NOTE: If some questions are unclear or not well defined to you, you can make your own assumptions and state them clearly in the answer sheet.

1. (18% total) You are the lead designer of a new processor. The processor design and compiler are complete, and now you must decide whether to produce the current design as it stands or spend additional time to improve it. Please consider and answer the following problems.

1.1 (10%) You discuss this problem with your hardware engineering team and arrive at the following options:
(a) *Leave the design as it stands*. Call this base machine *Mbase*. It has a clock rate of 500 MHz.
(b) *Optimize the hardware*. The hardware team claims that it can improve the processor design to give it a clock rate of 600 MHz. Call this machine *Mopt*.
The following measurements have been made using a simulator for *Mbase* and *Mopt*.

Instruction class	Frequency	CPI	
		Mbase	*Mopt*
A	40%	2	2
B	25%	3	2
C	25%	3	3
D	10%	5	4

What are the CPI and MIPS (million instructions per second) rate for *Mbase* and *Mopt*? Copy Table 1-1 to your answer sheet and fill in the blanks with your answers.

1.2 (8%) The compiler team proposes to improve the compiler for the machine to further enhance performance. Call this combination of the improved compiler and the base machine *Mcomp*. The instruction improvements from this enhanced compiler have been estimated as follows.

Instruction class	Percentage of instruction executed vs. base machine
A	90%
B	90%
C	85%
D	95%

For example, if the base machine executed 500 class A instructions, *Mcomp* would execute 0.9 × 500 = 450 class A instructions for the same program. What is the CPI for *Mcomp*? How much faster is *Mcomp* than *Mbase*? Copy Table 1-2 to your answer sheet and fill in the blanks with your answers.

Table 1-1:

Machine	CPI	MIPS rate
Mbase		
Mopt		

Table 1-2:

Machine	CPI	Execution time *Mbase* / Execution time *Mcomp*
Mcomp		

2. (10%) Here is a loop in C:

```
Loop:    g = g + A[i];
         i = i + j;
         if (i != h)  goto  Loop;
```

Assume that A is an array of 100 elements and the variables g, h, i, and j are in registers $s1, $s2, $s3, and $s4, respectively. In addition, assume that the base of the array A is in $s5. Other registers that can be used are $t0 and $t1. A possible MISP assembly code corresponding to this C loop is showed as follows:

```
Loop:    add    $t1, $s3, OPA
         add    $t1, $t1, $t1
         add    $t1, $t1, OPB
         lw     $t0, 0(OPC)
         add    OPD, $s1, $t0
         add    $s3, $s3, $s4
         bne    $s3, OPE, Loop
```

Please determine proper values for the operands (OPA, OPB, OPC, OPD, OPE). Copy the Table 2 to your answer sheet and fill in the operand values.

Table 2:

Operand	OPA	OPB	OPC	OPD	OPE
Value					

3. (10%) Figure 1 shows a 16-bit adder consisting of four 4-bit ALUs using carry lookahead. Assume that $gi = ai \cdot bi$ (generate) and $pi = ai + bi$ (propagate). Determine the value of P0, P1, P2, P3, G0, G1, G2, G3, and C4 (CarryOut) when adding two 16-bit numbers 0001101000110011 and 1110010111101011 with Carry_In bit equal to 0. Please copy Table 3 to your answer sheet and fill in the blanks with your answers.

Table 3:

Signal	P0	P1	P2	P3	G0	G1	G2	G3	C4
Value									

4. (15% total) Compare the single-cycle implementation, in which all instructions take 1 clock cycle, with the five-stage (IF, ID, EX, MEM, WB) pipelined implementation using the following eight instructions: load word (lw), store word (sw), subtract (sub), and (and), or (or), set-less-than (slt) and branch-on-equal (beq). The operation times for the major functional units are 2 ns for memory access, 2 ns for ALU operation, and 1 ns for register file read or write.

