Question 1

Use the following code fragment.

```assembly
loop: L.D F0,0(R2)
    L.D F4,0(R3)
    MUL.D F0,F0,F4
    ADD.D F2,F0,F2
    DADDUI R2,R2,#8
    DADDUI R3,R3,#8
    DSUBU R5,R4,R2
    BNEZ R5,loop
```

Assume the initial value of R4 is R2+792.

(a) Show the timing of this instruction sequence for MIPS pipeline without any forwarding and bypassing hardware. How many cycles does this loop take to execute?

(b) Show the timing of this instruction sequence for MIPS pipeline with normal forwarding and bypassing hardware. Assume the branch condition and target address are computed in the second stage (ID), and branch is handled by predicting not-taken. How many cycles does this loop take to execute?

(c) Assume MIPS pipeline with normal forwarding and bypassing hardware, and a single-cycle delayed branch. Schedule the code including the branch delay slot to minimize the total number of cycles for this loop. Write the scheduled code, and ashow its timing diagram. How many cycles does this loop take to execute in this case?

Question 2

Note: Page numbers refer to the 3rd edition of [HP] text.

Consider the example on Pages 305-308 of [HP] text (Chapter 4). Change the code to

```c
for (i=1000; i>0; i=i-1)
    x[i] = x[i] * s;
```

Follow the same pattern of the text example, discussing the effect of compiler scheduling, unrolling, and unrolling+scheduling. Assume FP multiplication unit has 6 stages. Hence the latency for an FP multiplication followed by another ALU operation that uses the result of multiplication is 5. Other latencies for FP multiplication are obtained in a similar manner (see Figure 4.1 on Page 304).

Question 3

Note: Page numbers refer to the 3rd edition of [HP] text.

First read the example on pages 309-311 of [HP] text (Chapter 4). Consider the following code segment
loop: L.D F2,0(R1)
     L.D F4,0(R2)
     MUL.D F6,F2,F4
     S.D F6,0(R1)
     DADDUI R1,R1,#8
     DADDUI R2,R2,#8
     BNE R5,R1,loop

(a) Unroll the loop 4 times. Keep the DADDUI instruction for each iteration, but keep only one BNE for the unrolled code. Also use the same registers in each loop copy. Show data dependencies involving registers R1 and R2 using arrows.

(b) Remove the DADDUI instructions, and adjust the offsets in L.D and S.D instructions accordingly. You still need two DADDUI instructions (one for R1 and one for R2) towards the end of the code.

(c) Indicate both the data dependencies and name dependencies within the body (using, for example, different colors for the arrows).

(d) Rename F registers, and write the code. Show how renaming eliminates name dependencies.

(e) Schedule the code to eliminate the stalls (or reduce them to the extent possible). Assume delayed branch. Calculate the total number of cycles for the execution of whole loop (1000 iterations).

10% of the assignment mark is for typesetting. Figures and special symbols can be drawn/written by hand.