



COMPUTER ENGINEERING

CHIANG MAI UNIVERSITY
CPE102 Computer Programming
Problem Set 13

1. จงศึกษาวิธีการเรียงลำดับแบบ Selection Sort แล้วเขียน Function `selectionSort()` ที่มี parameter เป็น Array ที่ต้องการเรียงลำดับ `d[]` และจำนวนเต็ม `N` ซึ่งเป็นความยาวของ Array นั้น โดย Function `selectionSort()` นี้จะทำการเรียงลำดับ Array `d[]` จากน้อยไปมาก

Source Code ส่วนของ Function `selectionSort()` ที่ต้องการ (ให้เติมส่วนของ Function ให้สมบูรณ์ โดยต้องใช้ Function `swap()` ที่มีให้)

```
template <typename T>
void swap(T d[],int x,int y){
    T temp = d[x];
    d[x] = d[y];
    d[y] = temp;
}

template <typename T>
void selectionSort(T d[],int N){

//Write your code here.

}
```

2. เขียน Function `randomMatrix()` ที่รับอินพุตเป็น Array 2 มิติ ที่มีขนาด 10x10 โดย `randomMatrix()` จะทำการสุ่มค่าใน Array นั้น ซึ่งค่าที่สุ่มจะเป็นเลขทศนิยม 3 ตำแหน่ง ตั้งแต่ในช่วง 0.000 – 9.999
3. จงเขียนโปรแกรมเพื่อคำนวณหาค่าผลคูณของ Matrix สองตัว โดยให้เพิ่มเติมส่วนของ Function ที่ขาดหายไปจาก Source Code ที่แสดงอยู่ในหน้าถัดไปนี้ ซึ่งจะมี Function ที่ยังไม่ได้ถูกนิยามอยู่ 3 Function คือ **`inputMatrix()`**, **`matrixMultiply()`** และ **`showMatrix()`** โดยที่

- **`inputMatrix()`** จะเป็น Function ที่ใช้รับอินพุต Matrix ที่จะนำมาคูณกันจากผู้ใช้ แล้วเก็บค่าที่ผู้ใช้อินพุตไว้ใน Array 2 มิติ จะเห็นได้ว่าใน Function **`main()`** Function **`inputMatrix()`** จะถูกเรียกใช้ 2 ครั้ง เนื่องจากการคูณกันต้องรับอินพุต Matrix ทั้งสิ้น 2 ตัวด้วยกัน (เก็บไว้ในตัวแปรชื่อ **`A`** และ **`B`**)
- **`matrixMultiply()`** จะเป็น Function ที่ใช้ในการคำนวณผลคูณของ Matrix สองตัว แล้วเก็บค่าไว้ใน Matrix ผลลัพธ์อีกตัวหนึ่ง สมมติว่า **`A`** และ **`B`** เป็น Matrix สองตัวที่จะนำมาคูณกันซึ่งมีขนาดเท่ากับ **`N × N`** (Source Code นี้กำหนดให้ **`N`** มีค่าเท่ากับ 3) และ **`C`** เป็น Matrix ที่เป็นผลลัพธ์ของการคูณ ที่มีขนาดเท่ากับ **`N × N`** เท่ากัน โดยสมาชิก แต่ละตัวของ **`C`** จะถูกนิยามจาก

$$C[i][j] = \sum_{k=0}^{N-1} (A[i][k] \times A[k][j])$$

- **showMatrix()** จะเป็น Function ที่ใช้ในการแสดงค่า Matrix ที่ต้องการ โดยจะพิมพ์ตัวเลขมาบรรทัดละ 1 แถวของ Matrix โดยระหว่างตัวเลขในแต่ละแถวจะมีช่องว่างคั่นอยู่ 1 ช่อง

ให้ นศ. เขียนฟังก์ชัน **inputMatrix()**, **matrixMultiply()** และ **showMatrix()** ที่ขาดหายไปนี้ให้สอดคล้องกับ Function prototype ที่กำหนดให้ และตัวอย่างผลลัพธ์ที่ให้ในหน้าถัดไป (ถ้าเปลี่ยนเลขของค่า **N** ใน Source Code แล้วกด Run โปรแกรมต้องยังทำงานได้อย่างถูกต้องกับค่า **N** ใหม่)

Source Code ในส่วนต้นและ Function **main()**

```
#include<iostream>
using namespace std;

const int N = 3;

void inputMatrix(double [][][N]);

void matrixMultiply(const double [][][N], const double [][][N], double [][][N]);

void showMatrix(const double [][][N]);

int main(){
    double A[N][N], B[N][N], C[N][N];
    cout << "Input the first matrix.....\n";
    inputMatrix(A);
    cout << "Input the second matrix.....\n";
    inputMatrix(B);
    matrixMultiply(A,B,C);
    cout << "Result is \n";
    showMatrix(C);
    return 0;
}

// Write definition of inputMatrix(),matrixMultiply() and showMatrix() here
```

ตัวอย่างการอินพุตและผลลัพธ์ที่ได้ (ตรงที่ขีดเส้นใต้คืออินพุตจากผู้ใช้)

```
Input the first matrix.....
Row 1: 1 2 3
Row 2: 4 5 6
Row 3: 7 8 9
Input the second matrix.....
Row 1: -2 -3 0
Row 2: 1.5 2 4.7
Row 3: 0 0.8 -1.2
Result is
1 3.4 5.8
-0.5 2.8 16.3
-2 2.2 26.8
```

Input the first matrix.....

Row 1: 0 1 0

Row 2: 1 2 4

Row 3: 7 7 7

Input the second matrix.....

Row 1: 0 5 9.1

Row 2: -0.1 0 0

Row 3: 0 0 5

Result is

-0.1 0 0

-0.2 5 29.1

-0.7 35 98.7