Not checked by me - Bhaskar

Tutorial 3

1\. a) # Assume $a0 is a[0], $a1 is b[0]
$t0 has i, $t1 has k, $t2 has N, $s0 has c.
$a2 is the max length of arrays 'a' and 'b'
$a3

```
bge $a2, $a3, a3-lower
a2 lower:
sll $a4, ~~$zero,~~ $a2, 2
j a4-set
a3 lower:
sll $a4, ~~$zero,~~ $a3, 2
a4-set:
~~or $t0, $zero, $t1~~     # Set i = k
sll $t0, $t1, 2            # i = 4k ; sll $t3, $t2, 2
                                      add $a2, $a0, $t0
                                      add $a3, $a1, $t0
loop-begin:
~~slt a $t2, $t2, $t2~~  #
bge $t0, $t3, end-loop
sltu $t4, $t0, $a4
beq $t4, $zero, end-loop

lw $t5, 0($a2)
add $t5, $t5, $s0
sw $t5, 0($a2)

addi $a2, $a2, 4
addi $a3, $a3, 4 ; addi $t0, $t0, 4 ; j loop-begin
end-loop:
```

b) Optimization already performed in the code above.

c) I can increment $a2 and $a3 by 4 after loading and storing to prepare for the next loop iteration

d) Both can be used, getting rid of the addi instructions in the end

2\. a) or, sll, add, add, srl → 5 R formats

loop → bge
slt u
beq
2 memory ops
add ×4
jump

⇒ 3 branch, 2 memory and 5 R format instrs
⇒ (5 R format) + (3 branch + 2 memory + 5R) × (N−k)

b) Using lw-inc and sw-inc, we get 3R ... The second set of brackets c)

3. a) No: of clock cycles Taken = 10^{10} clock cycles.
Finally, we have 9×10^{10} clock cycles

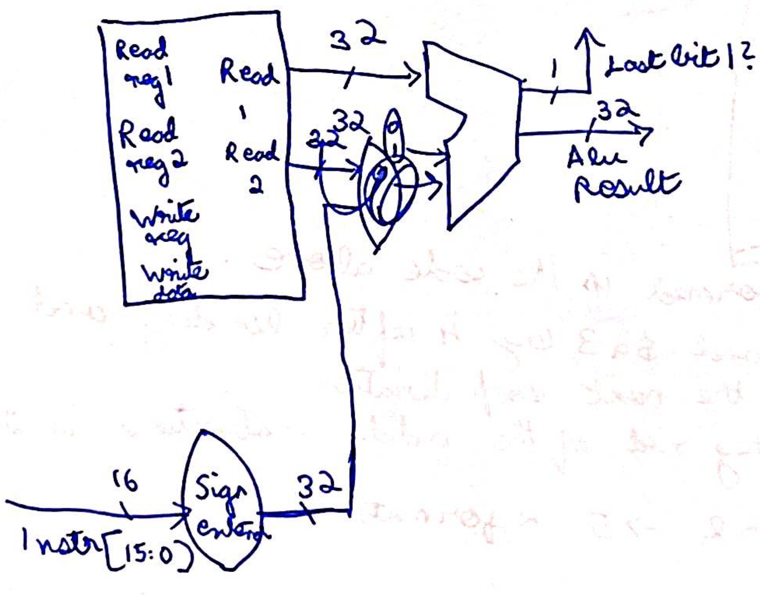
$\Rightarrow$ $\underline{5 \times 10^8}$ mult operations were replaced.

b) Let every other line in the code correspond to a single clock cycle
$\Rightarrow$ 8×10^9 instructions. Now we have 9×10^9 instead of 8.5×10^9 instructions $\Rightarrow$ $\frac{90}{85} = \boxed{\frac{18}{17}} \Rightarrow \boxed{\frac{1}{17}\,\%}$

