

Introduction to Computer Science

Lecture 6: Instructions, Assembly Language, and Machine Code

Iain Styles

I.B.Styles@cs.bham.ac.uk

Assessment and Schedule

- My last lecture
- Next week: **NO LECTURES**
- Instead, assessment prep
 - Quiz 1 (5% MSc, 10% YiCS), open Monday, closes Friday – you have 1 hour
 - See practice quiz on canvas
 - (MSc) Essay released – due 2 Dec. Use the week to think and prepare
 - Final assignment, Quiz 2, will be 5-9 Dec.

Remaining Weeks

- Weeks 5-8: Dr Ata Kaban, on the inner workings of Java
 - A.Kaban@cs.bham.ac.uk
- Weeks 9-11: Dr Dave Parker, on analysis of computer programs
 - d.a.parker@cs.bham.ac.uk



Recap: Decomposing Programs

```
if(a==1) b=0;  
else b=1;
```

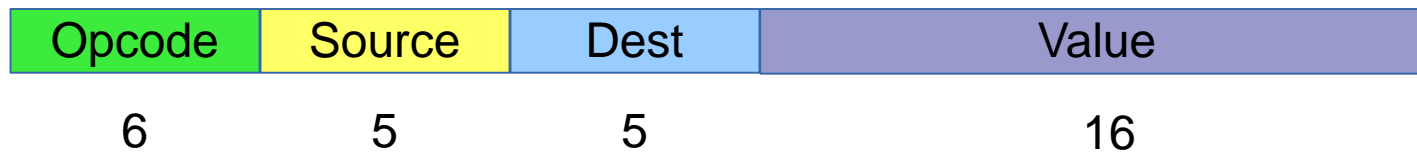
Load a:		lw \$8, &a
Value 1:		li \$9, 1
If a!=1:		bne \$8, \$9, L1
b=0:		li \$10,0
jump:		j L2
b=1:	L1:	li \$10,1
end:	L2:	sw \$10,&b

Some instructions use one register, some use two, some use three, some take values, some take addresses. How?

MIPS Instruction Types

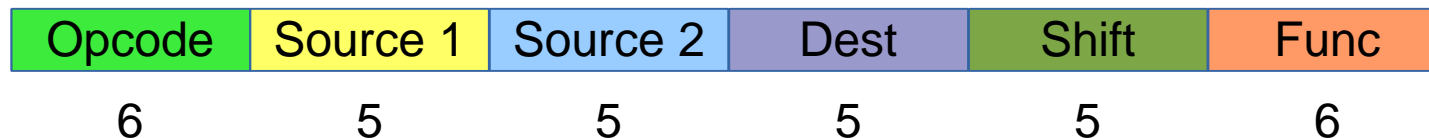
- I-Type

- Require one or two registers and a value



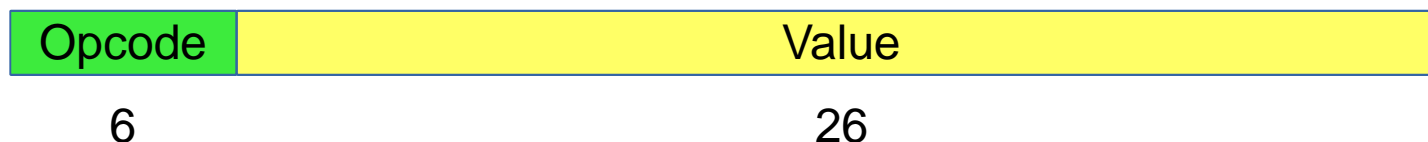
- R-Type

- Require three registers



- J-Type

- Require a value



A For Loop

```
y=0;
```

```
for (i=1; i<=x; i++){
```

```
    y = y + i;
```

```
}
```

MIPS Instructions

Load/Store	Load word	lw \$1 , &a	Load contents of address &a into register r1
	Store word	sw \$1 , &a	Store contents of register r1 into address &a
	Load immediate	li \$1 , 100	Load value 100 into \$1
Arithmetic	Add	add \$1,\$2,\$3	$r1 = r2 + r3$
	Subtract	sub \$1,\$2,\$3	$r1 = r2 - r3$
	Add immediate	addi \$1,\$2,100	$r1 = r2 + 100$
	Add unsigned	addu \$1,\$2,\$3	$r1 = r2 + r3$
	Subtract unsigned	subu \$1,\$2,\$3	$r1 = r2 - r3$
	Add immediate unsigned	addiu \$1,\$2,100	$r1 = r2 + 100$
Multiply/Divide	Multiply	mult \$1,\$2,\$3	$r1 = r2 \times r3$
	Multiply Unsigned	multu \$1,\$2,\$3	$r1 = r2 \times r3$
	Multiply Immediate	multi \$1,\$2,4	$r1 = r2 \times 4$
	Divide	div \$1,\$2,\$3	$r1 = r2 / r3$
	Divide unsigned	divu \$1,\$2,\$3	$r1 = r2 / r3$
	Divide Immediate	divi \$1,\$2,5	$r1 = r2 / 5$
Logical	AND	and \$1,\$2,\$3	$r1 = r2 \& r3$
	OR	or \$1,\$2,\$3	$r1 = r2 \mid r3$
	NOR	nor \$1,\$2,\$3	$r1 = \neg(r2 \mid r3)$
	AND immediate	andi \$1,\$2,100	$r1 = r2 \& 100$
	OR immediate	ori \$1,\$2,100	$r1 = r2 \mid 100$
	XOR immediate	xori \$1,\$2,100	$r1 = r2 \hat{100}$
	Shift left logical	sll \$1,\$2,10	$r1 = r2 \ll 10$
	Shift right logical	srl \$1,\$2,10	$r1 = r2 \gg 10$

A For Loop

Load x	lw \$8,&x
y=0	li \$9,0
i=1	li \$10,1
i-x	L1: sub \$11,\$10,\$8
if i-x>0 break	bgtz \$11,L2
y=y+i	add \$9,\$9,\$10
i++	addi \$10,\$10,1
Loop	j L1
End	L2: sw \$9,&y

Summary

- Examples of constructing “useful” programs at the machine level
- As you learn to program in Java,
- Next, the JVM
 - The software layer between Java and the physical hardware