1. Consider the code segment in RISC-V:

```
Loop:   lw      x1, 0(x2)       ; load X1 from address 0+x2
addiw   x1, x1, 1               ; x1 = x1 + 1
sw      x1, 0(x2)               ; store x1 at address 0+x2
addiw   x2, x2, 4               ; x2 = x2 +4
subw    x4, x3, x2              ; x4 = x3 – x2
bne     x4, xo -24              ; branch to loop if x4!= 0
```

Assume the initial value of x3 is x2 + 396.

a)

ANS.

```
nt x2;              // initialize X2
x3 = x2+396;        //As mentioned at the end of give code
do
{
x1 = x2;            //load value found at address x2 which means x1 will
get value of x2
x1 = x1+1;          //add 1 to x1 and assign value to x1
x2 = x1;            //store the value at 0(x2) which means assign x2 the
value of x1
x2 = x2+4;          //add 4 to x2 and assign the result to x2
x4 = x3-x2;         //subtract x2 from x3 and assign the result to x4
}while(x4!=0)       //check condition and if true branch else exit
```

b)

ANS.

The five stages are

1. Instruction fetch cycle (IF).

2. Instruction decode/Register fetch cycle (ID).

3. Execution/Effective address cycle (EX).

4. Memory access (MEM).

5. Write -back cycle (WB).

Instructions	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18
Loop: LD x1, 0(x2)	IF	ID	EX	DM	WB													
ADDI x1,x1 #1		IF	S	S	ID	EX	DM	WB										
SD x1, 0(x2)					IF	S	S	ID	EX	DM	WB							
ADDI x2,x2, #4								IF	ID	EX	DM	WB						
SUB x4,x3, x2									IF	S	S	ID	EX	DM	EB			
BNEZ x4, Loop												IF	S	S	ID	EX	DM	WB
LD x1, 0(x2)																IF	ID	EX

c)

ANS.

First 98 iterations take 15 cycles

Last iterations take 18 cycles

So Total = 15 x 98 + 18 x 1

= 1488

d)

ANS.

Instructions	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17
Loop: LD x1, 0(x2)	IF	ID	EX	DM	WB												
ADDI x1,x1 #1		IF	S	ID	EX	DM	WB										
SD x1, 0(x2)				IF	ID	EX	DM	WB									
ADDI x2,x2, #4					IF	ID	EX	DM	WB								
SUB x4,x3, x2						IF	ID	EX	DM	WB							
BNEZ x4, Loop							IF	ID	EX	DM	WB						
LD x1, 0(x2)									IF	ID	EX	DM	WB				

e)

ANS.

First 98 iterations take 8 cycles

Last iterations take 11 cycles

So Total = 8 x 98 + 11 x 1

= 795

f)

ANS.

Instructions	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17
Loop: LD x1, 0(x2)	IF	ID	EX	DM	WB												
ADDI x1,x1 #1		IF	S	ID	EX	DM	WB										
SD x1, 0(x2)				IF	ID	EX	DM	WB									
ADDI x2,x2, #4					IF	ID	EX	DM	WB								
SUB x4,x3, x2						IF	ID	EX	DM	WB							
BNEZ x4, Loop							IF	ID	EX	DM	WB						
DELAY BRANCH											SLOT						
LD x1, 0(x2)									IF	ID	EX	DM	WB				

g)

ANS.

Rewritten Code:

```
lw x1, 0(x2)            ; load X1 from address 0+x2
loop: addiw x1, x1, 1   ; x1 = x1 + 1
sw x1, 0(x2)            ; store x1 at address 0+x2
addiw x2, x2, 4         ; x2 = x2 +4
subw x4, x3, x2         ; x4 = x3 – x2
bne x4, xo -24          ; branch to loop if x4!= 0
lw x1, 0(x2)            ; load X1 from address 0+x2
```

h)

ANS.

Instructions	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17
LD x1, 0(x2)	IF	ID	EX	DM	WB												
ADDI x1,x1 #1		IF	S	ID	EX	DM	WB										
SD x1, 0(x2)				IF	ID	EX	DM	WB									
ADDI x2,x2, #4					IF	ID	EX	DM	WB								
SUB x4,x3, x2						IF	ID	EX	DM	WB							
BNEZ X4, Loop							IF	ID	EX	DM	WB						
LD x1, 0(x2)								IF	ID	EX	DM	WB					

i)

ANS.

```
LD x1, 0(x2)              ; load x1 from address 0+x2
Loop: DADDI x2, x2, #4    ; x2 = x2 +4
DADDI x1, x1, #1          ; x1 = x1 + 1
 SD x1, -4(x2)            ; store x1 at address -4+x2
DSUB x4, x3, x2           ; x4 = x3 – x2
BNEZ x4, Loop             ; branch to loop if x4!= 0
LD x1, 0(x2)              ; load R1 from address 0+x2
```

j)

ANS.

Instructions	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17
LD x1, 0(x2)	IF	ID	EX	DM	WB												
Loop: ADDI x2,x2, #4		IF	ID	EX	DM	WB											
ADDI x1,x1 #1			IF	ID	EX	DM	WB										
SD x1, -4(x2)				IF	ID	EX	DM	WB									
SUB x4,x3, x2					IF	ID	EX	DM	WB								
BNEZ x4, Loop						IF	ID	EX	DM	WB							
LD x1, 0(x2)							IF	ID	EX	DM	WB						
Loop: ADDI x1,x1, #1								IF	ID	EX	DM	WB					

k)

ANS.

First iteration takes 7 cycles

Next 97 iterations takes 6 cycles

Last iterations take 10 cycles

So Total = 7 x 1 + 6 x 97 + 10 x 1 = 599

l)

ANS.

Speedup of k over c = 1488/599

= 2.48

a speed up by a factor of 2.48

Speedup of k over e = 795/599

= 1. 327

a speedup of 32.7%

2. Computer C360 is built with no pipelining in single cycle of 7 ns: IF 1 ns, ID 1.5 ns, EX 1 ns, MEM 2 ns, and WB 1.5 ns. Designers consider building C470 using a five stage pipeline based on C360 data.

a)

ANS.

Clock cycle time = Maximum of delay of all instruction

maxOf (1, 1.5, 1, 2, 1.5)

= 2ns

b)

ANS.

After every 4 instructions in a 5 stage pipeline

CPI = 5/4 = 1.25

c)

ANS.

Speedup = CS360 execution time/CS470 execution time

= 7/2.5

=2.8

3. Consider the following code segment

```
fld f1, 0(x1)
fld f2, 0(x2)
fmult.d f3, f2, f1
fadd.d f3, f3, 1
addi x3, x3, 1
```

ANS.

Instructions	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22
LD F1, 0(x1)	IF	ID	EX	DM	WB																	
LD F2, 0(x2)		IF	ID	EX	DM	WB																
MUL.D F3, F2, F1				IF	S	ID	M1	M2	M3	M4	M5	M6	M7	DM	WB							
ADD.D F3, F3, #1						IF	S	S	S	S	S	S	S	S	ID	A1	A2	A3	A4	DM	WB	
ADD.D x3, x3, #1															IF	ID	EX	S	S	S	DM	WB

4.

ANS.

Execution time = I x CPI x Cycle time

So speedup = Execution time of 5 stage pipeline / Execution time of 12 stage

= (I x (6/5) x 1) / (I x (11/8) x 0.6)

= 1.45