

## Section 1C. Class Mtrx (40점)

## 1C.1 class Mtrx

```

/** Class_Mtrx.h */
. . . . // 필요한 헤더파일 포함, 전처리기 설정

class Mtrx {
public:
    Mtrx(int n_row, int n_col);
    Mtrx(istream& fin);
    ~Mtrx(); // destructor
    int getN_row() { return n_row; }
    int getN_col() { return n_col; }
    string getName() { return name; };
    void setName(string nm) { name = nm; };
    Mtrx addMtrx(const Mtrx&);
    Mtrx subMtrx(const Mtrx&);
    Mtrx mulMtrx(const Mtrx&);
    void fprintMtrx(ostream& fout);
private:
    string name;
    int n_row;
    int n_col;
    double **dM;
};

```

- class Mtrx는 헤더파일 Class\_Mtrx.h에 구현할 것

## 1C.2 class Mtrx 멤버함수 구현

- class Mtrx는 데이터 멤버로 문자열 name, 정수형 n\_row와 n\_col, double 자료형의 이중 포인터 dM을 가진다.
- class Mtrx의 생성자 Mtrx(int n\_row, int n\_col)는 행렬의 크기 (n\_row, n\_col)를 인수로 전달받아 필요한 2차원 배열을 동적으로 생성한다.
- class Mtrx의 생성자 Mtrx(istream & fin)는 지정된 입력 스트림 객체 fin으로부터 class Mtrx 객체의 데이터 입력을 수행한다. 입력 객체 fin에는 행의 크기와 열의 크기가 정수 자료형으로 제공되며, 행렬의 각 원소 초기값이 소수점 이하 값을 가지는 실수 자료형으로 제공된다. Mtrx(istream & fin) 함수에서는 행의 크기와 열의 크기를 지정된 입력 스트림 객체로부터 입력받고, 해당 크기의 2차원 배열을 동적으로 생성하며, 각 배열 원소 데이터를 추가로 읽어 데이터 멤버를 초기화 시킨다.
- class Mtrx의 소멸자인 ~Mtrx()는 동적으로 생성되었던 2차원 배열의 메모리를 반환한다.
- getN\_row() 멤버 함수는 행의 개수 n\_row를 반환
- getN\_col() 멤버 함수는 열의 개수 n\_col을 반환
- getName() 멤버 함수는 행렬의 이름 (name)을 반환
- setName(string nm) 멤버 함수는 행렬의 이름 (name)을 설정
- class Mtrx의 addMtrx(), subMtrx(), mulMtrx() 멤버함수는 각각 행렬의 덧셈, 뺄셈, 곱셈 연산을 수행한다.
- fprintMtrx(ostream & fout) 멤버 함수는 class Mtrx의 데이터를 지정된 출력 스트림 객체 fout으로 출력한다. 출력 포맷은 행렬의 이름을 함께 출력하며, 행렬을 표시하는 대괄호 ([, ])를 표시하고, 각 원소는 소수점 아래 2자리까지 출력한다.
- class Mtrx 멤버함수 들은 Mtrx.cpp 파일에 구현할 것

1C.3 main() 함수 구현, 파일 입력 및 실행 결과 파일출력

```
/** main.c */

. . . . // 필요한 헤더파일 포함, 전처리기 설정

void main()
{
    ifstream fin;
    ofstream fout;

    . . . . // fin 및 fout 초기화

    Mtrx mtrxA(fin);
    mtrxA.setName("mtrxA");
    mtrxA.fprintMtrx(fout);

    Mtrx mtrxB(fin);
    mtrxB.setName("mtrxB");
    mtrxB.fprintMtrx(fout);

    Mtrx mtrxC(fin);
    mtrxC.setName("mtrxC");
    mtrxC.fprintMtrx(fout);

    Mtrx mtrxD(mtrxA.getN_row(), mtrxB.getN_col());
    mtrxD = mtrxA.addMtrx(mtrxB);
    mtrxD.setName("mtrxD = mtrxA.add(mtrxB)");
    mtrxD.fprintMtrx(fout);

    Mtrx mtrxE(mtrxA.getN_row(), mtrxB.getN_col());
    mtrxE = mtrxA.subMtrx(mtrxB);
    mtrxE.setName("mtrxE = mtrxA.sub(mtrxB)");
    mtrxE.fprintMtrx(fout);

    Mtrx mtrxF(mtrxA.getN_row(), mtrxC.getN_col());
    mtrxF = mtrxA.mulMtrx(mtrxC);
    mtrxF.setName("mtrxF = mtrxA.mul(mtrxC)");
    mtrxF.fprintMtrx(fout);

    fout.close();

} // end of main()
```

