CSE341

Assignment 01

Name : Umme Abira Azmary

ID : 20101539

Section : 04

Ans. to the ques. no-01 :

The term "performance via prediction" refers to the processor studing users screen time behaviour and starts execution of most ~~usable~~ Preferable applications ahead of time to speed up processor's performance. However, if the prediction is incorrect, the processor discards the program with a slide delay to execute new processes. The delay is very small and user's can't even understand its impact. As most of the time CPU's prediction is usually correct, this speeds up the overall performance of processor.

Ans. to the ques. no-02:

(a) Amdahl's law emphasizes the importance of improving the overall execution time of a program by optimizing the execution time of the most frequently used or time-consuming part: To be specific, this law explains the overall improvement of a program's performance can be gained by optimizing the execution time of a single part of a system. For example: if a program

total execution time is 200sec in which the addition itself takes 120sec to execute and the rest of the operations use 80 sec, improving the execution time of addition will improve the overall performance of the program.

(b) The Amdahl's law is related to the 'Make the common case faster' design principle,

The reason is, common cases takes most of the time while executing a program and so improving the execution time of common cases automatically improves the overall performance time. For example, while executing an arithmatic operation, the addition is the most used operation, whereas loading a value from an array is not most executable one. For this reason, if the addition takes 70sec to execute in a 100sec program, the improvement of addition's execution time may improve the overall performance.

Ans. to the ques. no- 036:

Based on the ~~othe~~ given scenario, the pricing strategy is effecient to make a profit.

As it is mentioned that, ~~for~~ 1570 die the base cost was 5.595 and after yield adjustment the price goes upto ~~5.595~~ 7.83 and with addition 5 dollar for profit, they should not face any loss. The reason why they are still facing is maybe because of their labour or machines cost which is not mentioned here.

Ans. to the ques. no - 4:

(a) 18

(b) Add $8 \times 2 = 16$ cc

sub $5 \times 3 = 15$ cc

mul $3 \times 4 = 12$ cc

addi $2 \times 5 = 10$ cc

53 cc.

∴ CPU time = cc × Clock Period

$= 53 \times 3$

$= 159$ sec.

(c) Clock rate $= \frac{1}{3}$

$= 0.333$ Hz.

(d) In this scenario, I prefer to choose the Add instruction to speed up the execution of the system as the Add instruction is used most numbers of time in the program. Speeding the execution time of Add instruction automatically decreases the overall system's execution time and makes the system faster.

(c) From b, the execution time is 159 sec.

Reducing 1.2 times, the execution time is = 132.5 sec

$T_{affected} = 48$ sec.

$T_{unaffected} = (15+12+10) \times 3 = 111$ sec.

$$T_{improved} = \frac{T_{affected}}{n} + T_{unaffected}.$$

$$\Rightarrow \frac{159}{1.2} = \frac{48}{n} + 111.$$

$\Rightarrow n = 2.23$.

Here, improvement factor of 2.23 is required.

Ans. to the ques. no-05:

The opcode field is used to define the format of an instruction. While writing a machine code from assymbly, the opcode field defines which type of instruction need to be performed. For instrance, ADD x5, x6, x7 → here the opcode field holds the machine code value of 0110011 to indicate that R-type of instruction needs to be performed.

Ans. to the ques. no- 06:

To convert LD x9, 10[x21] to its equivalent RISC-V machine code, and store it, the machine will understand the size of the data being loaded into the register by the funct3 field.

The funct3 field determines the type of load operation and size based on the data. Here, LD defines Load DoubleWord and so in funct3 field 011 value will be stored.

Ans. to the ques. no-07:

Program counter is the register that holds the address of the current instruction being executed. Program counter holds a great importance as it is responsible to maintain the executable instruction's position. For instance:

Address	Instruction
0X1000	addi X1, X0, 5.
0x1004	jal x2, ~~0exit~~ 11.

Here, PC = 0X1000 points to the first instruction.

after executing the first instruction, PC increments to 0x1004 for execution.

Ans. to the ques. no-08:

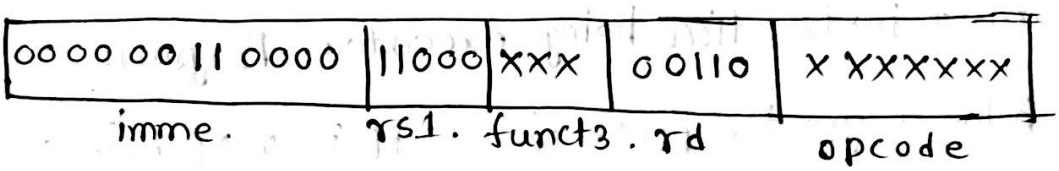
```
ld x5, 24(x20)
ld x6, 48(x20)
bne x5,x6,yes1.
slli x29, x6, 3
sd x29 , 48(x20)
beq x0, x0, exit
yes1 :
beq x5, x10, yes2
srli x28, x6, 4
sd x28, 48 (x20)
beq x0,x0,exit
yes2:
addi x27, x5, 2
sd x27, 24(x20)
exit
```

A = x20
B = x21

(b) There are no R-type instructions.

For I-type instruction:

ld x6, 48[x20]



For S type: sd x29, 48(x20).