1 Problem 2-1
Give the binary format of $-5.28$ as a double floating-point number.

2 Problem 2-2
Answer the following questions. For each question, show your experiments with C language with GCC compiler to check your arguments.

(a) In a regular C program, which is the representation of 0.0 ? +0.0 or −0.0. Please find the statement in the manual https://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html

that supports your answer.

(b) How do we specifically assign +0.0 and −0.0 ?

(c) Please give the definition of a function that returns the sign of a number with type float. Make sure it is correct on normal values as well as special quantities like ±0.0 and ±∞. Your function should return as follows for the special quantities:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>+0.0</td>
<td>1</td>
</tr>
<tr>
<td>-0.0</td>
<td>-1</td>
</tr>
<tr>
<td>∞</td>
<td>1</td>
</tr>
<tr>
<td>-∞</td>
<td>-1</td>
</tr>
<tr>
<td>NaN</td>
<td>0</td>
</tr>
</tbody>
</table>

(d) 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.

(1)  

$(x > y) ? x : y$

(2)  

$(x < y) ? y : x$

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