Lecture 7
Loading

CS 241: Foundations of Sequential Programs
Fall 2014

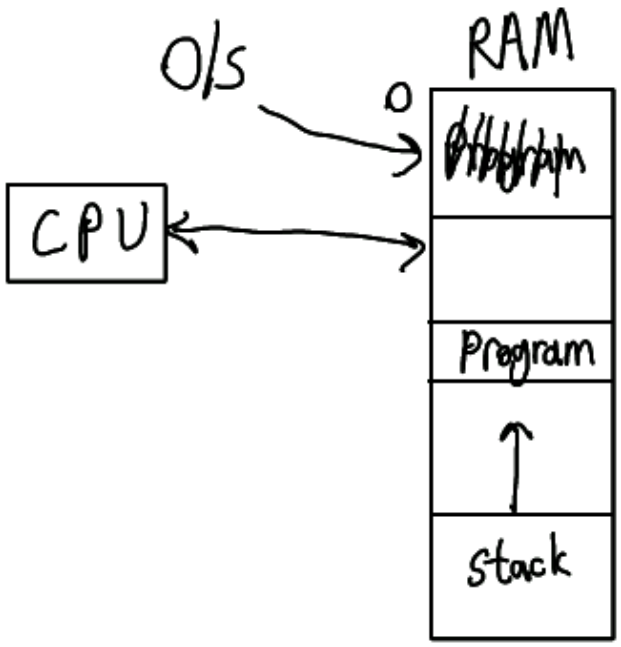
Troy Vasiga et al
University of Waterloo

