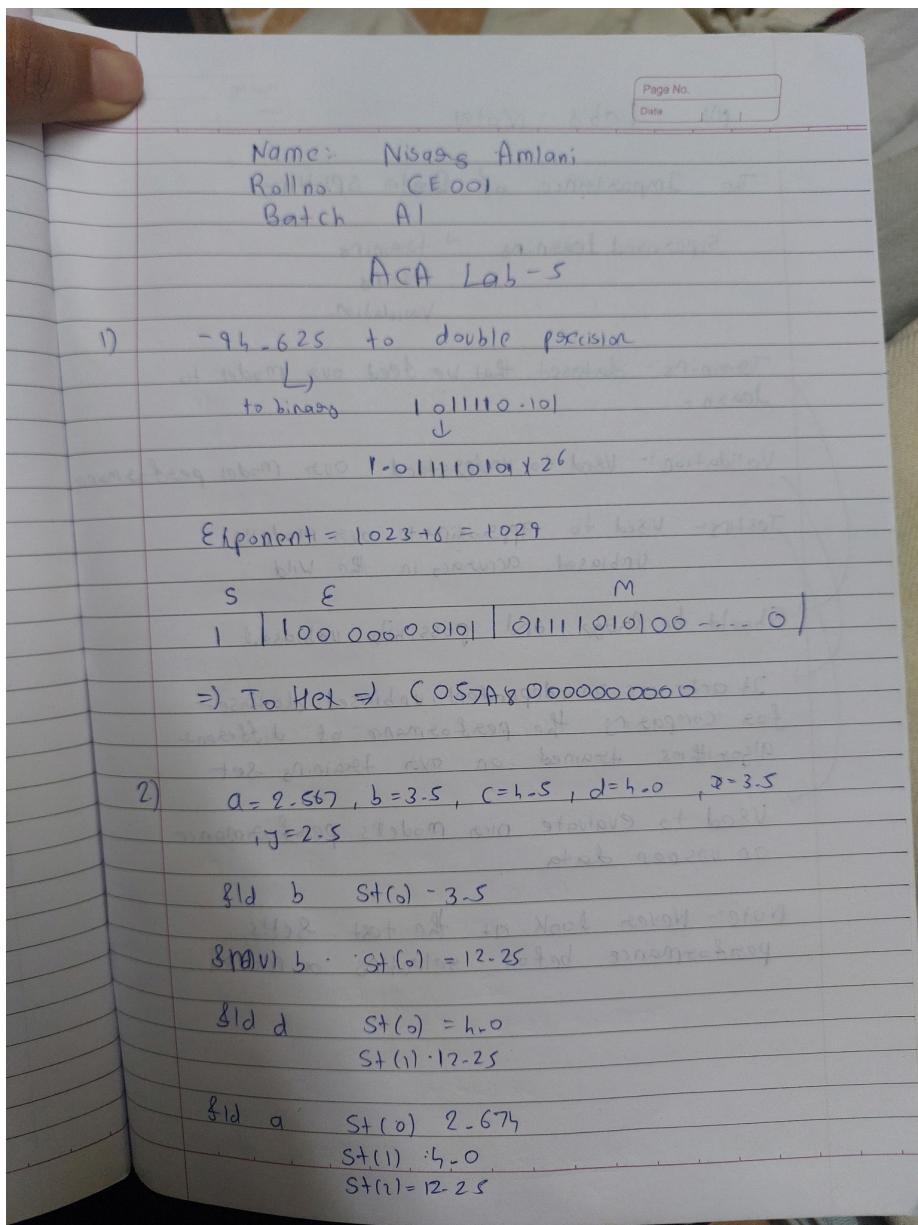


**Name : Nisarg .k.Amlani**  
**Roll:Ce001**  
**ID:22ceueg082**

**Lab :05**



$$\text{fmul } \cdot \text{St}(0) 10.699 \quad \text{St}(1) = 12.25$$

$$\text{fmulc } \text{St}(0) 48.1499$$

$$\text{St}(1) 12.258$$

$$\text{fsub } \text{St}(0) 35.899$$

$$\text{fsgn } \text{St}(0) 5.991$$

fig 3

$$(28.0 + 28.0 + 2) = 80$$

$$\text{fldx } \text{St}(0) 3.5 \quad 28.0 =$$

$$\text{fldy } \text{St}(0) 2.5$$

$$\text{fldz } \text{St}(1) 3.5 \quad 28.0 = \text{St}(0) + 28.0$$

$$\text{fly12x } \text{St}(0) 4.6267$$

$$\text{fld do } (\text{St}(0) - 3) = 28.0 - 3$$

$$\text{St}(1) 2.6267$$

$$\text{fsub } \text{St}(0) = 0.6267$$

$$\text{fxm } \text{St}(0) = 1$$

$$\text{St}(1) = 0.544 + 28.0 = 80$$

$$\text{fadd } \text{St}(0) = 1.544$$

$$\text{fldz } \text{St}(0) = 1$$

$$\text{St}(1) = 1.544 + 1$$

$$\text{fxch } \text{St}(0) = 1.544$$

$$\text{St}(1) = 1$$

$$\text{fscalc } \text{St}(0) = 3.088$$

$$\text{St}(1) = 1$$

3)