4.1 (5%) Please complete the Table 4-1 by filling in the time for each component to calculate the total time of each instruction executed in the single-cycle implementation. Assume that the multiplexors, control unit, PC accesses, and sign extension unit have no delay. Please copy Table 4-1 to your answer sheet and fill in the blanks with your answers.

4.2 (5%) What is the clock cycle time for the single-cycle implementation and the pipelined implementation? Please copy Table 4-2 to your answer sheet and fill in the blanks with your answers.

4.3 (5%) Consider a program consisting of 100 lw instructions and in which each instruction is dependent upon the instruction before it. What would the actual CPI be if the program were run on the single-cycle implementation and the pipelined implementation with a forwarding unit and a hazard detection unit? Please copy Table 4-2 to your answer sheet and fill in the blanks with your answers.

Table 4-1:

Instruction class	Instruction fetch	Register read	ALU operation	Data access	Register write	Total time
Load word (lw)						
Store word (sw)						
R-format (add, sub, and, or, slt)	2 ns	1 ns	2 ns		1 ns	6 ns
Branch (beq)						

Table 4-2:

Design	Clock cycle time	Actual CPI
Single-cycle implementation		
Pipelined implementation		

5. (12%) Increasing associativity requires more comparators, as well as more tag bits per cache block. Assuming a cache of 4K blocks, a four-word block size, and a 32-bit address, find the total number of sets and the total number of tag bits for caches that are direct mapped, two-way and four-way set associative, and fully associative. Please copy Table 5 to your answer sheet and fill in your answers.

Table 5:

Cache	The total number of sets	The total number of tag bits
Direct mapped		
Two-way set associative		
Four-way set associative		
Fully associative		

6. (15%) Here is a series of address references given as word addresses: 1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17. Assuming a direct-mapped cache with four-word blocks and a total size of 16 words that is initially empty, label each reference in the list as a hit or a miss and show the final contents of the cache. Please copy Table 6-1 and Table 6-2 to your answer sheet and fill in your answers.

Table 6-1:

Reference	1	4	8	5	20	17	19	56	9	11	4	43	5	6	9	17
Hit or miss	Miss	Miss														

Table 6-2:

Block #	0	1	2	3
Starting address				

7. (15% total) Suppose we have a processor with a base CPI of 1.0, assuming all references hit in the primary cache, and a clock rate of 1 GHz. Assume a main memory access time of 100 ns, including all the miss handling.

7.1 (7%) Suppose the miss rate per instruction at the primary cache is 5%. What is the effective CPI for the machine with one level of caching?

7.2 (8%) If we add a secondary cache that has a 10-ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 2%. What is the effective CPI for the machine with the two-level cache?

8. (5%) Suppose you want to achieve a speedup of 50 with 100 processors. What fraction of the original computation can be sequential?
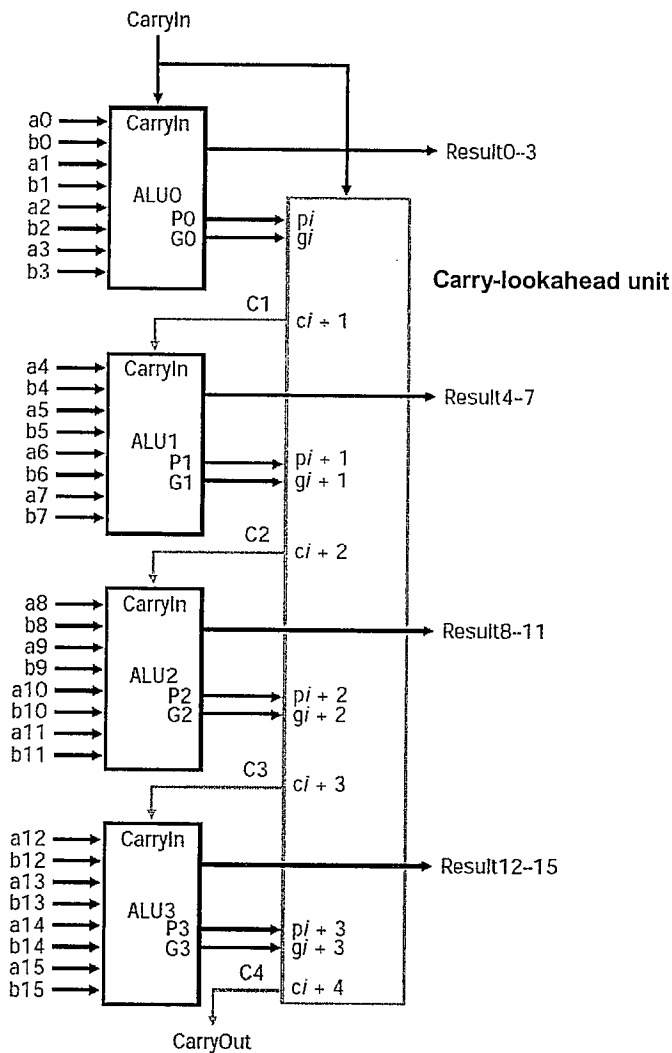

