國立臺灣師範大學資訊工程學系
九十九學年度第一學期
博士班資格考

考試科目：計算機結構
總分一百分
請在答案卷作答，在題目卷上作答不予計分

1. (15 分) Write the MIPS assembly code for the following fragments of C code:
   (a) f = (g + h) – (i + j)
   (b) if (i == j) f = g + h ; else f = g – h;

   Assume that variables *f, g, h, i* and *j* are stored in register $s0, $s1, …, $s4, respectively. You may also use other registers (such as $t0-$t9) for intermediate results. MIPS instructions you can use are *add, sub, addi, and, or, nor, andi, ori, sll, srl, beq*, *bne*, *slt, j*, and *jal*.

2. (10 分) A 16-bit unsigned integer is equal to decimal number 60000.
   (a) Which negative integer number does it represent if we interpret the same 16 bits as a signed integer using two's complement?
   (b) Give an example when overflow happens in 16-bit unsigned integers.

3. (15 分) In the IEEE 754 standard, a single precision floating-point number has a sign bit, 8 bits of exponent, and 23 bits of mantissa. It can represent fractions almost as small as $2.0 * 10^{-38}$ and numbers almost as large as $2.0 * 10^{38}$.
   (a) Give an example when underflow happens in a single precision floating-point variable.
   (b) NVIDIA has a "half" format that is similar to IEEE 754 except that it is only 16 bits wide, including a sign bit, 5 bits of exponent, and 10 bits of mantissa. What are the roughly the smallest fractions and the largest numbers that this "half" format can represent?

4. (10 分) A compiler designer is trying to decide between two code sequences for a particular computer. The hardware designers have supplied the following (Table on the left). For some particular C language statements, the compiler designer is considering two code sequences that require the following instruction counts (Table on the right).

	CPI for each instruction class		
	A	B	C
CPI	1	2	3

Code sequence	Instruction counts for each instruction class		
	A	B	C
1	2	1	2
2	4	1	1

(a) Which code sequence will be faster?
(b) What is the CPI for each sequence?

5. (15 分) The following program sequence is executed in a pipelined datapath with 5 pipeline stages (IF, ID, EX, MEM, and WB). The datapath has a forwarding unit and a hazard detection unit to resolve the data hazard problem.

```
add     $t1,$t1,$s2
lw      $s2,4($t1)
add     $t4,$s3,$s2
lw      $s5,4($t4)
sub     $t5,$s5,$s3
sw      $t5,4($t4)
add     $t4,$t4,$s2
```

At the first clock cycle, the first instruction (add $t1,$t1,$s2) enters the pipeline. It has no hazard with its preceding instructions. What is the instruction executed in each stage at the 7^{th} clock cycle? Explain the reason.

6. (12 分) The execution of computer instructions is subdivided into 5 stages, i.e. instruction fetch, decode, execute, memory read/write, and write back. Assume the execution time for each of these stages is 500ps, 200ps, 400ps, 500ps, and 200ps, respectively. If we implement it with the pipelined organization, a pipeline register with 40ps access time should be added between two consecutive stages.

(a) What is the achievable fastest clock frequency?
(b) What is the peak speedup when compared to the single cycle datapath?
(c) Each of the load-use hazard, branch instruction, and jump instruction needs to stall one clock cycle. A program is executed on this machine. In the instructions executed, 15% are load instructions, 15% are branch instructions, and 5% are jump instructions. 40% of the load instructions are followed by an instruction which uses the result. What is the actual CPI for executing this program?

7. (8 分) Explain how a superscalar processor is different from a conventional pipelined processor.

8. (15 分) Consider a computer has a 4-way set associative mapped cache with 32 two-word blocks. The replacement policy is LRU. A program is executed on this machine. The memory access sequence in word address is 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 3, 1, 2, 3, 7, 20, 21, 22, 23, 32, 33, 20, 0, and 1.

(a) Write down the final cache state. (An example cache state is shown below. You should make modifications according to the organization in this problem.)
(b) What is the cache hit rate for this program fragment?

Block Index	V	Tag	Data
0	1	1	Mem[0]
1	1	2	Mem[1]
…			
15	1	3	Mem[31]