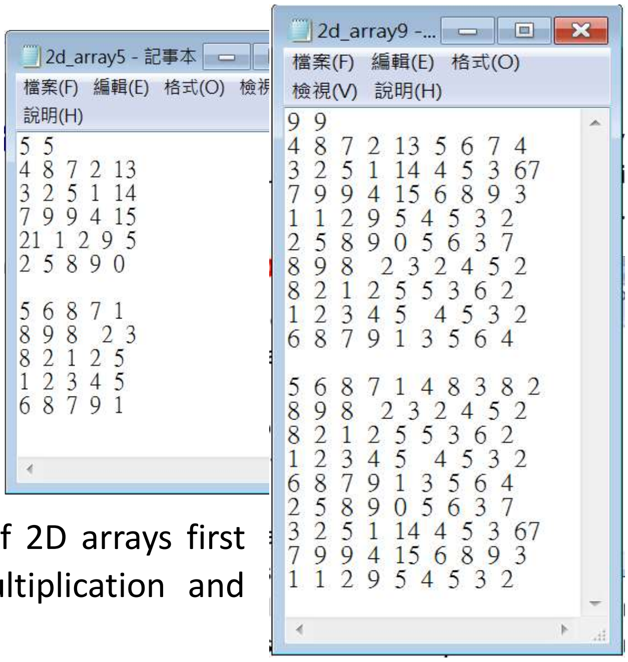
Program Homework #3
multidimensional array allocation and applications.
(due: PM:17:00, April 10, 2020)

- **Homework:**

In the lecture hours of chapter 2 array, we demonstrated programs for 2D and 3D dynamic memory allocation. Two 5x5 and two 9x9 two-dimensional arrays are provided. Based on these program segments, you are encouraged to write programs that can read the dimension of 2D arrays first and then perform matrix multiplication and matrix summation.



```
5 5
4 8 7 2 13
3 2 5 1 14
7 9 9 4 15
21 1 2 9 5
2 5 8 9 0

5 6 8 7 1
8 9 8  2 3
8 2 1 2 5
1 2 3 4 5
6 8 7 9 1
```

```
9 9
4 8 7 2 13 5 6 7 4
3 2 5 1 14 4 5 3 67
7 9 9 4 15 6 8 9 3
1 1 2 9 5 4 5 3 2
2 5 8 9 0 5 6 3 7
8 9 8  2 3 2 4 5 2
8 2 1 2 5 5 3 6 2
1 2 3 4 5  4 5 3 2
6 8 7 9 1 3 5 6 4

5 6 8 7 1 4 8 3 8 2
8 9 8  2 3 2 4 5 2
8 2 1 2 5 5 3 6 2
1 2 3 4 5  4 5 3 2
6 8 7 9 1 3 5 6 4
2 5 8 9 0 5 6 3 7
3 2 5 1 14 4 5 3 67
7 9 9 4 15 6 8 9 3
1 1 2 9 5 4 5 3 2
```

- The main program and reference program that dynamically allocates a simple 2D array with dimension Height x Width

