Homework Assignment-2

Roll no- MA17BTECH11004

Name - Hitesh Kumar Schrawat

Ans1- left most bit ⇒ signed bit → 0

Exponent = 01111101

= 125

⇒ actual exponent = 125 - 127 = -2

fraction = $2^{-2} + 2^{-4} + 2^{-6} - - - - + 2^{-22}$

significand = 1 + fraction

$= 1 + 2^{-2} + 2^{-4} + - - - + 2^{-22}$

$= \frac{1 - 2^{-22}}{1 - \frac{1}{4}} = \frac{1 - 2^{-22}}{3}$

So, number = $(-1)^0 \left(\frac{1 - 2^{-22}}{3}\right) \times 2^{-2}$

in decimal:

= 0.333 33331347

Ans2-

Number in FP = 5.6677

Single Precision = 0 10000001 01101010101110111001100

Actually stored = 5.66769981....

difference will be = 0.0000002

double Precision = 0 1000 0000001 [illegible]

mantissa → 01101010111011001100011000111111000101000000100100

Actually stored = 5.66769999....

difference = 0.0000000000000001

Ans3- We have to load values at 0x1000 & next 3 values in Vr1

assuming the address of r1 be 0x1000

or by using command

mov r0, 0x1000 Semantics
Vald vr1, [r1] vr1 ← ([0x1000], [0x1004], [0x1008], [0x100C])

Ans4: If policy is open row →
Misses → (4,15)
Hits → (7,150), (4,9), (6,10), (11,10)
Conflicts → (4,6), (6,150), (3,9), (4,12), (4,150), (5,10).
No. of Misses = 1

If policy is closed row →
All will be misses = 11
No of misses = 11

Ans5: Yes, reordering can improve hit rate.
Ordering will be
(4,15), (4,6), (6,150), (7,150), (4,150), (3,9), (4,9), (4,12), (5,10), (6,10), (11,10)

Ans6: The size of row-buffer should be high or large.
As due to spatial locality, we need to store next continous memory location values.

Ans7: Not always a yes or No.
Branch predition can reduce energy consumption if most of the prediction made are correct.
And energy consumption will be more if most of predictions are wrong

Ans8: Predection will be 10 + 4 + 19 - 2 + 2
= 33.70

so, it will predict → branch taken

now, Branch taken

updated weights ⇒ [11, −5, 20, −1, 3]

next prediction

= 11 + 5 + 20 − 1 + 3

= 38 > 0

predict → taken.

Ans 9- Voltage scaling will increase soft error vulnerability because charge (critical charge) for flipping bit will be reduced.

Ans 10- Example →

```
ld   r1, 4 [r2]
st   r1, 12 [r3]
```

where r_1, r_2 & r_3 are registers

Ans 11- sequence →

```
sub  r2, r1, r3
add  r1, r7, r8
```

This result in WAR hazard

Ans 12- example sequence

```
beq  .b1
mov  r1, 7
add  r3, r8, r5
:
:
.b1:
add  r8, r2, r1
```

Ans13- No, there forwarding path can lead to load-use hazard, thus, we need to add a pipeline bubble to avoid it & to make system work properly.

Ans14) Pipeline.

Cycles →	1	2	3	4	5	6	7	8	9	10
IF	(1)	(2)	(3)							
OF		(1)	(2)	(3)	(3)	(3)				
EX			(1)	(2)	(*)	(*)	(3)			
MA				(1)	(2)	(*)		(3)		
RW					(1)	(2)			(3)	

Here (*) is a bubble & (1), (2) & (3) are the instructions assigned respectively.

Ans15)

Request	Time Arrival	Open Row	Closed-Row
X_0	0	40 ns	40ns
Y_0	10	100 ns	100ns
X_1	100	160 ns	160ns
X_2	200	220 nS	220ns
Y_0	250	310 ns	290ns
X_3	300	370 ns	380ns

Ans16- Loosely coupled → If multiple unrelated programs are running in parallel.
Uses message passing technique typically for thread communication.
Strongly coupled → Set of programs running in parallel those share data, code, file & network connections.
Shared memory can be used by threads.

Ans17- No. of chips = 4×8 = 32
(Since each has size 1 megabit)

time for only 1 cell (single) = 10^{-7} s

& time of 1 chip = 10^{-1} s
4 chips can do, so after 8 rounds of refreshing, thus 0.8 s.

Ans18 i) Pipline.

Simple → Cycle →

fetch	Decode	add
	fetch	decode mul
		fetch decode add
		fetch decode sub
		fetch decode load

2) Instructions are given is order

2) For Superscalar →
Cycles →

1)	fetch	decode	add
2)		fetch	decode mul
3)		fetch	decode add
4)		fetch	decode sub
5)			fetch decode load

↳ ~~Instructions in order.~~

3) Out of order → Cycles.

fetch	decode	add	– → (add r10, r10, 5)
fetch	decode	add	– → (r8, r8, r2)
fetch	decode	sub	–
fetch	decode	–	mul
fetch	decode	–	load

↳ As given in order respectively

Ans 19) Coarse Grained Multithreading should be used

Ans 20) V. comp vr1, 50
V.gt.mul vr1; vr1, 50
vr1 → vector register

Ans 21) Since, chips provide parrallel machanism for each & every chip got 8-bit wide data, to provide 64-bit wide data.
Thus, 8 chips correspond to 1 rank
⇒ 32 chips in 4 ranks.
Total Capacity of DRAM = 32 × 16 G bit
$= 2^5 \times 2^4 \times 2^{30}$
$= 2^{39}$ bits = 64 G Bytes.
DRAM.

Ans 22- A C Z V K L Z V C H

The reuse distance of C is 4.
(distinct elements).

Ans 23- Cache tiling / blocking

```
for ( r=0 ; r<R; r+=B)
  for (col = 0; col < C ; col++)
for ( to=0; to< M; to++)
 for ( ti=0; ti<N ; ti++)
 for ( i=0, i < k, i++)
 for (j=0; j<k ; j++)
   for( int row = r ; row < min(r+B, R) ; row++)
```

Output-fmaps[to][row][col] += weights[to][ti][i][j] *
Input-fmaps[ti][s*row+i][s*col+j]

Ans24. For B1 to be a biased branch, the value of N could be certainly very large (for N > 10, it will be taken more than 90% of time & the % will increase as N increases)
There will be no condition on array[i] element.

For B2 to be unbiased, N can be conditioned or can't be conditioned ~~as B2 branch will depend on~~
to be large for executing the branch B2, many times
However, the array values should be divisible by 20, around half the time that is.
roughly 50% of values should be divisible by 20.

Ans25. For spatial multi-bit errors we can use bit-interleaving so, single errors on different streams (arrays) can be corrected.
To address temporal multi bit errors, we do scrubing. Therefore, they should be corrected before they exceed ECC correction capacity.