CS 2420 — Computer Science II  
Section 20  
Spring 2011  
Instructor: Matthew Flatt
Reminder

Section 20 is the “systems track.”
Course Content

- **Machine model**
  - Machine code
  - C

- **Data structures**
  - Balanced binary trees, hash tables, ...

- **Algorithms**
  - Binary search, dynamic programming, ...
CS 1410-20 Machine

(+ (* 3 4) 8)
→ (+ 12 8)
→ 20
CS 2420-20 Machine
# The Jam2000 Machine

## Memory

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>17</th>
<th>96</th>
<th>3</th>
<th>4</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>17</td>
<td>1</td>
<td>88</td>
<td>77</td>
<td>11</td>
<td>19</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
<td>11</td>
<td>26</td>
<td>2</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>81</td>
<td>5</td>
<td>97</td>
<td>31</td>
<td>45</td>
<td>90</td>
</tr>
</tbody>
</table>

## CPU

13

## Registers

| 40 | 8 | 37 | 9 | 13 | 44 | 9 |
The Jam2000 Machine

Disk → Memory

```
0 1 17 96 3 4 9
80 17 1 88 77 11 19
4 21 11 26 2 8 5
2 81 5 97 31 45 90
```

CPU → Screen

```
[13]: +0019 (Id R1 0) %
```

CPU → Registers

```
40 8 37 9 13 44 9
```

Screen

Registers
Everything is a Number

The number 1490: 1490

The “copy register 4 to 1” instruction: 1490

The name of the 1491st memory cell: 1490

A greenish-blue pixel: 1490
Everything is a Number

The number 1490: 1490
The “copy register 4 to 1” instruction: 1490
The name of the 1491st memory cell: 1490
A greenish-blue pixel: 1490

In fact, everything is represented by a number between -999999999 and 999999999

• but memory addresses are 0 to 9999
• only some numbers are sensible instructions
• the screen ignores some digits
Pixels

- Screen memory starts at 8000
- Digits encode red, green, and blue intensities:
  \[ ? \quad ? \quad d_r \quad d_r \quad d_g \quad d_g \quad d_b \quad d_b \]

<table>
<thead>
<tr>
<th>0</th>
<th>990000</th>
<th>9900</th>
<th>99</th>
<th>888888</th>
<th>999900</th>
</tr>
</thead>
</table>

A diagram showing a pixel with the specified intensity values.
Instructions

Numbers encode actions like “put the number $d_7d_6d_5d_4d_3d_2$ in register $x$”:

$$(\text{ldi } Rx \ d_7d_6d_5d_4d_3d_2)$$

$d_7 \ d_6 \ d_5 \ d_4 \ d_3 \ d_2 \ x \ 9$

“Put the number 77 in register 6”:

$$(\text{ldi } R6 \ 77)$$

$0 \ 0 \ 0 \ 0 \ 7 \ 7 \ 6 \ 9$$