Main program

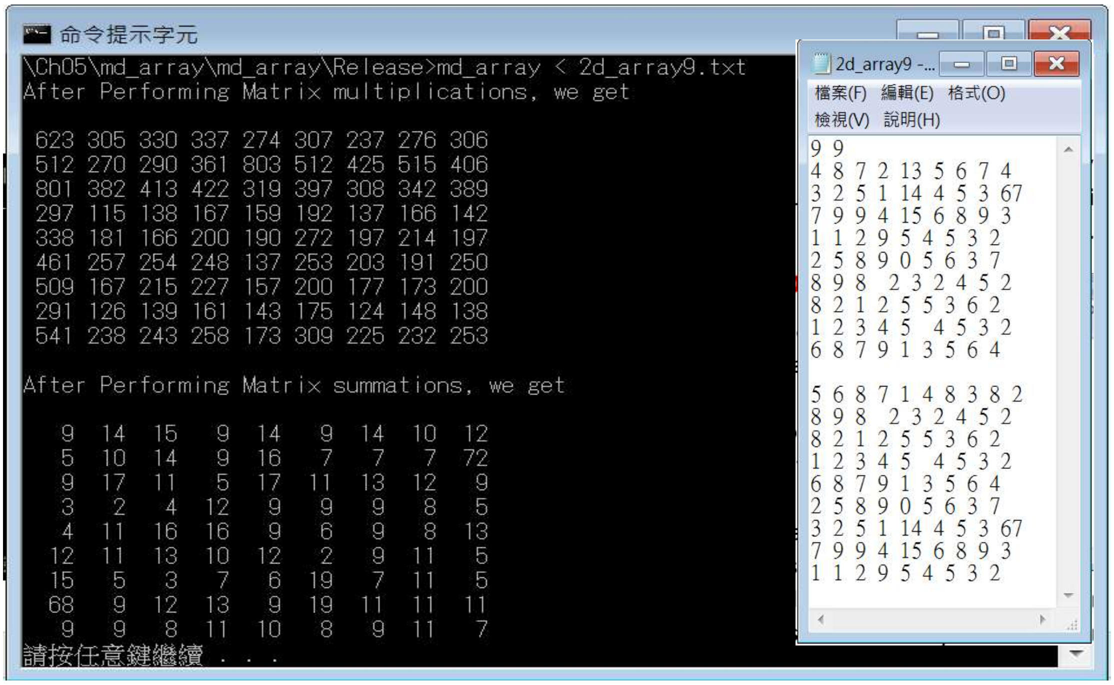
```
int main() {
    double **aptr,**bptr,**cptr;  \\ C = A*B
    int Row, Col;
        cin >> Row>> Col;

        aptr=set_2D_array(Row, Col);
        bptr=set_2D_array(Row, Col);
        cptr = multiplication_2D_array(aptr,bptr,Row, Col);
        print_2D_array(cptr, Row, Col);
        print_2D_array(sum_2D_array(aptr,bptr, Row, Col),Row, Col);
        system("pause");
  return 0; _
};
```

2D array memory allocation

```
#define HEIGHT 5
#define WIDTH 3
int main() {
   double **p2DArray;
// Allocate memory
   p2DArray = new double*[HEIGHT];
   for (int i = 0; i < HEIGHT; ++i)
          p2DArray[i] = new double[WIDTH];

    // Assign values
    p2DArray[0][0] = 3.6;
    p2DArray[1][2] = 4.0;
    // De-Allocate memory to prevent memory leak
  for (int i = 0; i < HEIGHT; ++i)
            delete [] p2DArray[i];

  delete [] p2DArray;

   return 0;
  }
```

Execution results: (The md_array.exe is provided for you for reference)

