Muazzam Shaheryar

CSC 3100

Decimal to IEEE754 Single and Double Converter Explanation

This is a very simple convertor really. In the start you are greeted with a menu where you can select what feature you want to use. The menu uses while loop and switch statements to work properly.

**Decimal to IEEE754 Single Precision**

In this method you are first prompted to enter a decimal value. This value is first checked if it is negative or positive and if it is negative then the negative value is removed and moved to the next part. It is then split into 2 halves by separating at the decimal place as number left to the decimal place are converted differently than those on the right. When separated the numbers on the right lose the decimal point so they become whole number in the eyes of the program. So, to change this, I first count the length of the numbers on the right and multiple 1 by 10 that many times and divide the number with this newly acquired number. This brings the decimal point back. The left side is converted first. It is converted how normally you would convert it, nothing special. The program has converted value already at least 3 or 4 times. This will continue throughout the program. Step have been taken like using BigDecimal to ensure preciseness of the result. Then the right side is converted like how it normally is. Then we find the exponent of the mantissa and then find the adjusted exponent. Then we make sure the mantissa is at 23 bits nothing more nothing less, we now get the sign number (0 or 1 depending on if the original number was positive or negative) and combine everything. All of this is done while also showing work. If I was not showing work, I could have used Integer.toBinaryString for faster coding in some places.

**Decimal to IEEE754 Double Precision**

Works basically the same as the single precsion one but with 64 bits instead of 32 and the adjusted exponents is changed accordingly and the added bits for mantissa are used properly

Overall this converter is simple, yet it took me 3-4 hrs. simply because of some little frustrating conversions every second and trying to maintain precision of the numbers.