40sfor40: A Malt Liquor Machine

COMP40 Roadmap

Emulate a machine with only one instruction: *CHUG!*

✓ **Ramp up your Programming Skills**

✓ **Big programs that teach you abstraction, pointers, locality, machine representations of data**

✓ **Intel Assembler Programming *The Bomb!***

✓ **Emulating your own hardware in software**

✓ **Building a Language Processor on your Emulator**

✓ **Building Useful Applications in your Language**
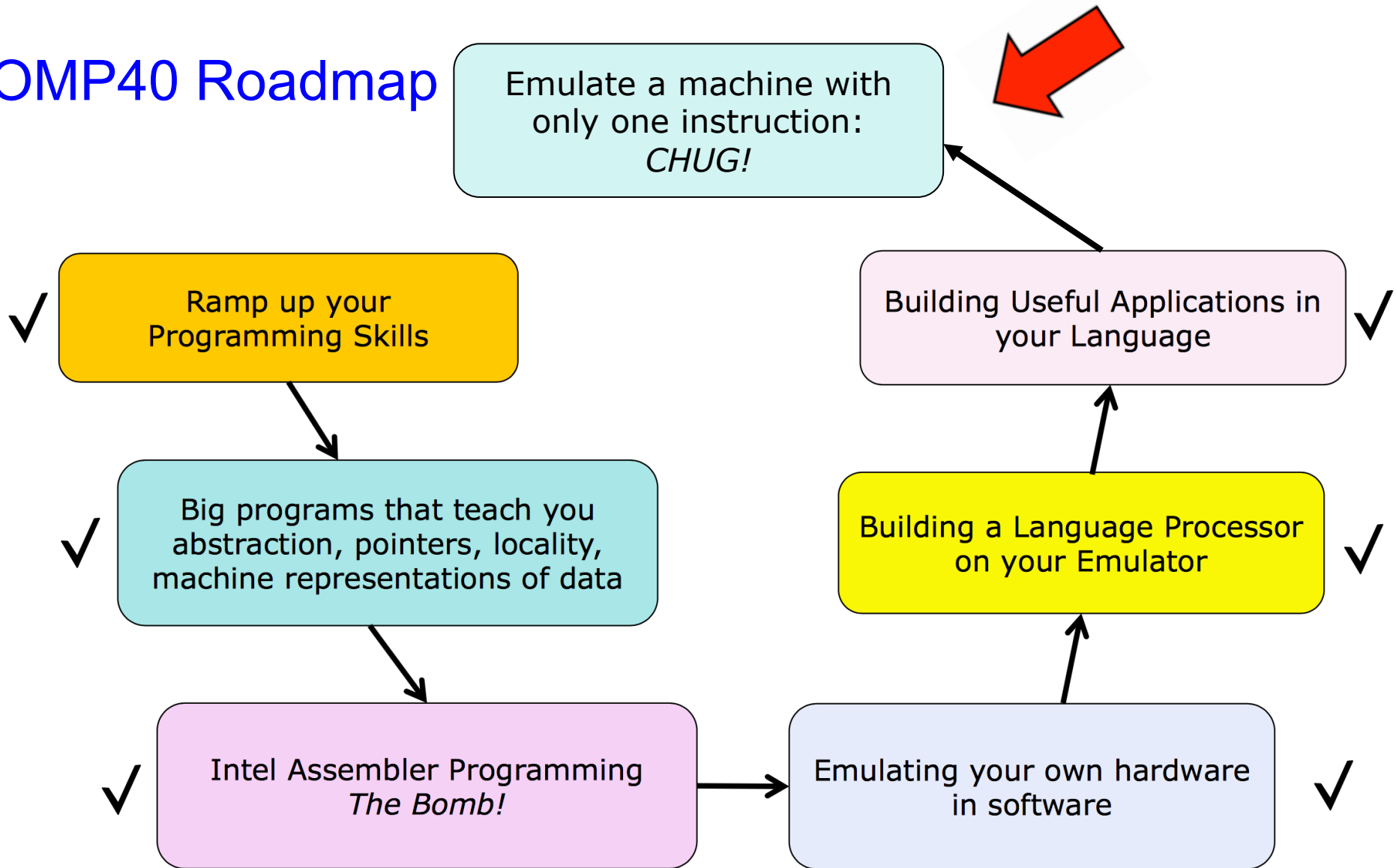
Introducing
40s for 40

40s for 40 highlights

- **Bottled (lack of) memory architecture.** Memory is organized into bottles, each containing forty ounces of premium malt liquor.

- **1 simple instruction.** Transfers an amount of liquid from bottled-memory to a register and halts when all bottles are empty (see instruction set semantics)

- **General purpose registers,** with varying capacity between 40 and 80 ounces.

Instruction set

- **One instruction, with opcode 0:** *chug (and halt if empty).*

- **Formally, takes a destination register, immediate value (amount to chug), and bottle from which to chug.** We might think of it this way in register transfer language (RTL):

```
m[BOTTLE] := m[BOTTLE] - AMOUNT_TO_CHUG
r[DEST] := AMOUNT_TO_CHUG
```

- **If *all bottles in memory are completely empty,* the machine halts successfully.**

- **Can we can perform arbitrary computation using only this instruction?**
  - Could you implement subtraction? Negative numbers? What about a `GETLIT` macro?

Disclaimer: Not official COMP40 lecture material.

What we expect of you

- **Your design document:**
  - *Very good* design documentation describes succinctly whether you plan on consuming 40 or 80 ounces of malt liquor. You may submit either `DESIGN` or `design.pdf.`

- **Your implementation (due May 10):**
  - The biggest task by far is to *choose a representation for bottles.* Common choices include `Olde_English_800_T`, `Colt_45`, and `St_Ides`.
  - Within these choices, there are both glass and plastic options (we recommend the safety and ease-of-use of plastic abstractions).
  - **Keep in mind your choice of brand** can affect performance by as much as 420%!

Disclaimer: Not official COMP40 lecture material.
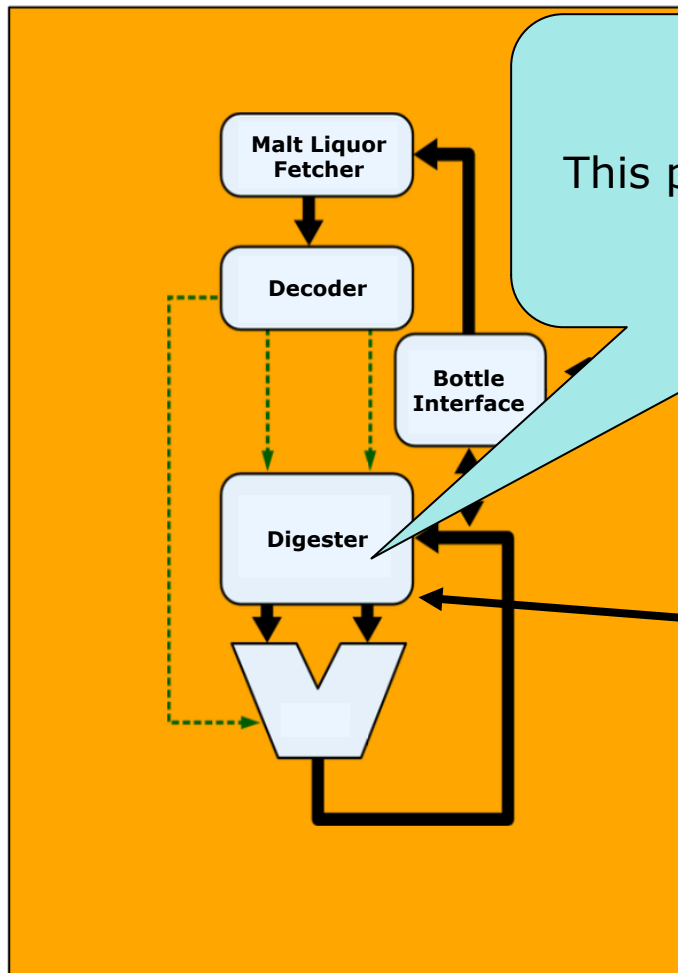
Example of a program run in the machine

Initial machine state: registers Joanne, Bob, Amanda Initialized to zero and 3 full bottles in memory.

```
chug Joanne,2,4
chug Joanne,2,4
chug Joanne,2,4
chug Joanne,2,10
chug Bob,0,8
chug Bob,0,10
chug Amanda,1,40
chug Joanne,2,18
chug Bob,0,22
```

All the bottles are empty, execution stops!

Disclaimer: Not official COMP40 lecture material.

Important things to keep in mind

- **Most experienced C programmers can understand the 40sfor40** specification in a couple of minutes, and can implement the machine in just a few hours following the final exam.

- **History footnote** *(from Wikipedia)*:
  - The term "malt liquor" is documented in England in 1690 as a general term encompassing both beer and ale. The first mention of the term in North America appears in a patent issued by the Canadian government on July 6, 1842, to one G. Riley for "an improved method of brewing ale, beer, porter, and other maltliquors."

- **Most importantly,** have *fun*, be *responsible*, and be *safe*!