

Name: _____
ID: _____

COMP228
Quiz 3

Marks: _____/12
Winter 2008

1. Suppose the following memory information is known in a NSAM program.

Memory Address	Memory Content
1000	2000
2000	4000
4000	1000

Suppose initially $ebx = 2000$ and the symbol X corresponds to memory address 1000. Determine the result in register eax and the number of memory operand accesses in executing each of the following instructions:

- (i) `mov eax, X`
Answer: $eax = \underline{1000}$; # memory operand accesses = 0
- (ii) `move eax, [X]`
Answer: $eax = \underline{2000}$; # memory operand accesses = 1
- (iii) `mov eax, [ebx+X]`
Answer: $eax = \underline{1000}$; # memory operand accesses = 1
- 2(a) The following is a partial code for computing $Z = (X + Y) * (X - Y)$ in a 3-address computer that allows both register and memory operand in an arithmetic instruction. Fill in the missing lines of instruction.

```
Add  R1, X, Y           //R1 is register, X &Y are memory operands
Sub   R2, X, Y
Mul   Z, R1, R2
```

- (b) Indirect addressing is an important and useful feature in programming because it
allows an address (pointer) for instruction or data to be dynamically changed during
program execution. This is useful in control linkage and dynamic data accesses.
3. A pipelined processor adopts the assembly line principle in processing instructions from a given program. Ideally instructions advance through the pipeline in every clock tick, as in an assembly line. Unfortunately, from time to time, the advancement of an instruction from one stage of the pipeline to the next stage may have to stall temporally. Such a stall is typically caused by (i) data dependency and (ii) control dependency in the program.