this Pointer, Constant Functions, Static Data Members, and Static Member Functions
this Pointer \((11.1)\)

- functions - only one copy of each function exists in memory independent of the number of objects instantiated using the class declaration
- data members - each unique object of a particular class has space allocated for the data members of the class
- this - is a pointer that can be used to access an objects data members. this is an implicit argument to all class methods, constructors, and destructors.
Example of this pointer

```cpp
#ifndef RATIONAL_H
#define RATIONAL_H

using namespace std;

class Rational
{
public:
    Rational(int, int);
    print();

private:
    int numerator;
    int denominator;
};

#endif
```
Example

#include "Rational.h"

Rational::Rational(int numerator, int denominator)
{
    (*this).numerator = numerator;
    (*this).denominator = denominator;
}

Rational::print()
{
    cout << numerator << '/' << denominator;
}
Pointers

- Accessing data members and pointers using pointers
- (*this).numerator can be replaced with
- this->numerator
class Time
{
  private:
    int hour;
    int minute;
    int second;
  public:
    Time();
    Time(int hour = 0, int minute = 0, int second = 0);
    int getHour();
    int getMinute();
    int getSecond();
    void setTime(int hour, int minute, int second);
    void printUniversal();
    void printStandard();
}; // end class Time
const

• Many things can be specified as const in C++

• Examples:
  ◦ Objects
  ◦ Member Functions
  ◦ Data members
  ◦ Function arguments
const Objects

• Principle of least privilege

• What happens when we declare any object to be a const?

• Example:
  ○ const int SIZE = 50;

• What do you think it means if I have
  ○ const Time dinnerTime(18, 30, 0);

• What member functions of class Time do you think dinnerTime can call?
const Member Functions

- A const object can only call const functions.

- How do we declare member functions to be const?
  - Use the const keyword in both the function prototype and the function definition.
  - Appears after the parameter list.

- const member functions CANNOT modify data members (i.e. the current instantiation of the class).
class Time
{
  private:
    int hour;
    int minute;
    int second;
  public:
    Time();
    Time(int = 0, int = 0, int = 0);
    int getHour() const;
    int getMinute() const;
    int getSecond() const;
    void setTime(int, int, int);
    void printUniversal() const;
    void printStandard() const;
}; // end class Time
Object Details

• What does memory look like after creating multiple objects of a class?

• For example:
  
  ○ `Time t(3, 45, 00);`
  ○ `Time t2(5, 29);`
  ○ `Time t3(14);`
  ○ `Time t4;`
  ○ `Time *pTime = new Time();`