Ideal Speedup = 5  
 $VCB = 9y.$   
 $CB = 15y.$

Date

Branch Taken 85% 3 Cycles  
 Branch not Taken 15% 2 Cycles

a)  $VCB = 0.09 + 13 \cdot 0.85 + 0.15 + 0.27$

$$CB = (3 \cdot 0.85 + 2 \cdot 0.15) \\ = 2.85$$

$$\text{Speedup} = \frac{5}{1 + 0.27 + (2.85) \cdot 0.15} = 2.95$$

$$\% \text{ of loss} = \frac{5 - 2.95}{5} \cdot 100 = 41.2\%$$

$$= 41.2\% = 41.2$$

b)  $VCB = 0.09 + 1$

$$CB = (0.85 + 0.15) \\ = 0.1275$$

$$\text{Speedup} = \frac{5}{1 + 0.09 + 0.1275}$$

$$= 4.106$$

$$\% \text{ of loss} = \frac{5 - 4.106}{5} \cdot 100 = 18.08\%$$

- 17-881

$$Y_{\text{of gain}} = h_1 - 2 - 17 - 84$$

= 22.29% 19 19

948A 1000 110 0086 5-8

0-16 Correct

(c) BTB21 found in 91% → sum 111%  
BTB2 not found 8% → 0.08 Incorrect  
BTB2 not found 9% → 0.09 Correct

$$VCB = 0.04 \times 3 = 0.12$$

$$CB = (0.85 * 3) + (0.15 * 2)$$

$$T_2 = 850 - 803 = 2.85 \text{ m} \quad 0.4 \quad 0.1$$

$$CB \text{ delay} = 2.85^t - 0.0900A$$

• 0-2565m \$90.4 891

$$\text{Misprediction} = 0.91^+ 0.95^+ 0.85$$

Misprediction = 10 - 10375

Page 2 Page 3 Page 4 Page 5

$$\text{Total delay} = 0.09(0.12)$$

APP 2160 (0-2565)

$$= 0.6479$$

$$\text{Speed up} = \frac{5}{1+0.6479} = 4.695$$

$$\% \text{ of loss} = \frac{5 - 4.645}{5} * 100$$

= 67.

## L. Array

PE	PE	PE	PE	PB
LD <sub>1</sub>	LD <sub>2</sub>	LD <sub>3</sub>	LD <sub>4</sub>	LD <sub>8</sub>
Add <sub>1</sub>	Add <sub>2</sub>	Add <sub>3</sub>	Add <sub>4</sub>	Add <sub>8</sub>
MUL <sub>1</sub>	MUL <sub>2</sub>	MUL <sub>3</sub>	MUL <sub>4</sub>	MUL <sub>8</sub> $\Rightarrow$ 6 cycles
SUB <sub>1</sub>	SUB <sub>2</sub>	SUB <sub>3</sub>	SUB <sub>4</sub>	SUB <sub>5</sub>
DIV <sub>1</sub>	DIV <sub>2</sub>	DIV <sub>3</sub>	DIV <sub>4</sub>	DIV <sub>5</sub>
ST <sub>1</sub>	ST <sub>2</sub>	ST <sub>3</sub>	ST <sub>4</sub>	STS <sub>8</sub>

Vector:  $(S0 \cdot 0 + S1 \cdot 0) = 80V$   
 $(S2 \cdot 0 + S3 \cdot 0) = 90V$

LD	ADD	MUL	SUB	DIV	ST
LD <sub>1</sub>					
LD <sub>2</sub>	ADD P <sub>0,0</sub> + S <sub>2,0</sub>				
LD <sub>3</sub>	ADD P <sub>0,0</sub> + S <sub>3,0</sub>				
LD <sub>4</sub>	ADD P <sub>0,0</sub> + S <sub>1,0</sub>	MUL S <sub>0,0</sub>	SUB <sub>1</sub>		
LD <sub>5</sub>	ADD P <sub>0,0</sub> + S <sub>0,0</sub>	MUL S <sub>1,0</sub>	SUB <sub>2</sub>	DIV <sub>1</sub>	
		ADD P <sub>0,0</sub> + MUL <sub>3</sub>	SUB <sub>3</sub>	DIV <sub>2</sub>	ST <sub>1</sub>
		MUL S <sub>2,0</sub>	SUB <sub>4</sub>	DIV <sub>3</sub>	ST <sub>2</sub>
			SUB <sub>5</sub>	DIV <sub>4</sub>	ST <sub>3</sub>
				DIV <sub>5</sub>	ST <sub>4</sub>
					STS <sub>8</sub>

10 cycles

## Ex1A : Hello World Program

```
#include<stdio.h>

int main()
{
    int ID = 0;
    printf("hello(%d)", ID);
    printf("world(%d)", ID);

}
```

```
● └─(nisarg@fedora)-[~/Documents/pgms]
$ gcc -fopenmp 1.c

● └─(nisarg@fedora)-[~/Documents/pgms]
$ ./a.out
hello(0)world(0)
```

## Ex 1B : Write a multithreaded program that prints “hello world”

```
#include "omp.h"
#include <stdio.h>

int main()
{
    #pragma omp parallel
    {
        int ID = omp_get_thread_num();
        printf("hello(%d)", ID);
        printf("world(%d)", ID);
    }
}
```

```
hello(0)world(0)
hello(8)hello(1)hello(4)world(4)
hello(3)world(3)
hello(7)world(7)
world(8)
world(1)
hello(5)world(5)
hello(2)world(2)
hello(11)world(11)
hello(9)world(9)
hello(6)world(6)
hello(10)world(10)
```

## Ex 1C : Write a multithreaded program to perform addition

```
#include "omp.h"
#include <stdio.h>

int main()
{
    int sum = 0;
    int arr[100];
    for(int i = 0 ;i<100 ;i++)
    {
        arr[i] = 1;
    }

#pragma omp parallel
{
    int ID = omp_get_thread_num();
    for(int i = 0; i<100;i++)
    {
        sum += arr[i];
        printf("Sum = (%d)\n",sum);
    }
}
```

}

