Problem 1

Give the binary format of $-5.28$ as a double floating-point number. If the number can not be represented exactly, it should be rounded to the nearest number using rounding-even scheme.

Problem 2

Answer the following questions. For questions requiring experiments in C language, compile your code with the GCC compiler. The code and output should be included.

(a) Are the data types float and double in C language guaranteed to conform to the IEEE 754 standard? Please find statements from the C99 standard

[https://www.open-std.org/jtc1/sc22/wg14/www/docs/n1256.pdf](https://www.open-std.org/jtc1/sc22/wg14/www/docs/n1256.pdf)

to support your answer.

(b) In a regular C program, what value does the expression $0.0$ convey? +0.0 or −0.0? Please find the statement in the manual


that supports your answer and perform experiments with a C program.

(c) According to the C99 standard, what function (or macro) from the standard libraries can we use to get the sign bit of a floating-point number? Find statements from the C99 standard to support your answers.

(d) How do we specifically assign +0.0 and −0.0 in C language? Experiment with C codes and validate the assigned value using the function you found in (c).

(e) Assuming that the float variables in your implementation of C conform to IEEE 754, we can write our own function to get the sign bit using bitwise operators. Please write a C function that returns the sign bit of a float variable. Your function should return 0 for positive numbers and 1 for negative numbers. Your experiment should ensure the function is correct on normal, denormalized values and special quantities ±0.0 and ±∞.

(f) Suppose we have two floating-point numbers $a < 0$ and $b$, where $b$ is a number that is neither NaN nor ±∞.

Also, we have a C program that contains the following line:

$$c = a / \max(b, 0.0);$$

We wish to guarantee that

$$c < 0$$

always holds (You can assume that $b$ is not too large, so no underflow occurs when calculating $c$).

Which implementation should we use for the “max” function? Explain your choice and validate your answer with experiments.

(1) $$(x > y) ? x : y$$

(2) $$(x < y) ? y : x$$

Hint: “max(b, 0.0)” should not return any negative number.