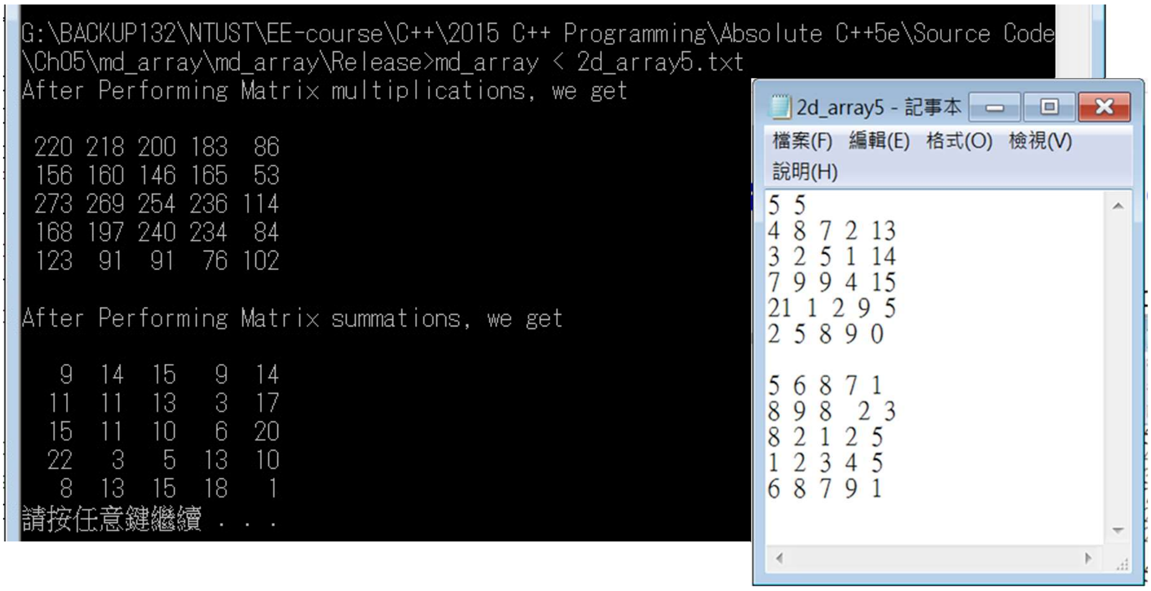
For the 9x9 2d array, the multiplication and summation results would be



```
\Ch05\md_array\md_array\Release>md_array < 2d_array9.txt
After Performing Matrix multiplications, we get

 623 305 330 337 274 307 237 276 306
 512 270 290 361 803 512 425 515 406
 801 382 413 422 319 397 308 342 389
 297 115 138 167 159 192 137 166 142
 338 181 166 200 190 272 197 214 197
 461 257 254 248 137 253 203 191 250
 509 167 215 227 157 200 177 173 200
 291 126 139 161 143 175 124 148 138
 541 238 243 258 173 309 225 232 253

After Performing Matrix summations, we get

   9  14  15   9  14   9  14  10  12
   5  10  14   9  16   7   7   7  72
   9  17  11   5  17  11  13  12   9
   3   2   4  12   9   9   9   8   5
   4  11  16  16   9   6   9   8  13
  12  11  13  10  12   2   9  11   5
  15   5   3   7   6  19   7  11   5
  68   9  12  13   9  19  11  11  11
   9   9   8  11  10   8   9  11   7
請按任意鍵繼續 . . .
```

```
9 9
4 8 7 2 13 5 6 7 4
3 2 5 1 14 4 5 3 67
7 9 9 4 15 6 8 9 3
1 1 2 9 5 4 5 3 2
2 5 8 9 0 5 6 3 7
8 9 8  2 3 2 4 5 2
8 2 1 2 5 5 3 6 2
1 2 3 4 5  4 5 3 2
6 8 7 9 1 3 5 6 4

5 6 8 7 1 4 8 3 8 2
8 9 8  2 3 2 4 5 2
8 2 1 2 5 5 3 6 2
1 2 3 4 5  4 5 3 2
6 8 7 9 1 3 5 6 4
2 5 8 9 0 5 6 3 7
3 2 5 1 14 4 5 3 67
7 9 9 4 15 6 8 9 3
1 1 2 9 5 4 5 3 2
```

For the 5x5 2d arrays, the results would be



```
G:\BACKUP132\NTUST\EE-course\C++\2015 C++ Programming\Absolute C++5e\Source Code
\Ch05\md_array\md_array\Release>md_array < 2d_array5.txt
After Performing Matrix multiplications, we get

 220 218 200 183  86
 156 160 146 165  53
 273 269 254 236 114
 168 197 240 234  84
 123  91  91  76 102

After Performing Matrix summations, we get

   9  14  15   9  14
  11  11  13   3  17
  15  11  10   6  20
  22   3   5  13  10
   8  13  15  18   1
請按任意鍵繼續 . . .
```

```
5 5
4 8 7 2 13
3 2 5 1 14
7 9 9 4 15
21 1 2 9 5
2 5 8 9 0

5 6 8 7 1
8 9 8  2 3
8 2 1 2 5
1 2 3 4 5
6 8 7 9 1
```

- ## Matrix multiplication example:

http://www.c4learn.com/c-programs/c-program-to-multiply-two-3-x-3.html

- Extra Bonus will be given if you can process the 3x3x3 array summation.