Figure 1

可能會用到電子計算機之計算式所得之數值會加註在題目後；如有需電子計算機計算之數值而題目後未列入，則請詳記列出計算公式與過程，此時之計算結果以計算式為答案，不因電子計算機之所計算數值而加以扣分。例如，題目提示 ln0.8 = -0.224, 但您的答案為 3ln0.2，或 3(ln0.1+ln2)，均為正確答案。

1. (15%) Solve the Differential Equation:
   $(2x+ye^{xy})dx + (xe^{xy})dy = 0, y(1)=0$
   (Hint: Is this an exact differential equation?)

2. (20%) Laplace Transform
   (a) (12%) Solve the following differential equation by Laplace Transform and Inverse Laplace Transform
   $$y''+2y'+10y=e^{-t} \sin t; y(0)=0, y'(0)=1$$
   (b) (8%) Let $f(t)=t$ and $g(t)=\cos t$. Find the convolution of f and g (i.e., $f*g$)

3. (25%) Matrices and Linear Systems
   (a) (5%) Given $A^{-1}=\begin{bmatrix} 1 & 2 & -1 \\ 3 & 4 & 2 \\ 0 & 1 & -2 \end{bmatrix}$, $B^{-1}=\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ -2 & 3 & 2 \end{bmatrix}$, find $(AB)^{-1}$
   (b) (10%) Solve the following linear system by using Gauss-Jordon elimination:
   $$\begin{cases} x_1 - x_2 + x_3 = 4 \\ 3x_1 + 2x_2 + x_3 = 2 \\ 4x_1 + 2x_2 + 2x_3 = 8 \end{cases}$$
   Please show step-by-step operations; otherwise, no points will be given.
   (c) (5%) Find Rank(A), $A=\begin{bmatrix} 3 & 1 & 4 & 0 \\ 1 & 0 & 1 & -2 \\ 2 & 1 & 3 & 2 \end{bmatrix}$
   (d) (5%) Find matrix A representing the linear transformation that maps (x_1, x_2) onto $(2x_1-5x_2, 3x_1+4x_2)$

4. (25%) Eigen value and matrix diagonization
   A finite state machine (FSM) has two states: state S1 and state S2. Let Prob(S1→S2) denote the probability that the machine moves from state S1 to state S2 in a cycle. Assume that the state transition probabilities are fixed in every cycle: Prob(S1→S1)=0.3, Prob(S1→S2)=0.7, Prob(S2→S1)=0.6, Prob(S2→S2)=0.4. Initially, the FSM is in state S1 at cycle 0.

