Question 1 (30 points):

NVIDIA has defined a “half-precision” floating pointing format for use in its Graphics Processing Units (GPUs). A floating-point number is represented in this format in 16 bits as follows: the most significant bit is the sign bit, next there are 5 bits used for the exponent, and 10 bits for the fraction. This format is illustrated below:

15	14 10	9 0
S	*exponent*	*fraction*

The exponent is expressed in excess-16 format (also known as a bias representation). Given the binary representation above, the decimal value of the number represented can be computed by the following expression:

$$N = \begin{cases} (-1)^S \times 0.0 & \text{if } exponent = 0 \text{ and } fraction = 0 \\ (-1)^S \times 0.fraction \times 2^{-14} & \text{if } exponent = 0 \text{ and } fraction \neq 0 \\ (-1)^S \times 1.fraction \times 2^{exponent-15} & \text{if } 0 < exponent < 31 \\ (-1)^S \times \infty & \text{if } exponent = 31 \text{ and } fraction = 0 \\ NaN & \text{if } exponent = 31 \text{ and } fraction \neq 0 \end{cases}$$

a. (**8 points**) Give the bit pattern for the representation of the number -4.125_{10} in this notation.

15	14 10	9 0
1	10001	0000 1000 00

$4.125_{10} = 4.0 + 0.125 = 4 + \frac{1}{8} = (-1)^0 \times 100.001 = 1.00001 \times 2^2$
$\Rightarrow exponent - 15 = 2 \Rightarrow exponent = 17$

Let `A = 0x7800` and B = 0x4D00 be two floating pointing numbers in this format.

b. (**8 points**) What is the value of `A` and the value of `B`? Express each of these values both in normalized base-two notation and in decimal notation.

`A =`

15	14 10	9 0
0	11110	00 0000 0000

$A = (-1)^0 \times 1.0 \times 2^{30-15} = 1.0 \times 2^{15}$
$A = 1000\ 0000\ 0000\ 0000_2 = 2^{10} \times 2^5 = 1024 \times 32 = 32768_{10}$

`B =`

15	14 10	9 0
0	10011	01 0000 0000

$B = (-1)^0 \times 1.01 \times 2^{19-15} = 1.01 \times 2^4$
$B = 10100_2 = 16 + 4 = 20_{10}$

c. **4 points**) What is the true value of $A + B$ expressed in decimal notation? In other words, what is the value of $A + B$ if an infinite precision could be used to compute the addition and to store the result?

$A + B = 32768 + 20 = 32788_{10}$

d. (**5 points**) Assume a floating-point unit uses the NVIDIA format presented above. This unit has no guard, no round, and no sticky bits. What is the value of $A + B$, expressed both in normalized base-two notation and in decimal notation, computed by this machine?

To align `A` with `B`, we need to move the binary point of `B` eleven positions to the left. Therefore:

$B = 0.0000\ 0000\ 0010\ 1 \times 2^{15}$

```
          mantissa

  A = + 0.0000 0000 00

  B = + 1.0000 0000 00
----------------------------------------------

A+B =   1.0000 0000 00
```

Therefore $A + B = B = 1.0 \times 2^{15} = 32768_{10}$

e. (**5 points**) Assume a floating-point unit uses the NVIDIA format presented above. This unit has one guard, one round, and one sticky bit. What is the value of $A + B$, expressed in normalized base-two notation, computed by this machine?

```
          mantissa       Guard  Round Sticky

  A = + 1.0000 0000 00|    0      0       0

  B = + 0.0000 0000 00|    1      0       1
----------------------------------------------

A+B =   1.0000 0000 00|    1      0       1
```

Now we have to round up because of the sticky bit. Therefore the result is:

$A + B = 1.0000\ 0000\ 01 \times 2^{15} = 1000\ 0000\ 0010\ 0000_2 = 32768 + 32 = 32800_{10}$