a PDF format.

C Programming Language

1. ⟨ `Prime number factorization` ⟩ [30 points]

   Find a data file `input.dat` which contains a list of positive integers.

   (a) Make a code to read in the file `input.dat`. The code should not assume that the number of lines in the file is known.

   `[HINT]` You may use `feof`() function to check the end of file.

   This code will provide the following set of input integers:

   $$S_1 = \{x_i | \ x_i > 0, \ i = 0, 1, 2, 3, \cdots, (N-1)\} \tag{1}$$

   Here, x_i is a positive integer (an input number). The number of elements in S_1 is N.

   (b) For each element x_i, make a code which finds all the prime numbers less than x_i.

   $$S_i^p = \{p_j | \ p_j < p_{j+1}, \ p_j \leq x_i, \ 0 \leq j \leq N_i^p\} \tag{2}$$

   Here, S_2 is a set of prime numbers (p_j).

   (c) For each element x_i, make a code to perform a prime number factorization as follows.

   $$x_i = \prod_{j=0}^{N_i^p} (p_j)^{n_j} \tag{3}$$

   Here, p_j is a prime number, and $n_j \geq 0$ is an integer.

   (d) The results should be printed into an output file: `output.dat` under the following conditions.

   - Print results for the set S_i^p for each x_i as follows. For example, if $x_2 = 20$,

   $$S_2^p = \{\ 2, \ 3, \ 5, \ 7, \ 11, \ 13, \ 17, \ 19\} \tag{4}$$

   - Print results for the prime number factorization, as follows.

   $$x_2 = 20 = 2^2 * 5 \rightarrow (2, 2) * (5, 1) \tag{5}$$

   $$(a, b) = a^b, \qquad (2, 2) = 2^2 = 4, \qquad (5, 1) = 5^1 = 5 \tag{6}$$

2. ⟨ `Factorial` ⟩ [30 points]

When you calculate factorial of a number (ex: 100!), you will be caught in a problem of precision. The number is too big to be represented by the integer (int) type in C/C++ language. Note that the integer type (int) number should be smaller than $2^{32} = 4294967296$. Therefore if the number is bigger than 2^{32}, it is not possible to represent it using the integer type. In the case of gamma functions or factorial of a number (ex: 100!), it is usually much bigger than 2^{32}. Hence, in order to calculate this big number, one needs a code which can handle an arbitrarily high precision in integer arithmetics. The main goal of this problem is to write a program which can calculate factorial (gamma function) of an arbitrary integer to a full precision. Note that one should not use a double precision floating point number, which can not represent the full precision. One should use an array of integers to represent the gamma function of an arbitrary positive integer. The code should be general enough to calculate $x!$ for an arbitrary integer x and print it in its full precision.

`[HINT]` For the mathematical definition of the gamma function and factorial, refer to Chap. 13 of the book: Mathematical Methods for Physicists (7th Edition) by Arfken, Weber, and Harris.

Make a code which satisfies the following conditions.

(a) In the class, you learned how to define arrays on the data memory using the `malloc()` and `calloc()` functions. You must use these to represent a very big number, which can not be represented by the interger type.

(b) In the class, you learned self-referential structures. You must use these to represent a very big number, which can not be represented by the interger type.

(c) `[HINT]` Think about an array of integers and each element of the array represent a part of a big integer. For example, let each integer in the array represent two digits of the big integer. Then all you need to know is how to multiply a number to this array.

(d) The code must also convert the full precision number of $x!$ into a double precision number and print it.

3. $\langle$ `File I/O` $\rangle$ [40 points]

Find the data file "`data.2023-10-27`". This file has the data format of two columns: the first column gives values for the x variable and the second column gives values for the function $f(x)$. Please note that $0 \leq x \leq 6$. Each line of the data has a value of x and the corresponding value of $f(x)$.

(a) $\langle$ `File I/O` $\rangle$ Make a code to read in the data of $(x, f(x))$ and determine the number of the whole data sets. You must use the `feof()` function to check the end of file.

(b) $\langle$ `Statistical Analysis` $\rangle$ Make a code to calculate the average and statistical error of $f(x)$ for each x value. Let us say that $f(x, i)$ represents the data at coordinate x for the ith data set. Then the statistical average ($\langle f(x) \rangle$) and error ($\sigma(x)$) is defined as

$$\langle f(x) \rangle = \frac{1}{N} \sum_i f(x, i) \tag{7}$$

$$\sigma^2(x) = \frac{1}{N(N-1)} \sum_i \Big[f(x, i) - \langle f(x) \rangle \Big]^2 \tag{8}$$

(c) $\langle$ `Print results` $\rangle$ Print results for the average and error of $f(x)$ as a function of x into a file named `stat.out`.

(d) $\langle$ `Plot results` $\rangle$ Make a plot of the average $\langle f(x) \rangle$ and its statistical uncertainty $\sigma(x)$ as a function of x. The plot must be made into a PDF format (plot file name: `fig_1.pdf`).

`[HINT]` : Use `gnuplot` to make a pdf file of the plot. A manual for `gnuplot` is posted on the web site.

(e) $\langle$ `Interpretation` $\rangle$ From the statistical analysis on the results for $f(x)$, let us guess that the functional form is $f(x) = ax^3 + b$, and then obtain the statistical average of a and b.

`[HINT]` One possible idea: Then you may choose couple of data points randomly to determine a and b. Then you can obtain a statistical sample for a and b.

(f) $\langle$ `Advanced question` $\rangle$ Obtain the statistical error of a and b.

`[HINT]` If you have a statistical sample, you can obtain the average and error. The subtle part is how to treat correlation between a and b. If this question is too difficult for you, then you may skip it.