   (a) (5%) Give a matrix A representing the state transition probabilities.
   (b) (5%) At the end of the second cycle, what is the probability that the FSM is in state S1? What is the probability that the FSM is in state S2?
   (c) (15%) At the end of the 1000th cycle, what is the probability that the FSM is in state S1? What is the probability that the FSM is in state S2?
   (Hint: You need to convert A to a diagonal matrix)

5. (15%) Fourier Series

Find the Fourier series of the function $f(x) = \begin{cases} -k, & -2 < x < 0 \\ k, & 0 < x < 2 \end{cases}$, and $f(x+4) = f(x)$.

※ *Please detail your answer for each question in the answer sheet.*

1. **Multiple-choice questions (total 20%; each subquestion deserves 4%):**

   There may be **ZERO or MORE correct answers** in each subquestion. If there is no correct answer, please give the answer "None".

   (1) Build a binary search tree for the input sequence 12, 5, 7, 18, 19, 15, 14, 6, 13. It is assumed that the tree root is on level 1. (A) There are five levels in the tree. (B) 14 is on level 3. (C) If we want to search 16, the number of required node comparisons is 3. (D) The subtree rooted at 15 has 3 nodes (including 15).

   (2) Which statement(s) is correct for an AVL tree? (A) The absolute value of the level difference of any two leaves is at most one. (B) The absolute value of the height difference of any two subtrees on the same level is at most one. (C) A deletion needs at most two rotation operations to preserve an AVL tree to be a height-balanced tree. (D) After a new node is inserted, the tree height will not increase if rotation operations are performed.

   (3) In a stack structure, let u denote a push operation and o denote a pop operation. After a sequence of stack operations, consisting of 3 pushes and 3 pops, an input sequence 123 may change its order. For example, after uouuoo is performed, 123 will become 132. Which sequence(s) cannot be obtained from input 123 by stack operations? (A) 213 (B) 231 (C) 312 (D) 321.

   (4) Which statement(s) is correct for binary search on a sequence? (A) The sequence must be sorted. (B) Binary search can be implemented by an array. (C) Binary search can be implemented by a linked list. (D) The hashing technique is an implementation of binary search.

   (5) Which statement(s) is correct for arithmetic expressions? (A) There is no parenthesis in a postfix expression. (B) The first symbol in a prefix expression must be an operator. (C) The first symbol in a postfix expression must be an operator. (D) The amount of operands in a prefix is more than that of operators.

2. **Printed results in C codes (total 10%; each subquestion deserves 5%):**

   What are printed in each of the following program segments?

   (1)
   ```
   int a[ ]={5,7,9,11,13};
   int *p;
   int i=2;
   p=a;
   *(p++) = ++i;
   printf("%d %d %d %d %d",a[0],a[1],a[i++],*p,*(p+1));
   ```

   (2)
   ```
   int b=26;
   printf("%d", ((b >> 3) << 2) & ((b <<2) >> 3));
   ```

3. **Two-way insertion sort (total 10%):**

   Suppose the output sorted sequence is increasing. The two-way insertion sort is a modification of the simple insertion sort (straightforward insertion sort), described as follows. After read the input elements, a separate output array a[0], a[1], a[2], ... a[n-1] is used to store the sorted sequence. This output array acts as a circular structure, that is, the right position of a[i] is a[i+1] if $0 \leq i \leq n-2$, and the right position of a[n-1] is a[0]. The first input element is put into a[0] initially. Once a contiguous group of elements are in the array, room for a new input element is made by shifting all smaller

elements one step to the left or all larger element one step to the right. The choice of left-shift or right-shift to perform depends on which would cause the smallest amount of shifts. Use the following 6 input elements to illustrate how this algorithm works. You have to show the content of the output array after each input element is inserted into the array.

27, 35, 43, 31, 29, 33.

4. Huffman tree (total 10%):

Huffman algorithm is a variable-length encoding method, in which a Huffman tree is constructed. Suppose seven symbols A, B, C, D, E, F, G with frequencies 8, 15, 1, 6, 15, 3, 7, respectively, are given. You have to obey the following rules for constructing the Huffman tree:

(1) The label of each leaf node is a single symbol. The label of each internal node is the concatenation of the labels of its left child and right child.

