Question 1. (4 points)

What is a stack in the context of MIPS procedure call?

What are the contents of a stack? Explain with the help of a neat figure.

Question 2. (2 points)

List two design principles behind the MIPS instruction set design. Explain each with example.

Question 3. (4 points)

Explain the difference between static and dynamic branch prediction.

Give examples of two dynamic branch prediction techniques. Briefly explain one of them.

Question 4. (3 points)

Broadly, there are three kinds of cache based on the way a block from memory is mapped to a location in cache. Name the 3 types with a very short description of each.

Question 5. (2 points)

Reduction is a programming technique used in parallel programming. Explain the technique briefly. You may use summation of numbers in an array as an example. It is not required to write code.

Question 6. (2 points)

What is meant by a bus in a computer system? What is the difference between a synchronous and an asynchronous bus?

Question 7. (2 points)

Who gets the control when there is a page-fault (with regards to virtual memory) is it the operating system or is it the hardware? Why?

For the following questions, Q8 to Q13, it is mandatory to write couple of lines of explanation to discuss your approach to solve the problem.

Question 8. (4 points)

Consider two different implementations of the same instruction set architecture. There are four classes of instructions, A, B, C, and D. The clock rate and CPI of each implementation are given in the following table.

	Clock Rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D
P1	2.5 GHz	1	2	3	2
P2	3 GHz	2	2	2	1.5

Given a program with 10^6 instructions divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D, what is the execution time for each implementation?

Question 9. (4 points)

The following problem deals with translating from MIPS to C/Java. Assume that the variables f, g, h, i, and j are assigned to registers $s0, $s1, $s2, $s3, and $s4, respectively. Assume that the base address of the arrays A and B are in registers $s6 and $s7, respectively.

```
sub    $t0, $s3, $s4
slli   $t1, $t0, 2
add    $t2, $t1, $s6
lw     $t0, 0($t2)
sw     $t0, 32($s7)
```

Write the final C/Java code. Show the intermediate steps you work out for each assembly statement. Hint: slli is the same as sll except that the second argument is a number.

Question 10. (3 points)

For the contents of registers $s1 = 0000 … 0001_b and $s2 = 1111 …1111_b$, what is the value of $t0 for the following assembly code? All registers are 32 bit long.

```
add    $t0, $s1, $s2
```

Is the result in $t0 the desired result, or has there been overflow?

Question 11. (4 points)

Consider the datapath shown in the figure. If we only have to support the load instruction (lw $t0, offset ($s0))

a) Which components are utilized by the load instruction?

b) What are the values of control signals (those signals that are shown in the figure) generated for this instruction?

c) What is the clock cycle time? Assume the following latencies for each block in the datapath.

I-Mem	ADD	Mux	ALU	Regs	D-Mem	Sign-Extend	Shift-left-2
200 ps	70ps	20ps	90ps	90ps	250ps	15ps	10ps
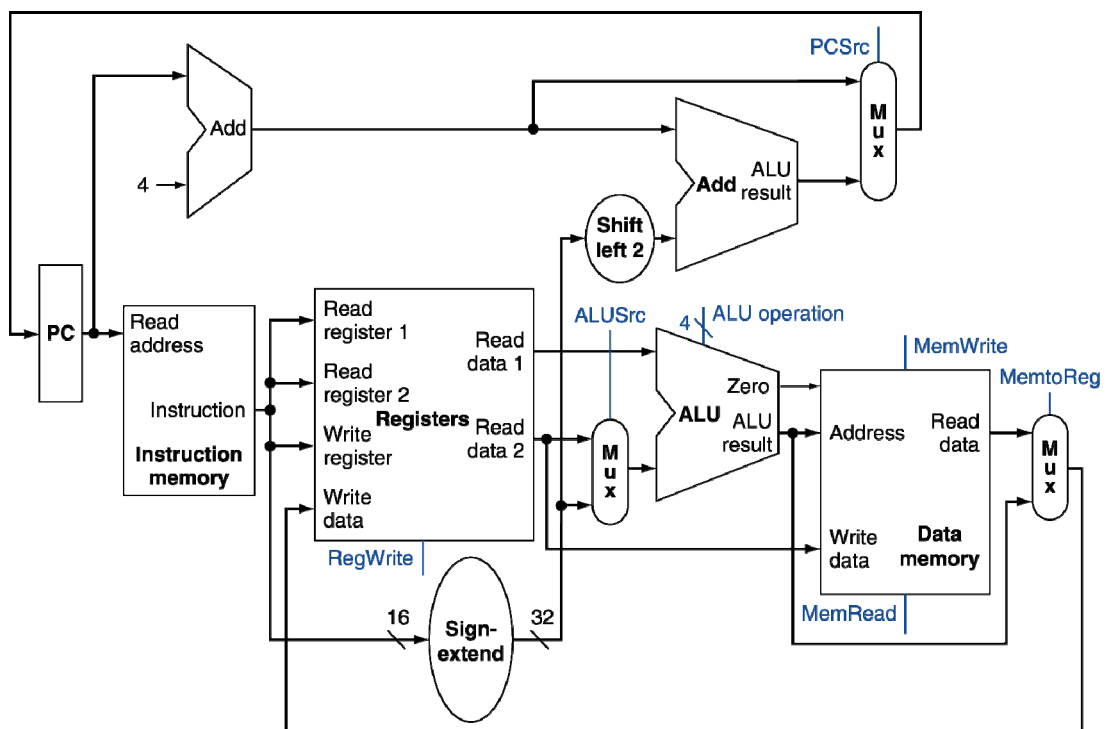
Question 12. (3 points)

Show the pipelined execution for following instructions. What kind of hazard is generated? What is the total number of cycles required to execute the instructions?

SW   R1, 16(R2)
LW   R3, 8(R4)
ADD   R4, R1, R4
SLLi   R5, R5, 2

Question 13. (3 points)

Consider the following code

for (int i = 0; i < 10,000,000; i++)
       sum += A[i];

Assume each element of A is 4 bytes and the variable *sum* is kept in a register. Consider the following cache configurations and choose the one that is best (minimum) in terms of cost, where cost is the product of cache size and miss rate. The configuration is given as
<Number of blocks/lines> :< Associativity> :< Block/line size>

a. <1024> :< 1> :< 32>
b. <512> :< 2> :< 64>
c. <512> :< 1> :< 64>
d. <2048> :< 1> :< 32>

Give reasons for your answer.