Homework #4

1) Encod the following numbers as single-precision floating point numbers.

a) -8192

$= 10\ 0000\ 0000\ 0000.00\ldots$

$= (1.0) \times 2^{13}$

$\Rightarrow$ Encode

$= (-1)^1 \times 2^{13} \times 1.0000..0$

$s = 1,\ e = 140,\ m = \underbrace{000..0}_{23\text{ bits}}$

$1\ \underbrace{100|0}_{C}\ \underbrace{110|0}_{6}\ \underbrace{000|}_{0}\underbrace{0000|}_{0}\ 0000|0000|0000|0000|$

$\quad\quad 0 \quad 0 \quad 0 \quad 0$

$\Rightarrow$ 32b hex value : 0x C600 0000

b) $1.0625_{(10)} = 1.0001.00$

$\Rightarrow$ Encode : $(-1)^0 \times 2^{127-127} \times 1.0001..000$

$s = 0,\ e = 127,\ m = \underbrace{0001..0}_{23\text{ bits}}$

$0\ 011|1\ 111|1\ 000|1\ 000|0\ 000|0000|0000|0000|$

$\quad 3 \quad F \quad 8 \quad 8 \quad 0 \quad 0 \quad 0 \quad 0$

$\Rightarrow$ 32b hex value : 0x 3F88 0000

c) 0.4 = 0.0110011001 1

→ 1.100110011 × $2^{(-2)}$

→ S = 0, e = 125, m = 10011001100

0 011|1 110|1 100|1100|1100|1100|1100|1100|

3 E C C C C C C

32 b hex value : 0x 3E CC CCCC

d) 256 + 1/512

1 0000 0000. 0000 0000 1.. (2)

= 1.0000 0000 0000 0000 1..... × 2^8

S = 1, e = 135, m = 0000 0000 0000 00001 ....

Encode :

1 100|0 011|1 000|0 000|0 1 00|0000|0000| 0 000|

C 3 8 0 4 0 0 0

→ 32 b hex value 0x C3804000

e) 2.0009765 62 (10)

= 10.0000000001

= 1.0000 0000 001... × 2^1

s = 1, e = 128, m = $\underbrace{0000\,0000 \quad 001..}_{23\text{ bits}}$

1 100|0000|0 000|0 000|0 001|0000| 0000|0000|

C 0 0 0 1 0 0 0

→ 32 b hex value = 0x C000 1000

5)

Memory Virtualization : is a technique used by operating Systems to hide the complexity of programs being allocated to dozens of different blocks scattered all over the memory. with virtual memory programs can assume their memory always starts at 0, keeping things simple and consistent

Virtual memory takes program addresses and maps them to RAM addresses

Role of virtualization in memory protection :
Since memory virtualization maps the program's addresses to different addresses in the RAM , or allocate each program its own memory, programs are better isolated from one another and cannot access each other's data
→ This will help protect against malicious software like viruses
→ Program can't corrupt each other

ⓑ Fragmentation : A process is loaded and removed from memory, the free memory space after removing is broken into pieces
→ After sometimes that process cannot be allocated back to the memory because of small size and the memory block remains unused
This problem is called fragmentation
ⓒ Paging : is a non-contiguous allocation in order to reduce the complexity of program

6)
A cache : is a hardware or software component that stores data so that future requests for that data can be retrieved faster
Principle of Locality
① Temporal Locality : refers to the reuse of specific data items within a short time frame . After fetching an item from main memory and storing it in cache, any subsequent calls for that data can be done faster as programs running on a processor tend to use the same variables and structures
② Spatial locality : refers to the tendency of data items that will be needed soon to reside in memory locations near or adjacent to items that are needed now . This can be the result of either programmers or compilers clustering items in memory
③ Algorithmic locality is the tendency of applications to perform operations on related data items though not in any short time period and despite the fact the items are not each other in memory

Since the cache already stores copies of data and instructions from RAM, the CPU don't have to go back to retrieve the data and the data already stored only have to wait to be used by the CPU

3)

```
bro32:
        MOV R1, R0
        MOV R0, #0
        MOV R2, #0x8000 0000
        MOV R3, # 0X 0000 0001
bro-loop:
        TST R1, R2
        ORRNE R0, R0, R3
        MOVS R2, R2, LSR #1
        MOV R3, R3, LSL #1
        BNE bro-loop
        BX LR
```

Times		clock		total clk
1	x	1	=	1
1	x	1	=	1
1	x	1	=	1
1	x	1	=	1
32	x	1	=	32
32	x	1	=	32
32	x	1	=	32
32	x	1	=	32
31	x	3	=	93
1	x	1	=	1
1	x	2	=	2

→ Total clocks:

$1 \times 4 + 32 \times 4 + 93 + 1 + 2 = 228$

frequency of the clock : 4 GHz

$T = \frac{1}{f} = \frac{1}{4} = 0.25$ nano sec

$\rightarrow$ execution time $= (0.25)(228) = 57$ nanosec

2) Assume float $x = 4,194,304$

$x = 4,194,304 = 2^{22} = 1.0 \times 2^{22}$

$\rightarrow S = 0, \; e = 149, \; f = \underbrace{0.\ldots0}_{23 \text{ bits}}$

$2^{22} \times 1.0000\,0000\,0000\,0000\,0000\,000$

The smallest positive number that can be added to x is 1

The ratio of the large to the smallest single-precision floating point number that can be added together without a loss of accuracy : 2^{22}

4) The value of SP is : 0x 2000 1024