(2) When two nodes are merged, always set the node of label with less lexical order as the left child, and the other as the right child.

(3) If more than two nodes have the same least frequency, then you have to merge the two that have the least and the second least label in lexical order.

Please draw the full Huffman tree.

5. True-false questions (total 14%; each subquestion deserves 2%):

Please mark either 'O' or 'X' to respectively answer "yes" or "no" for each subquestion. Notice that you will get **-2 point** for each wrong answer until no point can be deducted from this question.

(1) The need-to-know principle dictates that programs, users, and even systems should be given just sufficient privileges to perform their tasks. This principle is very critical for system protection, which can prevent the mischievous, intentional violation of an access restriction by a user.

(2) A deadlock situation must arise if any of the following four conditions holds in a computer system: mutual exclusion, hold and wait, no preemption, and circular wait. Therefore, to vanquish the deadlock, we have to see what conditions cause that deadlock and then try to conquer each of them.

(3) We can realize a dynamic relocation by using a relocation register in memory management. For example, suppose that the CPU asks a logical address of 123 and the relocation register has a value of 1877. Then, the physical address will be 2000.

(4) A trap door may be included in a compiler. Therefore, the compiler would generate the standard object code as well as a trap door. However, this activity can be still found by conducting a very careful search of the source code of the compiled program.

(5) An attack that one participant in a communication pretends to be someone else is called the man-in-the-middle attack. Such an attack breaches the correctness of identification. A cracker can use this attack to gain access that he/she would not normally be allowed or escalate his/her privileges.

(6) RAID level 0+1 refers to a combination of RAID levels 0 and 1, where RAID 0 provides the performance, while RAID 1 supports the reliability. Generally speaking, this level provides better performance than RAID 5. It is common in environments where both performance and reliability are important.

(7) Location independence and location transparency are two related notions regarding name mappings in a distributed file system. A location-transparency naming scheme is a dynamic mapping, since it can map the same file name to different locations at two different times.

Therefore, location transparency is a stronger property than location independence.

6. **Choice questions (total 10%; each subquestion deserves 2%):**

Please select the **BEST ONE ANSWER** for each subquestion from the following box:

A.	page fault	B.	process	C.	page replacement
D.	worm	E.	virus	F.	authentication
G.	Trojan horse	H.	encryption	I.	class
J.	bnode	K.	demand paging	L.	inode
M.	thread	N.	cnode	O.	decryption

(1) The smallest unit of CPU utilization, which can be executed independently by OS

(2) The technique that loads pages only as they are needed

(3) A code fragment embedded in a legitimate program, which is designed to self-replicate & infect other programs

(4) The security technique that supports nonrepudiation

(5) A file-control block in most UNIX file systems

7. **Disk scheduling (total 6%):**

Suppose that you are given a disk queue with requests for I/O to blocks on cylinders 98, 183, 37, 122, 14, 124, 65, and 67. Let the disk head start at cylinder 53. Please show the disk scheduling result by

(1) The shortest-seek-time-first algorithm (3%)

(2) The SCAN algorithm (3%)

8. **Distributed synchronization (total 10%):**

(1) In a distributed system, deciding the event ordering is usually difficult. Why? (2%)

(2) Let A, B, C be events and '$\rightarrow$' denote the happen-before relation between two events. Please explain the three principles of the happen-before relation in a distributed system using the above notations. (6%)

(3) What is an atomic transaction? (2%)

9. **Memory management (total 10%):**

(1) What is a reentrant code? (2%)

(2) How does the CPU know whether a page is available or not from the page table? (2%)

(3) Suppose that it takes 20 nanoseconds to search the TLB (translation look-aside buffer) and 100 nanoseconds to access memory. If we want the effective memory-access time no longer than 122 nanoseconds, what is the hit ratio for the TLB? (6%)

There are 9 problems in this test. Note that you should write down detailed steps for the solution to each problem; otherwise, no credits for that problem will be given.