Correctness

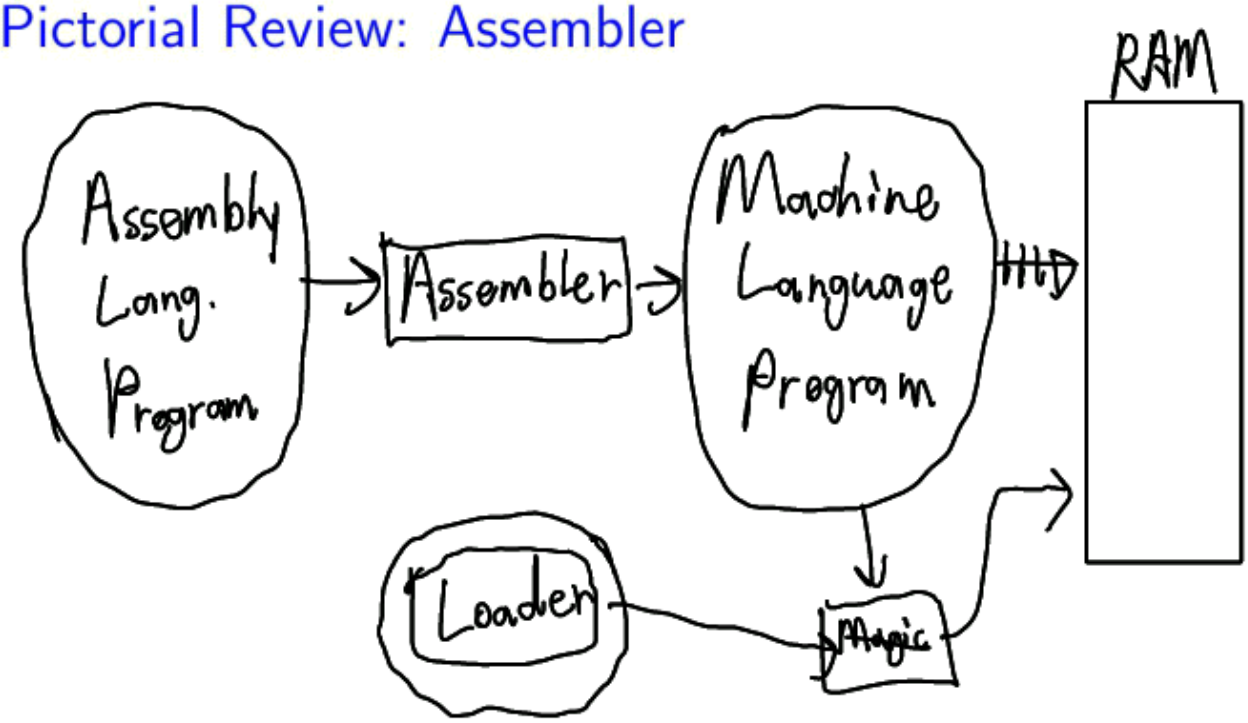
- **Read the spec carefully!**

- think reasonably and unreasonably
- remembering memory
- running times do matter
- **Don't use Marmoset as your only testing tool!**

Pictorial Review: CPU and RAM

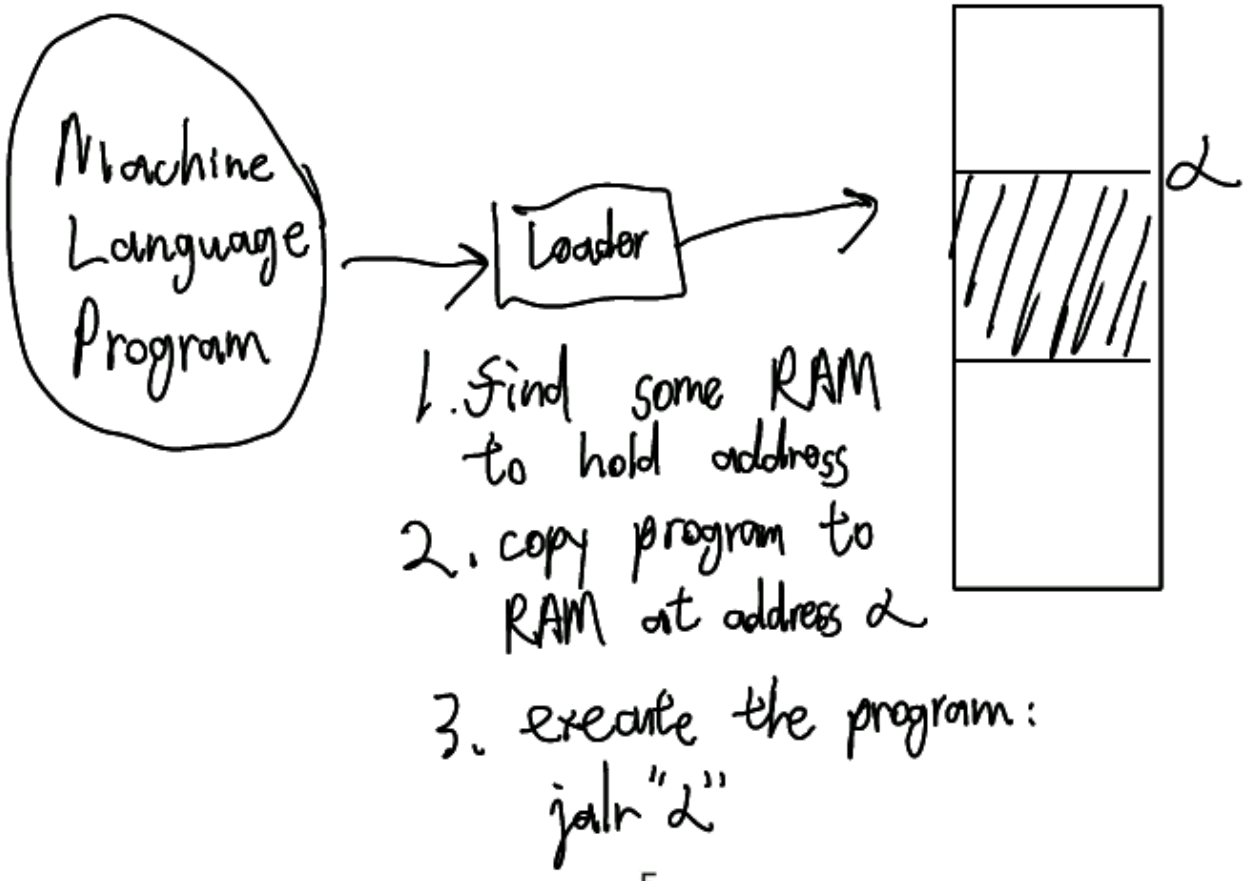


Pictorial Review: Assembler



The magic of loaders

This is what the operating system does!



