1. (5%) Suppose we are building and supporting a real-time operating system for embedded systems. Which of the followings is not considered essential in our design?

   (a) To choose hard real-time and software real-time scheduling policies for applications

   (b) To develop real-time OS kernels

   (c) To estimate the overall system throughput to see if it matches the system requirements

   (d) To implement a parallel file system for applications

2. (5%) Which one of the following descriptions about garbage collection for storage management is correct?

   (a) The garbage collection problem is not addressed in Java language.

   (b) The garbage collection problem can be solved by tombstone schemes.

   (c) A hardware support for garbage collection with Java processors can help the performance of a java environment.

   (d) Almost every language, including C, C++, Java, Matlab, and Pascal, provides solutions to the garbage collection problem automatically for users.

3. (5%) Suppose that the head of a moving-head disk with 200 tracks, numbered 0 to 199, is currently serving a request at track 143 and has just finished a request at track 125. The queue of requests is kept in the FIFO order:

   86, 150, 92, 175, 95, 149, 4

   Choose one among the following algorithms to have the minimum amount of head movements to satisfy the above requests.

   (a) FCFS

   (b) SCAN

   (c) LOOK

   (d) C-SCAN

4. (5%) Explain the concept and functionality of "lseek operations" (normally used in Unix) for file systems?

5. (5%) Consider three different schemes in memory allocation, First-fit, Best-fit, and Worst-fit. Given memory partitions of 100K, 500K, 200K, 300K, and 600K (in order), and processes of 210K, 410K, 110K, and 420K (in order), which algorithm makes the best use of memory?

6. (a) (5%) For what two purposes is a long-term scheduler needed? (b) (5%) Give two situations in which a mid-term scheduler is needed.

7. (a) (6%) What is "deadlock prevention"? Explain it in detail. (b) (4%) Draw a diagram to show the relation among safe states, unsafe states, and deadlock.

8. (5%) If a solution to the critical section problem is correct, it must satisfy three necessary requirements. Show that one of the necessary requirements is not satisfied by the following solution, which uses the atomic instruction test-and -set. (In the solution, lock is false at the beginning.)

```
repeat
    while test-and-set(lock) do no-operation;          /* enter section */
          critical section
    lock = false;                                      /* exit section */
          remainder section;
until false;
```

9. (10%) Design a circuit, which adds two arbitrarily long, same-sized numbers. There is another input signal, ADD. When ADD=1, the two input numbers start entering your circuit and you should output the sum of them. When ADD=0, your circuit outputs 0.

10. (7%) In an 8-bit, load-store computer, arithmetic and logic operations use an accumulator as an implicit operand. Thus the instruction "ADD R1" means "accumulator ← accumulator + R1". The instructions may be 8-bit or 16-bit long. The 8-bit instructions have the following format: a 4-bit opcode followed by a 4-bit register number or an immediate number (depending on the opcode). The opcode of ADD is 0001 and that of ADDI (add immediate) is 1001. Suppose the memory location 89 of this computer contains the binary number 10011100. (a) If this memory location is considered to contain an instruction, then what instruction it is? (b) If it is interpreted as a data, and we would like to do an arithmetic-right-shift on it for 3 bits. What is the result? (c) If you were to design the instruction format for arithmetic right shift on a register, how you will do?

11. (8%) (a) Identity all possible RAW, WAR, and WAW hazards that are presented in the following code segment, if nothing is known about the instruction pipeline.

```
LD      AX, M[100]      ; AX ← M[100]
ADD     AX, BX          ; AX ← AX +BX
MOV     CX, 1           ; CX ← 1
ST      M[100], AX      ; M[100] ← AX
ST      M[200], BX      ; M[200] ← BX
ADD     CX, M[200]      ; CX ← CX + M[200]
```

(b) Suppose the code segment is executed on a pipeline with five stages: instruction fetch, operand fetch, memory read (assuming no need to calculate the effective address), ALU operation, and write back. Hardware will detect any hazard and stall the affected instruction. Draw a space-time diagram to show how the code segment is executed through this pipeline.

12. (7%) (a) Describe the format of a floating-point representation. (b) Show the single-precision IEEE 754 binary representation of the decimal floating-point number –1.25.

13. (10%) (a) What is polling? (b) What is interrupt-driven I/O? (c) What is DMA? (d) Try to compare them.

14. (8%) (a) The frequency of all loads and stores in gcc is 36%. Assume an instruction cache miss rate for gcc of 2% and a data cache miss rate of 3%. If a machine has a CPI (clock cycles per instruction) of 2 without any memory stalls and the miss penalty is 50 cycles for all misses, determine how much faster a machine would run with a perfect cache that never missed. (b) Suppose we double the clock rate of the machine, how much faster will the machine be with this faster clock, assuming the same miss rate as described above?