1. [9%] Ben works as a computer operator at National Sun Yat-sen University. One evening he finds that fifteen computer programs have been submitted earlier that day for batch processing. In how many ways can Ben order the processing of these programs if (a) [3%] there are no restrictions? (b) [3%] he considers five of the programs higher in priority than the others and wants to process those five first? (c) [3%] he first separates the programs into four of top priority, eight of lesser priority, and three of least priority, and he wishes to process the fifteen programs in such a way that the top-priority programs are processed first and the three programs of least priority are processed last?

2. [12%] A computer science professor has thirteen different programming books on a bookshelf. Six of the books deal with C++, the others with Java. In how many ways can the professor arrange these books on the shelf (a) [3%] if there are no restrictions? (b) [3%] if the languages should alternate? (c) [3%] if all the Java books must be next to each other? (d) [3%] if all the C++ books must be next to each other and all the Java books must be next to each other?

3. [12%] Four hundred coins numbered 1 to 400 are put in a row across the top of a cafeteria table. Four hundred students are assigned numbers (from 1 to 400) and are asked to turn over certain coins. The student assigned number 1 is supposed to turn over all the coins. The student assigned number 2 is supposed to turn over every other coin, starting with the second coin. In general, the student assigned the number n, for each $1 \le n \le 400$, is supposed to turn over every nth coin, starting with the nth coin. (a) [4%] How many times will the 400th coin be turned over? (b) [4%] Which coin(s) will be turned over as many times as the 400th coin? (c) [4%] Please show two coins that will be turned over more than 18 times?

4. [15%] Consider parking motorcycles and compact cars in a row of spaces where each cycle requires one space and each compact needs two. Find and solve a recurrence relation for the number of ways to park motorcycles and compact cars in a row of n spaces (a) [5%] if all cycles are identical in appearance, as are the cars, and we want to use up all the n spaces. (b) [5%] if the motorcycles come in two distinct models and the compact cars come in three different colors, and we want to use up all the n spaces. (c) [5%] if all cycles are identical in appearance and the compact cars come in three different colors, and empty spaces are allowed.

5. [12%] Consider the equation $c_1 + c_2 + c_3 + c_4 = 23$ where $-3 \le c_1$, $-2 \le c_2$, $-2 \le c_3 \le 0$, and $2 \le c_4$. (a) [4%] Find the generating function for the number of integer solutions to the equation. (b) [3%] The number of integer solutions to the above equation equals to the coefficient of x^n of the above generating function. What is n? (c) [5%] Please find the coefficient of x^{100} in the above generating function.

6. [9%] Determine the sequence generated by each of the following exponential generating functions. (a) [3%] $f(x) = 2e^{2x}$ (b) [3%] $f(x) = 5e^{3x} - 3e^{2x}$ (c) [3%] $f(x) = 1/(1-x) - x^2$.

7. [10%] For $n \geq 1$, let $X_n = \{1, 2, 3, ..., n\}$. $P(X_n)$ denotes the power set of X_n. (a) [8%] Let e_n be the number of edges in the Hasse diagram for the partial order $(P(X_n), \subseteq)$. Please find a recurrence relation for e_n and then solve the recurrence relation to get e_n. (b) [2%] Please find v_n, the number of vertices of the diagram in (a).

8. [11%] Please answer the following questions. (a) [4%] Find all generators of the cyclic group $(\mathbf{Z}_8, +)$. (b) [4%] Find all generators of the cyclic group $(\mathbf{Z}_5 - \{0\}, *)$. (c) [3%] If G is a cyclic group of order n, how many distinct generators does it have?

9. [10%] Please find an encryption function $E: \{1, 2, 3, ..., 26\} \rightarrow \mathbf{Z}$ such that $E(m_1) + E(m_2) = E(m_1 + m_2)$ for every m_1, m_2 in $\{1, 2, 3, ..., 26\}$ and please also find the decryption function corresponding to E.