Loader pseudocode

Version 0.1

```
loop
    decide which program to run
    figure out length of the program (n)
    find n words of storage at address α
    read program into memory at α
    set up program (e.g., twoints)
    put α into a register (say $29)
    jalr $29
endloop
```

CS 350 / OS course

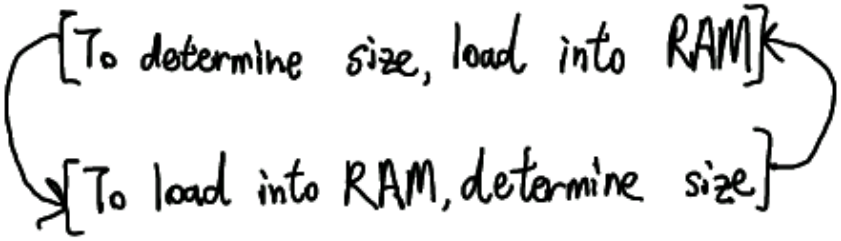
find space

copy program

execute

Chickens and eggs

- What is the size of the program we want to load?



- Isn't the loader also a program?

↳ microloader / Boot loader

BIOS "firmware"

"Boot strapping"

Example Program

Symbol Table
0x10 fortytwo

RAM

0	0x00001814
4	0x00000010
8	------
c	------
10	0x0000002a

```
0  lis $3
4  .word fortytwo
8  lw $3, 0($3)
c  jr $31
10 fortytwo:
     .word 42
```

Why does this work? When does it not work?

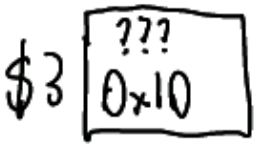
↳ base address is 0.

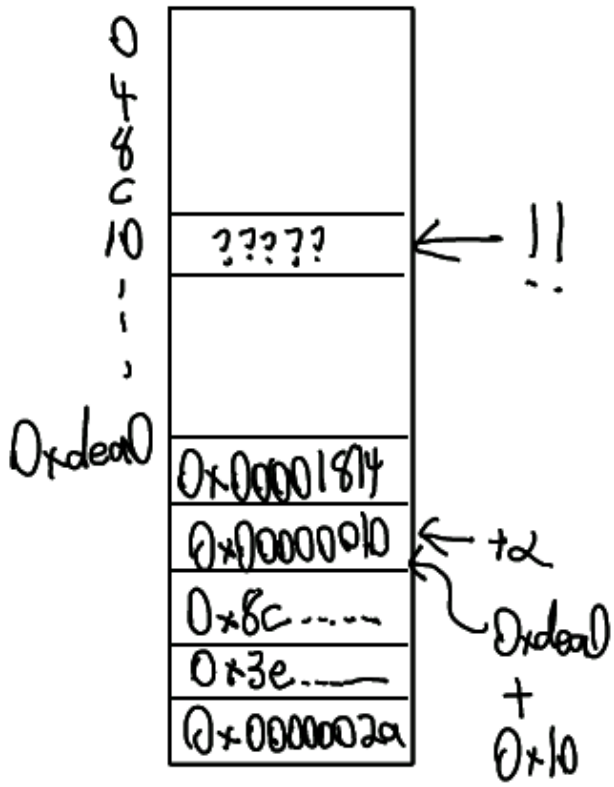
Notice how `mips.twoints` works:

```
@ubuntu1204-004[101]% java mips.twoints
Usage: java mips.twoints <filename> [load_address]
```

Example Program Again

```
lis $3
.word fortytwo
lw $3, 0($3)
jr $31
fortytwo:
  .word 42
```





What breaks here? How can we fix it?

