1. Identify the data hazards in the following code, and fix them using *nop* (software).

```
sub $2, $1, $3
and $12, $2, $5
or  $13, $6, $2
add $14, $2, $2
sw  $15, 100($2)
```

The data hazard is present in the $2 register. The *sub* instruction will take three cycles to write to the $2 register, but the *and* and *or* instructions are attempting to access $2 before the register has a value in it. As a result, two *nop* instructions should be inserted after the *sub* instruction to offset the data access.

```
sub $2, $1, $3
nop
nop
and $12, $2, $5
or  $13, $6, $2
add $14, $2, $2
sw  $15, 100($2)
```

2. Identify and fix the data hazards shown below by showing the fowarding paths (hardware).

```
add $3, $4, $6
sub $5, $3, $2
lw  $7, 100($5)
%add £8, £7, £2
```

There are two data hazards in this set of instructions. In line two, register $3 is being accessed by the *sub* instruction before it is stored by the *add* instruction. The other hazard is with the $7 register, which is used by the *add* instruction before a value is loaded by the *lw* instruction.

3. Compare the performance for the single-cycle, multi-cycle, and piplined control using the following instruction mix.

	Loads	Stores	Branches	Jumps	ALU
Instruction Mix	25%	10%	11%	2%	52%
Clock Cycles	5	4	3	3	4

The performance for a single cycle machine is 200ps for memory access, 100ps for ALU operations, and 50ps for register read and write.

   (a) What is the clock cycle time for a single-cycle data path?

   (b) What is the average CPI for the multi-cycle design?

   (c) What is the average CPI for the piplined design?

   (d) What are the average instruction times for each designs?

4. Assume an instruction cache miss rate for a program is 2% and a data cache miss rate is 4%. If the processor has a CPI of 2 without any memory stalls and the miss penalty if 100 cycles for all misses, determine how much faster a processor with a perfect cache that never misses would be. The frequency of all loads and stores is 36% for the SPECint2000.