Lecture 9.2

Topics

1. Pointer to Objects – Recalled
2. Pointers as Member Data – Brief

---

1. Pointer to Object – Recalled

A pointer to object can be declared just as those for variables. Mostly, a pointer to object will be pointing to a dynamic object such as follows,

```
Fraction* frPtr;
frPtr = new Fraction(); // dynamic object through a
                        // default constructor
```

2. Pointers as Member Data – Brief

A pointer can also be made as a member data of a class. The following descriptions will show a simple example.

```
// Class Specification File
/**
 * Program Name: dynamicArray.h
 * Discussion:  Class with a pointer member data
 *                Pointer member data is used to create
 *                 a dynamic array of int's
 */
#ifndef DYNAMICARRAY_H
#define DYNAMICARRAY_H

class DynamicArray {
public:
  DynamicArray();
  DynamicArray( const DynamicArray& arg );

  ~DynamicArray();

  int* getDataPtr();
  void setDataPtr( int* arg );

private:
  int size;
  int* dataPtr;
};

#endif
```

And,

```
// Implementation File
/**
 *Program Name:  dynamicArray.cpp
 *Discussion:    Class with dynamic array of int's
 */
#include <iostream>
#include "dynamicArray.h"
using namespace std;
```

```
DynamicArray::DynamicArray() {
  // TODO code
}

DynamicArray::DynamicArray( const DynamicArray& arg ) {
  // TODO code
}

DynamicArray::~DynamicArray() {
  // TO DO code
}

int DynamicArray::getSize() {
  //TODO code

  return 0;
}

void DynamicArray::setSize( int arg ) {
  // TODO code
}

int* DynamicArray::getDataPtr() {
  // TODO code

  return 0;
}

void DynamicArray::setDataPtr( int* arg ) {
  // TODO code
}
```

Explanations will be provided in class.