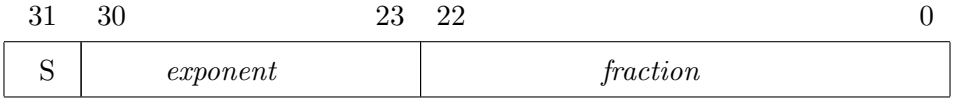
Question 1 (20 points): Write a subroutine called `BiggerThanTen` that receives a value x in `$a0`. The value x is represented in 32-bit IEEE 754 floating-point representation. `BiggerThanTen` returns one of the following combination of values:

Return Values		**Meaning**
$v1	**$v0**	
1	1	x is +infinity or -infinity
1	0	x is not a number
0	1	x is larger than +10
0	0	x is smaller than or equal to +10

Recall that the 32-bit IEEE 754 representation has the following specification:



$$N = \begin{cases} (-1)^S \times 0.fraction \times 2^{-126} & \text{if } exponent = 0 \\ (-1)^S \times 1.fraction \times 2^{exponent-127} & \text{if } 0 < exponent < 254 \\ (-1)^S \times \infty & \text{if } exponent = 255 \text{ and } fraction = 0 \\ NaN & \text{if } exponent = 255 \text{ and } fraction \neq 0 \end{cases}$$

- (**10 points**) What is the binary representation of +10.0 in the IEEE 754 floating-point representation?

- (**20 points**) Write the MIPS subroutine `BiggerThanTen`. Follow all the MIPS subroutine calling conventions. You are not allowed to use any floating point instructions in your subroutine.