//Απαντήσεις ερωτήσεων 1ου μέρους.

Ερώτηση 1. Η εκτέλεση του προγράμματος ολοκληρώνεται σε 40 κύκλους.

Ερώτηση 2.Υπάρχουν 16 κύκλοι καθυστερήσεων.

Ερώτηση 3.Οι κύκλοι που χάνονται λόγω κινδύνων ελέγχου είναι 3. 3 κύκλοι flush.

Ερώτηση 4.

Περίπτωση 1: η ori(ori $t0,$at,0) περιμένει για 2 κύκλους την lui ($at,0x00001001)να γράψει τον καταχωρητή at που η ίδια χρειάζεται.

Περίπτωση 2: η addi (addi $t0, $t0, 8) περιμένει την ori να τροποποιήσει τον καταχωρητή t0 που η ίδια χρησιμοποιεί, έχουμε καθυστέρηση 2 κύκλων.

Περίπτωση 3: η εντολή lw(lw $t3, 0($t0)) περιμένει για 2 κύκλους την εντολή addi (addi $t0, $t0, 8) να γράψει τον καταχωρητή t0 που η ίδια τον χρειάζεται.

Περίπτωση 4: η εντολή lw(lw $t2, 4($t0)) περιμένει για 2 κύκλους την εντολή addi (addi $t0, $t0, -8)να γράψει τον καταχωρητή t0 που η ίδια χρησιμοποιεί.

Περίπτωση 5: η εντολή beq(beq $t1, $zero, notTaken) περιμένει για 2 κύκλους το τέλος της lw(lw $t1, 0($t0)) αφού χρειάζεται την τιμή του t1 που τροποποιεί η lw.

Περίπτωση 6: η εντολή slt(slt $t6, $t4, $t5) περιμένει για 2 κύκλους την εντολή or (or $t5, $t3, $t2) να γράψει τον καταχωρητή t5 αφού χρειάζεται την τιμή του.

Περίπτωση 7: η εντολή beq (beq $t6, $zero, taken)περιμένει για 2 κύκλους την εντολή slt (slt $t6, $t4, $t5)να γράψει τον καταχωρητή t6 αφού χρειάζεται την τιμή του.

Περίπτωση 8: η εντολή addi(addi $t7, $t7, 2) περιμένει για 2 κύκλους την εντολή sub (sub $t7, $t2, $t1)να ολοκληρώσει το γράψιμο στον καταχωρητή t7 αφού τον χρειάζεται.

Ερώτηση 5. Στις περιπτώσεις 1,2,3,4,6,8 θα μπορούσαν να εξαφανιστούν πλήρως οι καθυστερήσεις αν υπήρχε η δυνατότητα της προώθησης. Η προώθηση θα γινόταν από την αρχή του σταδίου MEM στην αρχή του σταδίου ID και συγκεκριμένα σε μία από τις εισόδους της ALU.

Ερώτηση 6. Θα μπορούσε να μειωθεί στις περίπτωση 5 και 7.

Ερώτηση 7. Το πρόγραμμα θα χρειαζόταν 26 κύκλους ρολογιού.