**C Language Programming: Homework #6**
**Assigned on 12/08/2015(Tuesday), Due on 12/15/2015(Tuesday)**

1. Write a program that can input a float or double number and print out its bit pattern and vice versa (input a 32-bit or 64-bit pattern and output its value).
   Note: you should use the three techniques mentioned in the class:
   (a) an integer pointer to float or double,
   (b) union, and
   (c) bit field
2. Please check:
   1. Is it correct that the value,
      *1.17549435082228750796873653722224567781866555677208752150875170627841725945472717285156050000000000000000000000000000000e-38f* ,
      is the smallest floating point number as stated in the textbook. If not, what is the smallest floating point number ?
   2. What is the bit pattern of f=0.0
   3. run

```
f1 = 1.17549435082228750796873653722224567781866555677208752150875170627841725945472717285156050000000000000000000000000000000e-38f;
f2 = 1.17549435082228750e-38f;

if( f1==f2 ) { printf("%100e = %100e", f1, f2); }
else { printf("%100e != %100e", f1, f2); }
```

   Explain the result.

---

# Requirement :

1. Write two programs named **hw6_1.c** and **hw6_2.c**.
- In hw6_1.c, you should use integer pointer to convert number.
- In hw6_2.c, you should use union to convert number.

Remember that you can input float or double number and vice versa in both programs.

2. Input number can be negative.
3. Question2 , please answer three questions on report.
   Ex:
   2-1 : Yes, because…
   2-2 : No, because…

4. Here is the input Example:

➢ Please follow the order of input like example below

“

float number,
binary number to float,
double number,
binary number to double

”

```
Input the float number:-3.5
11000000011000000000000000000000
Input binary number to convert float number:
11000000011000000000000000000000
-3.500000e+00

Input the double number:
-3.5
1100000000001100000000000000000000000000000000000000000000000000
Input binary number to convert double number:
1100000000001100000000000000000000000000000000000000000000000000
-3.500000e+00
```

You can use the Executable named **a.out** in /home/data/hw6 or this website to verify your answer.

Score:

- Integer pointer : 25%
- Union : 25%
- Correctness : 30%
- Report : 20 %