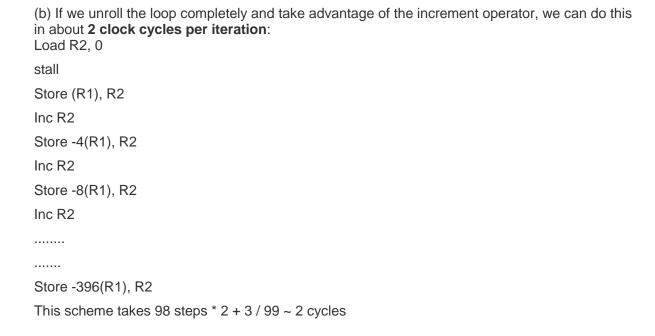Question 1

1A

Reaching Definitions

Ans: Forward

1B

Available Expressions

Ans: Forward

IC

Live Variable Analysis

Ans: Backward

ID

Copy Propagation

Ans: Forward

Question 2

2A

Reaching Definitions

Ans: Union

2B

Available Expressions

Ans: Intersection

2C

Live Variable Analysis

Ans: Union

2D

Copy Propagation

Ans: Intersection

Question 4

4A

Ans: x

4B

Ans: x

4C

Ans: x

4D

Ans: x

4E

Ans: x, y

4F

Ans: x, y, z

4G

Ans: x, y, z

4H

Ans: z

Question 9

(a)
store 0(R1), 0
Loop:
Load R3 0(R1)
stall
Inc R3
Store -4(R1) R3
Add R1, R1, -4
stall
Jne R1, R0, Loop
stall

For unrolled code we get about **6 clock cycles per iteration.**

(b) If we unroll the loop completely and take advantage of the increment operator, we can do this in about **2 clock cycles per iteration**:

Load R2, 0

stall

Store (R1), R2

Inc R2

Store -4(R1), R2

Inc R2

Store -8(R1), R2

Inc R2

....

....

Store -396(R1), R2

This scheme takes 98 steps * 2 + 3 / 99 ~ 2 cycles

We can also do an alternate scheme which does not unroll the loop completely, and unrolls it 4 at a time, this gives **3 clock cycles per iteraion.**