CS180 Recitation 1
Lecture Recap

- **Class**
  - Defines a blueprint for creating 'many' of a 'kind'.
  - Allows to store properties common to everyone in a 'class'.
  - eg: 'Courses offered at Purdue'
  - What are the properties common to all Courses?

- **Object**
  - An instance of the Class
  - 'CS180 Spring 2012' is one instance of 'Courses Offered At Purdue'
  - For CS180, what are the values for the properties you identified above.
Recap contd..

- Properties common to all Courses Used when defining a class
  - Number of students (What should the 'type' of this property)
  - Class Room
  - Class Timing
  - Instructor

- One instance - 'CS180 Spring 2012' (Name another 'instance' of this class)

- Values for properties defined above
  - Number of students: 180
  - Class Room: RPH 172
  - Class Timing: 12:30 pm - 1:20 pm - T, R

- Another object of this class will have the same properties, but different values for those properties. (MA101, 50, MATH 406, M, W, F 1-1.50)
Object Oriented Programming

- Programs are modeled as Objects and interactions between them.
- eg: 'Student' objects interact with 'Course' objects for enrollment.
Example

- Enroll a student 'Tom' for course 'CS180' in an 'Object Oriented' way
- Create class 'Course' with properties like
  - Maximum #students.
  - Current enrollment
  - List of registered students
  - Other properties?
- Create class 'Student' with properties like
  - Name of Student
  - Year or study
  - Other properties?
Create object CS180 of class - Course
- Let Max #students = 180
- Let Current enrollment = 0
- List of registered students empty for now
- Other values?

Create object Tom of class - Student
- Name = 'Tom'
- Year = 'Freshman'
- Other properties?
- how to enroll Tom for CS180?
Example Contd..

- Create a method ‘enroll’ in Class - Course.
- This method should
  - Check if there is vacancy. (check if current enrollment less than max students)
  - Add Tom to the list of registered students.
  - Increase current enrollment to 1
- Student object will invoke this enroll method.
- This allows a programmer to create instances representing all courses and all students from just 2 classes.
Why use Objected Oriented Programming

- Makes it easier to write programs.
- Makes it easy to understand other programs. Why is this important?
- Allows 'code reuse'. How?
- Many more features eg?
Number Systems

▸ Decimal Number System
  ▸ 10 distinct digits 0, 1, 2 ... 9.
  ▸ Base is 10 (Because there are 10 unique digits)
  ▸ All numbers in this system should use these and only these digits. 12A3 - invalid

▸ How is a number formed in decimal number system?

\[
125 = 100 + 20 + 5 \quad (1)
\]
\[
= 1 \times 10^2 + 2 \times 10^1 + 5 \times 10^0 \quad (2)
\]

▸ Every digit is multiplied by Base raised to a positional value
▸ Positional value increases from right to left

▸ Should Base always be 10? What if we use only 2 distinct digits?
Binary Number System

- **Uses only 2 distinct digits** Why? Because electrical systems understand high and low
  - 1 and 0. Base is 2
  - All numbers should be formed using 1 and 0 only
    - 1121 - Invalid. Why?
    - 10110 - Valid.

- How is a number formed in Binary number system?
  - Just like decimal, but use 2 as base.

\[
1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \quad (4)
\]
\[
= 8 + 4 + 0 + 1 \quad (5)
\]
\[
= 13_{10} \quad (6)
\]

- Note how place value doubles as we move left
- Can base be any value? Can base be 8? 16?
Converting from Binary to Decimal

- Based on how binary numbers are formed.
- We use a subscript or letter to indicate the number system.
  - 1101₂ or 1101b
  - 1296₁₀ or 1296d
- Another example: Convert 1101₁₂ to decimal

\[
1101₁₂ = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \quad (7)
\]
\[
= 16 + 8 + 0 + 2 + 1 \quad (8)
\]
\[
= 27₁₀ \quad (9)
\]
Some Binary numbers and their corresponding decimal values.

\[
\begin{align*}
0_2 &= 0_{10} & (10) \\
1_2 &= 1_{10} & (11) \\
10_2 &= 2_{10} & (12) \\
11_2 &= 3_{10} & (13) \\
100_2 &= 4_{10} & (14) \\
101_2 &= 5_{10} & (15) \\
110_2 &= 6_{10} & (16) \\
111_2 &= 7_{10} & (17)
\end{align*}
\]
Decimal to Binary

- Identify powers of 2 that sum to the decimal digit
  - $11_{10} = 8 + 2 + 1$
  - Place 1 in positions identified above and 0 otherwise
    
    | 8 | 4 | 2 | 1 |
    |---|---|---|---|
    | 1 | 0 | 0 | 1 |
  
  - Too tedious for large decimal numbers

- Long Division Method (Commonly used).
  - Repeatedly divide by 2 and identify the remainder
  - What is the highest and lowest remainder possible when we divide by 2?
  - Example convert $23_{10}$ to binary
Decimal to Binary - Example

- Convert $23_{10}$ to binary

  \[
  \begin{array}{c|c|c}
  2 & 23 & \rightarrow 1 \\
  \hline
  2 & 11 & \rightarrow 1 \\
  \hline
  2 & 5 & \rightarrow 1 \\
  \hline
  2 & 2 & \rightarrow 0 \\
  \hline
  & 1 & \\
  \end{array}
  \]

- Read the remainder in reverse order i.e $10111$

- $23_{10} = 10111_2$

- Verify by converting back to decimal
  - $10111_2 = 2^4 + 2^2 + 2^1 + 2^0 = 16 + 4 + 2 + 1 = 23_{10}$
Problem Solving Using Algorithms

- Find maximum of three values
- Let a, b and c represent any three integer values
- Solve on black board.