Qi:Answer the following questions briefly:

i) A half word 8295h is loaded into a 32-bit register. What would be the content of the register if the number is (a) signed, and (b) unsigned? (4)

ANS:

Binary (a) 1111 1111 1111 1111 1000 0010 1001 0101

(b) 0000 0000 0000 0000 1000 0010 1001 0101

Hexa

(a) FFFF8295

(b) 00008295

ii. Differentiate between ILP (Instruction Level Parallelism) and DLP (Data Level Parallelism. (4)

ANS:

Instruction-level parallelism (ILP) is a measure of how many of the instructions in a computer program can be executed simultaneously.

Data Level Parallelism (DLP): A data parallel job on an array of 'n' elements can be divided equally among all the processors.

iii. What are temporal and spatial locality observed in the behavior of programs? (4)

Two different types of locality have been observed.

Temporal locality states that recently accessed items are likely to be accessed soon.

Spatial locality says that items whose addresses are near one another tend to be referenced close together in time.

iv. What are the characteristic features of servers? (4)

First, availability is critical. Consider the servers running ATM machines for banks or airline reservation systems. Failure of such server systems is far more catastrophic than failure of a single desktop, since these servers must operate seven days a week, 24 hours a day.

A second key feature of server systems is scalability. Server systems often grow in response to an increasing demand for the services they support or an expansion in functional requirements. Thus the ability to scale up the computing capacity, the memory, the storage, and the I/O bandwidth of a server is crucial.

Finally, servers are designed for efficient throughput. That is, the overall performance of the server—in terms of transactions per minute or web pages served per second—is what is crucial. Responsiveness to an individual request remains important, but overall efficiency and cost-effectiveness, as determined by how many requests can be handled in a unit time, are the key metrics for most servers.

V. Why are benchmarks comprised of a large number of programs instead of a single program? (4)

To overcome the danger of placing too many eggs in one basket, collections of benchmark applications, called benchmark suites, are a popular measure of performance of processors with a variety of applications. Of course, such collections are only as good as the constituent individual benchmarks. Nonetheless, a key advantage of such suites is that the weakness of any one benchmark is lessened by the presence of the other benchmarks. The goal of a benchmark suite is that it will characterize the real relative performance of two computers, particularly for programs not in the suite that customers are likely to run.

vi. Why are RISC processors referred to as a Load/Store architecture? (4)

A load–store architecture is an instruction set architecture that divides instructions into two categories: memory access (load and store between memory and registers), and ALU operations (which only occur between registers). In RISC-V architecture consist of only load-store registers, all operations are performed on registers.

vii. Which addressing mode is used to encode branch target address, if the target is known at compile time? How is the address encoded? (4)

PC-relative addressing mode, the displacement from the target address is encoded in the instruction.

OR

What are the classes of computers according to Flynn's classification scheme?

4 categories of Flynn's classification of multiprocessor systems by their instruction and data streams:

Single Instruction, Single Data (SISD): SISD machines executes a single instruction on individual data values using a single processor.
Single Instruction, Multiple Data (SIMD): An SIMD machine executes a single instruction on multiple data values simultaneously using many processors.
Multiple Instruction, Multiple Data (MIMD): Each processor fetches its own instructions and operates on its own data.
MISD
MISD (Multiple-Instruction streams, Singe-Data stream): Each processor executes a different sequence of instructions.

viii. When does a phase ordering problem occur while using compiler optimization techniques? Give an example to illustrate. (4)

Enforced ordering of some optimization introduce phase ordering problem. Example: Problem including without knowing the size of the procedure.
[Page A-26 in book]
ix. Identify the operations performed (instruction) for each of the following expressions (4)

Regs [x1] $\leftarrow_{64}$(Mem [60 + Regs [x4]]$_0$ )$^{32}$## Mem [60+Regs[x4]]

LW R1,60(R4) Load word

Mem [500 + Regs(x2)] ← Regs [x3]$_{48..63}$

SH R3, 500(R2) Store half

Q2 (a) Execution time of a program P1 on M1 is given as 5 m Sec. The clock frequency of M1 is 3.24 GHz. What are the total number of instructions in Pl for the following information available about P1: (9)

| Instruction type | No. of cycles | Percentage |
|---|---|---|
| Type A | 2 | 22% |
| Type B | 1 | 27% |
| Type C | 4 | 17% |
| T3pe D | 5 | Remaining |

f = $3.24\times10^{9}$ GHz
Execution time = $5\times10^{-3}$ Sec
$CPI_{total}$ = ?
$CPI_{total}$ = (0.22 x 2) + (0.27 x 1) + (0.17 x 4) + ( 0.34 x 5)
= 0.44 + 0.27+ 0.68+ 1.70
$CPI_{total}$ = 3.09
CPU time (Execution time) = $\frac{IC \times CPI}{f_{clock}}$
$5 \times 10^{-3} = \frac{IC \times 3.09}{3.24\times10^{9}}$

IC = $\frac{5\times3.24\times10^{9-3}}{3.09}$

IC = $5.24 \times 10^{6}$ Insctructions

Q2 (b) The vector unit of a processor is enhanced to get a speedup of 10. What percentage of vectorization is required in applications to get an overall speedup of 2? (10)

Overall speedup = $\frac{1}{\{(1-F_e)+\frac{F_e}{Sp_e}\}}$

2= $\frac{1}{\{(1-x)+\frac{x}{10}\}}$

$2 - 2x + \frac{1}{5}\ x = 1$

$\frac{9x}{5} = 1$

$x = \frac{5}{9}$

= 0.556

Or 56 %

Q3: What are the desirable instruction set properties that help a compiler writer? (5)

Four properties in ISA

1. Regularity
2. Provide principle, not solution …(explain)
3. Simplify tradeoffs among alternatives. .(explain)
4. Provide instructions that bind the quantities known at compile time as constants. (explain)