Spring 2015

CSC-111 -Intro to Computer Science

SPRING 2015

############ LECTURE 1: Tuesday, Jan 20 2015

CSC-111 Intro to Computer Science

Prof. Andrew Tjang

atjang@cs.rutgers.edu

Office: HLL407

Hours: TTH 2-3pm

Other Lecture: 3:20-4:40pm Monday-Wednesday

-- No Attendance Required --

Recitations:

-Group problem solving

-Led by undergraduate student.

-PARTICIPATION REQUIRED

Book: Intro to Programming Using Java v.7 by David J. Eck

Assignments: Sakai: grades, assignments, notes from prof. CODELAB: practice coding.

############ LECTURE 2: Thursday, Jan 22 2015

BINARY:

High and Low electricity signal enable us send binary signals to the processor.

0 1 true false yes no high Low -- 0: LOW VOLTAGE --1: high VOLTAGE

Bit: either 0 or 1.

4 bits = 2^4: 16 permutations. Every bit doubles the amount of information we can store.

INFORMATION REPRESENTATION

Clock: procesor intervals (low/high)

Data:

AMI:

LOGIC AND ARITHMETIC

ALU: arythmetic logic unit.

-Logic gates AND OR XOR

\* Boolean logic on binary values

\* Truth tables

-Can do arithmetic with logic gates?

\* Simple adder.

AND truth table

-------------------

green | red | green ^ red

0 1 0

1 0 0

1 1 1

0 0 0

OR truth table

-------------------

green | red | green || red

0 1 1

1 0 1

1 1 1

0 0 0

XOR truth table

-------------------

green | red | green + red

0 1 1

1 0 1

1 1 0

0 0 0

Unary Operator "NOT"

-------------------

green | red | Not !Green

0 1 1

1 0 0

1 1 0

0 0 1

1+1 = 0\* (when it has to fit in one bit. \* There's a 1 overflow)

8 bits = 1 byte 2^8 = 256 we can store 0-255 numbers in a byte

Last Most Significatn bit tells me if number is negative or positive. 0 = negative, 1 = positive.

Characters that can be stored: 256

WHAT IS Programming

------------------------------

-Procedural vs Declarative (HTML) vs Functional vs OOP.

-Java as procedural and OOP.

-Algorithmic Thinkin

+ Methodical

+ Verifiable (formally or through testing)

THE PROGRAMMING PROCESS

------------------------

1. Problem Analysis

+ Inputs, outputs, error conditions.

2. Program Design

3. Algorithm Design

+ will use flowcharts

4. Coding

+ will use a programming language

5. Testing

+ test case construction, debugging.