1C.4 입력 데이터 파일 및 실행 결과

<div>5 7</div> <div>1.1 2.2 3.3 4.4 5.5 6.6 7.6</div> <div>3.3 4.4 4.4 5.5 1.1 7.7 8.8</div> <div>5.5 6.6 5.5 3.3 2.2 4.4 6.6</div> <div>7.7 8.8 2.2 7.7 2.2 1.1 9.9</div> <div>9.9 1.1 3.3 2.2 9.9 6.6 9.9</div> <div>5 7</div> <div>1.1 0.0 0.0 0.0 0.0 1.1 6.6</div> <div>0.0 1.1 0.0 0.0 0.0 2.2 7.7</div> <div>0.0 0.0 1.1 0.0 0.0 3.3 8.8</div> <div>0.0 0.0 0.0 1.1 0.0 4.4 9.9</div> <div>0.0 0.0 0.0 0.0 1.1 5.5 9.9</div> <div>7 5</div> <div>1.1 2.2 3.3 4.4 5.5</div> <div>6.6 7.7 2.2 3.3 4.4</div> <div>5.5 1.1 7.7 8.8 3.3</div> <div>2.2 5.5 3.3 2.2 4.4</div> <div>6.6 4.4 3.3 2.2 7.7</div> <div>2.2 1.1 9.9 5.5 4.4</div> <div>3.3 2.2 9.9 6.6 9.9</div>	<div>mtrxA =</div> <div><div>1.10 2.20 3.30 4.40 5.50 6.60 7.60</div><div>3.30 4.40 4.40 5.50 1.10 7.70 8.80</div><div>5.50 6.60 5.50 3.30 2.20 4.40 6.60</div><div>7.70 8.80 2.20 7.70 2.20 1.10 9.90</div><div>9.90 1.10 3.30 2.20 9.90 6.60 9.90</div></div> <div>mtrxB =</div> <div><div>1.10 0.00 0.00 0.00 0.00 1.10 6.60</div><div>0.00 1.10 0.00 0.00 0.00 2.20 7.70</div><div>0.00 0.00 1.10 0.00 0.00 3.30 8.80</div><div>0.00 0.00 0.00 1.10 0.00 4.40 9.90</div><div>0.00 0.00 0.00 0.00 1.10 5.50 9.90</div></div> <div>mtrxC =</div> <div><div>1.10 2.20 3.30 4.40 5.50</div><div>6.60 7.70 2.20 3.30 4.40</div><div>5.50 1.10 7.70 8.80 3.30</div><div>2.20 5.50 3.30 2.20 4.40</div><div>6.60 4.40 3.30 2.20 7.70</div><div>2.20 1.10 9.90 5.50 4.40</div><div>3.30 2.20 9.90 6.60 9.90</div></div>	<div>mtrxD = mtrxA.add(mtrxB) =</div> <div><div>2.20 2.20 3.30 4.40 5.50 7.70 14.20</div><div>3.30 5.50 4.40 5.50 1.10 9.90 16.50</div><div>5.50 6.60 6.60 3.30 2.20 7.70 15.40</div><div>7.70 8.80 2.20 8.80 2.20 5.50 19.80</div><div>9.90 1.10 3.30 2.20 11.00 12.10 19.80</div></div> <div>mtrxE = mtrxA.sub(mtrxB) =</div> <div><div>0.00 2.20 3.30 4.40 5.50 5.50 1.00</div><div>3.30 3.30 4.40 5.50 1.10 5.50 1.10</div><div>5.50 6.60 4.40 3.30 2.20 1.10 -2.20</div><div>7.70 8.80 2.20 6.60 2.20 -3.30 0.00</div><div>9.90 1.10 3.30 2.20 8.80 1.10 0.00</div></div> <div>mtrxF = mtrxA.mul(mtrxC) =</div> <div><div>119.46 95.37 207.13 149.38 192.61</div><div>122.21 108.90 239.58 182.71 205.70</div><div>133.10 116.16 202.07 174.24 193.60</div><div>145.20 162.14 203.28 175.45 242.00</div><div>153.67 118.58 263.78 204.49 283.14</div></div>
(a) 입력 데이터 파일 (Matrix_data.txt)	(b) 출력 데이터 파일 (output.txt)	