.word <label> ← absolute address!!

What else breaks in MIPS?

↳ beq/bne?

↳ relative address

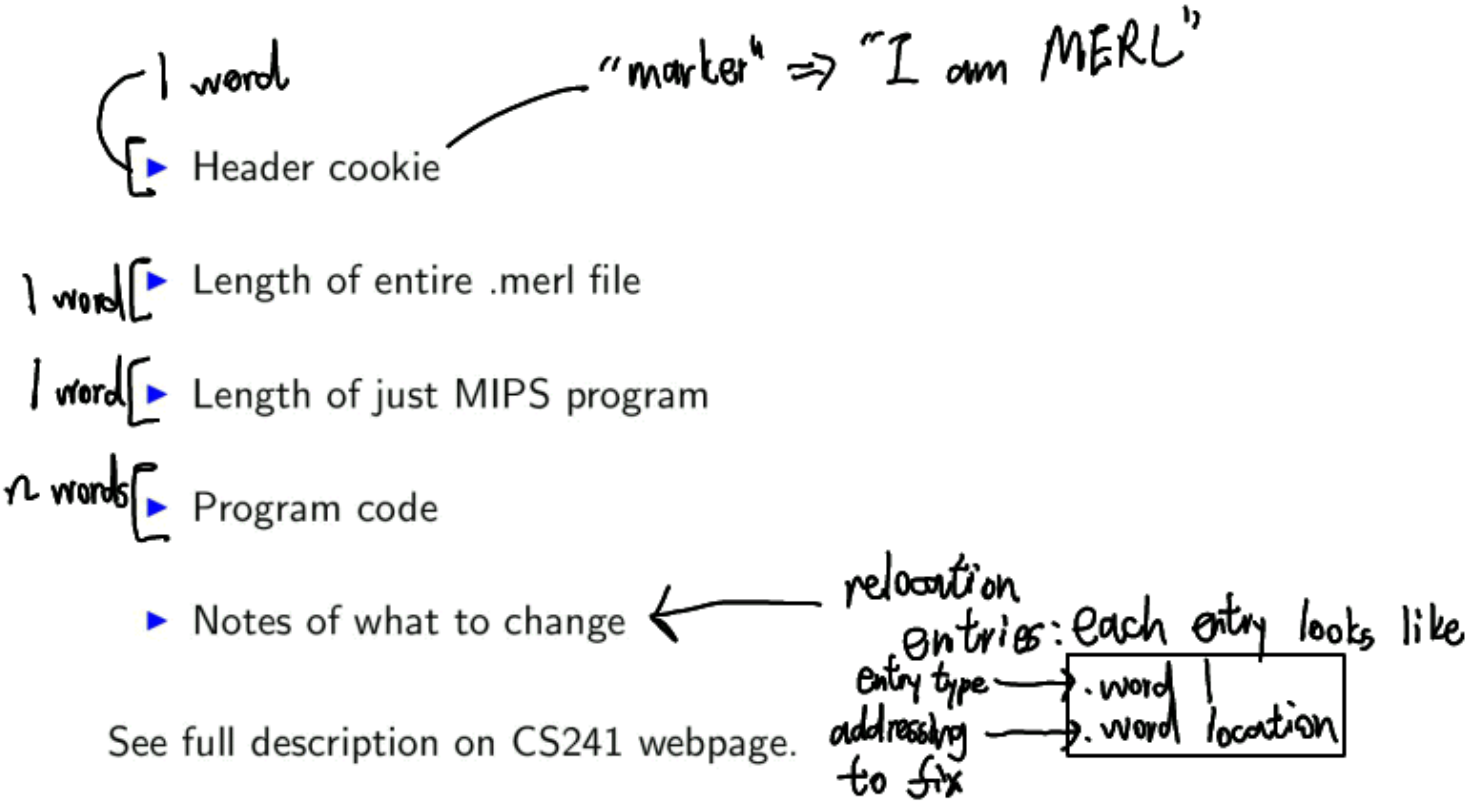
∴ $5ine_9$

Solving this problem

MERL

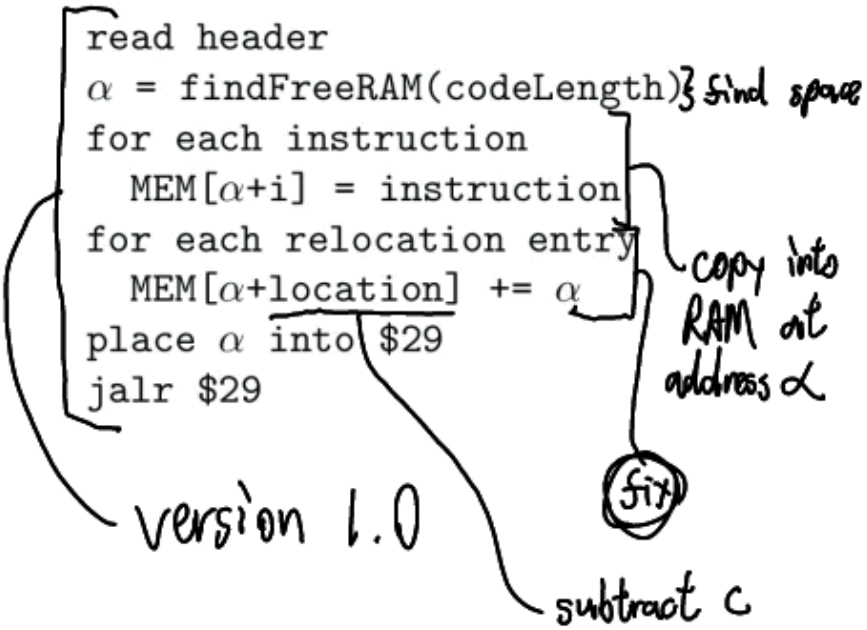
MIPS
Executable
Relocatable
Linkable
Format

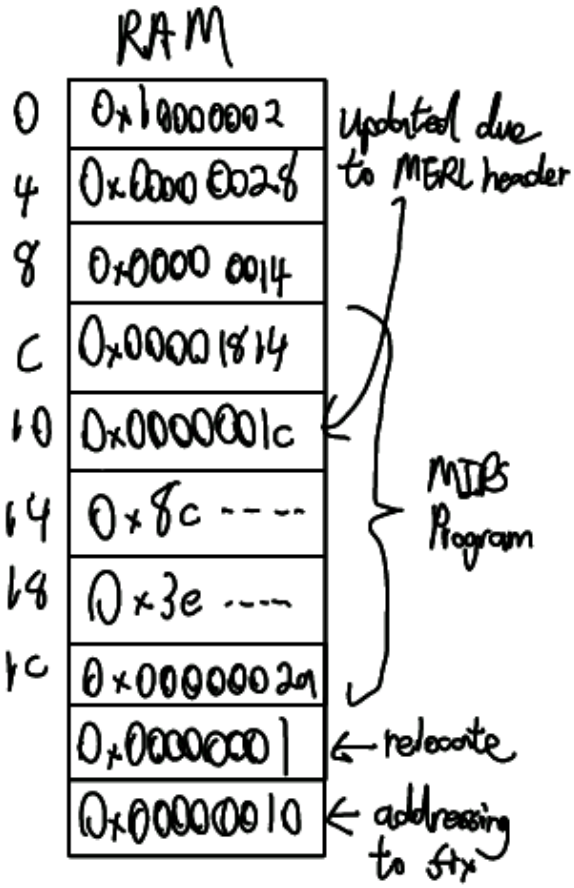
MERL Format



See full description on CS241 webpage.

Relocating Loader Pseudocode





A MERL Assembler

- Pass 1 changes:
  - ↳ keep track of length
  - ↳ when I encounter .word <label>
    - ⇒ record a relocation entry with the current location
  - ↳ start counting address at 0xC

- Pass 2 changes:
  - ↳ output header
  - ↳ output notes