L13: ILP

Name: _____

ID: _____

Question: Find all data dependences in this instruction sequence.

```
I1: ADD R1, R2, R1
I2: LW  R2, 0(R1)
I3: LW  R1, 4(R1)
I4: OR  R3, R1, R2
```

Instruction Scheduling Example

Consider the following loop code

```
lp: lw   $t0, 0($s1)      # $t0=array element
    addu $t0, $t0, $s2    # add scalar in $s2
    sw   $t0, 0($s1)      # store result
    addi $s1, $s1, -4     # decrement pointer
    bne  $s1, $0, lp      # branch if $s1 != 0
```

Must “schedule” the instructions to avoid pipeline stalls

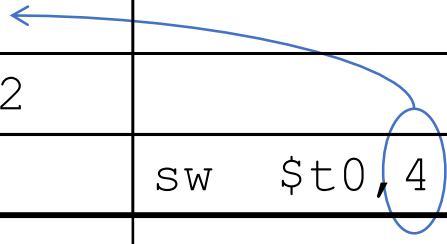
- Instructions in one bundle must be independent
- Must separate load use instructions from their loads by one cycle
- Notice that the first two instructions have a load use dependency, the next two and last two have data dependencies
- Assume branches are perfectly predicted by the hardware

The Scheduled Instruction (Not Unrolled)

	ALU or branch	Data transfer	CC
			1
			2
			3
			4
			5

The Scheduled Instruction (Not Unrolled)

	ALU or branch	Data transfer	CC
lp:		lw $t0,0($s1)	1
	addi $s1,$s1,-4		2
	addu $t0,$t0,$s2		3
	bne $s1,$0,lp	sw $t0,4($s1)	4
			5



- Four clock cycles to execute 5 instructions
- CPI of 0.8 (versus the best case of 0.5)
- IPC of 1.25 (versus the best case of 2.0)
- `noop`s don't count towards performance

Unrolled Code Example

```
lp: lw   $t0,0($s1)     # $t0=array element
    lw   $t1,-4($s1)    # $t1=array element
    lw   $t2,-8($s1)    # $t2=array element
    lw   $t3,-12($s1)   # $t3=array element
    addu $t0,$t0,$s2    # add scalar in $s2
    addu $t1,$t1,$s2    # add scalar in $s2
    addu $t2,$t2,$s2    # add scalar in $s2
    addu $t3,$t3,$s2    # add scalar in $s2
    sw   $t0,0($s1)     # store result
    sw   $t1,-4($s1)    # store result
    sw   $t2,-8($s1)    # store result
    sw   $t3,-12($s1)   # store result
    addi $s1,$s1,-16    # decrement pointer
    bne  $s1,$0,lp      # branch if s1 != 0
```

The Scheduled Code (Unrolled)

	ALU or branch	Data transfer	CC
			1
			2
			3
			4
			5
			6
			7
			8

The Scheduled Code (Unrolled)

	ALU or branch	Data transfer	CC
lp:	addi $s1,$s1,-16	lw $t0,0($s1)	1
		lw $t1,12($s1)	2
	addu $t0,$t0,$s2	lw $t2,8($s1)	3
	addu $t1,$t1,$s2	lw $t3,4($s1)	4
	addu $t2,$t2,$s2	sw $t0,16($s1)	5
	addu $t3,$t3,$s2	sw $t1,12($s1)	6
		sw $t2,8($s1)	7
	bne $s1,$0,lp	sw $t3,4($s1)	8

- Eight clock cycles to execute 14 instructions
- CPI of 0.57 (versus the best case of 0.5)
- IPC of 1.8 (versus the best case of 2.0)