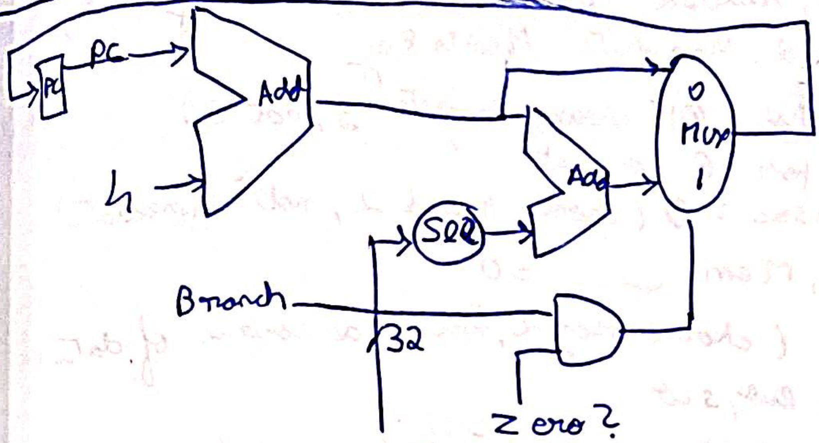
4. a) != is not defined between floats with infinite precision. Instead, consider the absolute value of (est × est − 2) and bound it below, say, 10^{-5}.

b) Newton - Raphson iteration

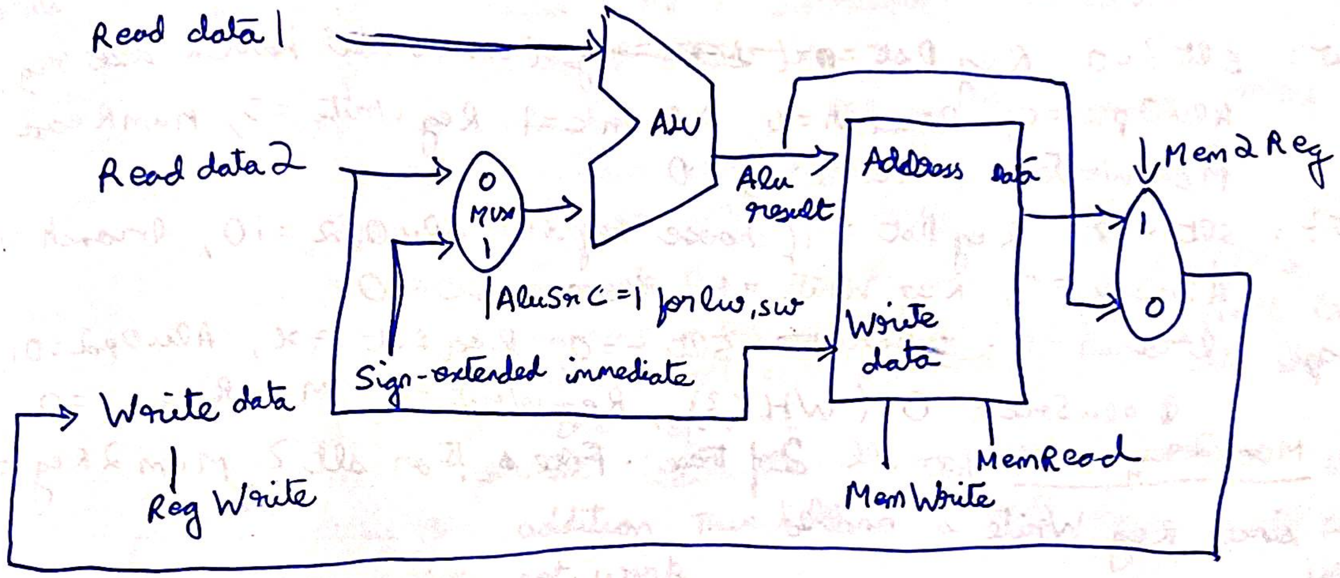
5.



Handling beq:



Handling lw, sw:



Put together: