Computer Systems and Networks

ECPE 170 – Jeff Shafer – University of the Pacific

MARIE Programming

Schedule

- **Today**
  - Review / discuss MARIE assembly homework problems
  - **Opportunity to re-submit HW #10 until midnight**
  - Office hours this afternoon: 1:30pm+

- **Friday 24th**
  - Start Chapter 5
  - **Quiz 3!**
    - Topic: Assembly programming!
    - *I will give you Table 4.7 from the book*

Recap on I/O Instructions

↗ **INPUT**
  - ↗ **Where is the value from the keyboard stored?**
  - ↗ The accumulator!

↗ **Output**
  - ↗ **Where does the value on the display come from?**
  - ↗ The accumulator!

Homework 4.32

↗ **Write a MARIE subroutine to subtract two numbers**

↗ **What do we need in our program to handle the mechanics of a subroutine?**

- ↗ Arguments to the function (i.e. input data)
- ↗ Return value from the function
- ↗ A way to jump to the function
- ↗ A way to return from the function when finished

Homework 4.33

↗ **Write a MARIE program to traverse a linked list and print the data stored in each node**

   ↗ In this case, the data/desired output is the sequence 1 2 3 4 5

   ↗ The linked list is scrambled in memory
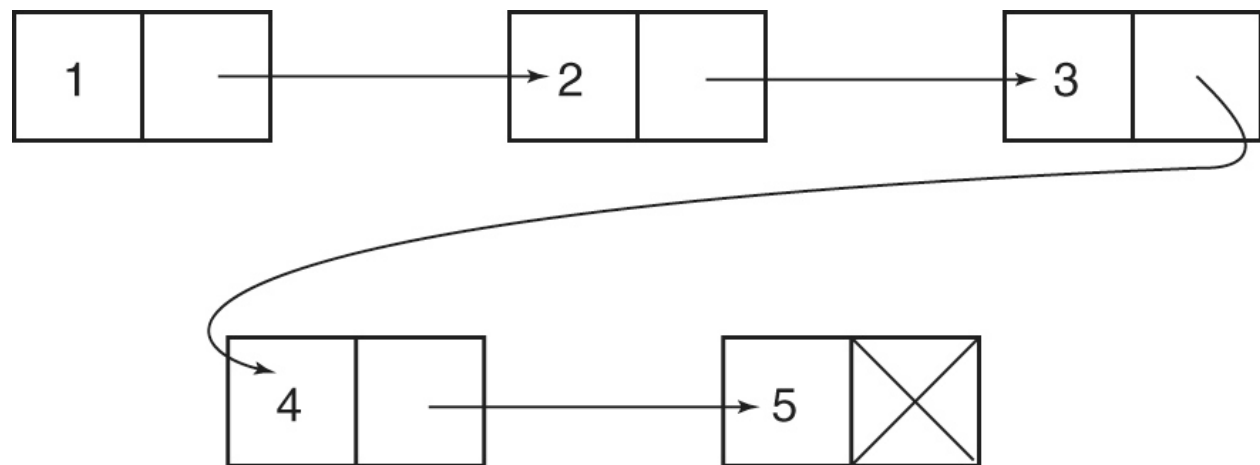
Homework 4.33

```
Addr,  Hex ____        / Top of list pointer
Node2, Hex 0032        / Node's data is the character "2."
       Hex ____        / Address of Node3.
Node4, Hex 0034        / Character "4."
       Hex ____
Node1, Hex 0031        / Character "1"
       Hex ____
Node3, Hex 0033        / Character "3"
       Hex ____
Node5, Hex 0035        / Character "5"
       Hex 0000        / Indicates terminal node.
```
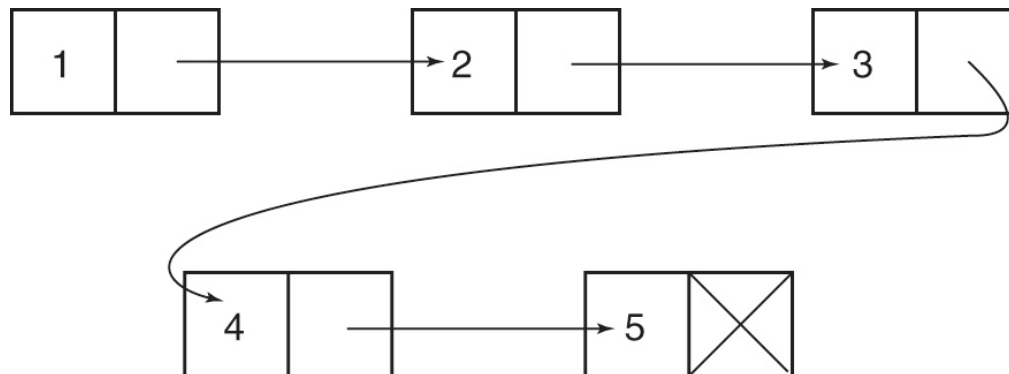
Discuss Quiz

Clever Tricks

↗

Clever Tricks – Faking LOADI

↗ MARIE has LOAD-Indirect (LOADI) and STORE-Indirect (STOREI) instructions

   ↗ But clever programmers don't need them!

↗ **How could I "emulate" the LOADI X instruction using several non-indirect MARIE instructions?**

```
CLEAR       / Put 0 in AC
ADDI X      / Add indirect value from Mem[Mem[X]]
```

Clever Tricks – Faking STOREI

- ↗ **How could I "emulate" the `STOREI X` instruction using several non-indirect MARIE instructions?**
    - ↗ This is harder!

- ↗ Idea: Take advantage of the stored program concept
    - ↗ Instructions are just data

- ↗ We need a sequence of instructions that construct a STORE instruction with the desired address

- ↗ This would be a good application of a subroutine
    - ↗ Pass the value to store in AC, place the address in a parameter variable

Clever Tricks – Faking STOREI

↗ **Equivalent code to** `STOREI X:`

```
            LOAD        STROPCODE    / Get opcode
            ADD         X            / Combine addr
            STORE       STOREI       / Save
STOREI,     HEX         0       / Data: build instruction
                                / here, then execute it
            / Program continues here...
...
...
STROPCODE,  STORE       0       / Data: Just opcode
                                / for store
```