

School of Mechanical and Control Engineering

Handong Global University, Rep. Korea

Smart Sensor Systems Lab.



수치해석 과제제출 코드 템플릿 소개 및 활용법 안내

Mod. 2021.10.07

목차

1. 과제 템플릿 제공 배경
2. 템플릿 다운로드 및 샘플 구동
3. 템플릿 설명 및 이용법
4. 기타 안내사항

과제 템플릿 제공 배경

과제 채점시 고려사항 :

- 주어진 역학/회로문제 등을 잘 풀었는가?
- 함수가 주어진 과제문제의 행렬에 대해 올바르게 구동되는가?
- 과제 문제 외 임의의 행렬에 대해서도 함수가 정상적인 기능을 수행하는가?

기존 채점방식의 어려움 :

- 과제문제 오답 시 행렬과 함수 중 어디서 문제가 발생했는지 파악 어려움
- 각기 다른 학생들의 코드에 함수 검증을 위한 임의의 행렬 입력 시 어려움이 있음

해결방안:

- 절대경로 및 파일명을 지정, 이로부터 행렬에 접근

템플릿 다운로드 및 샘플 구동

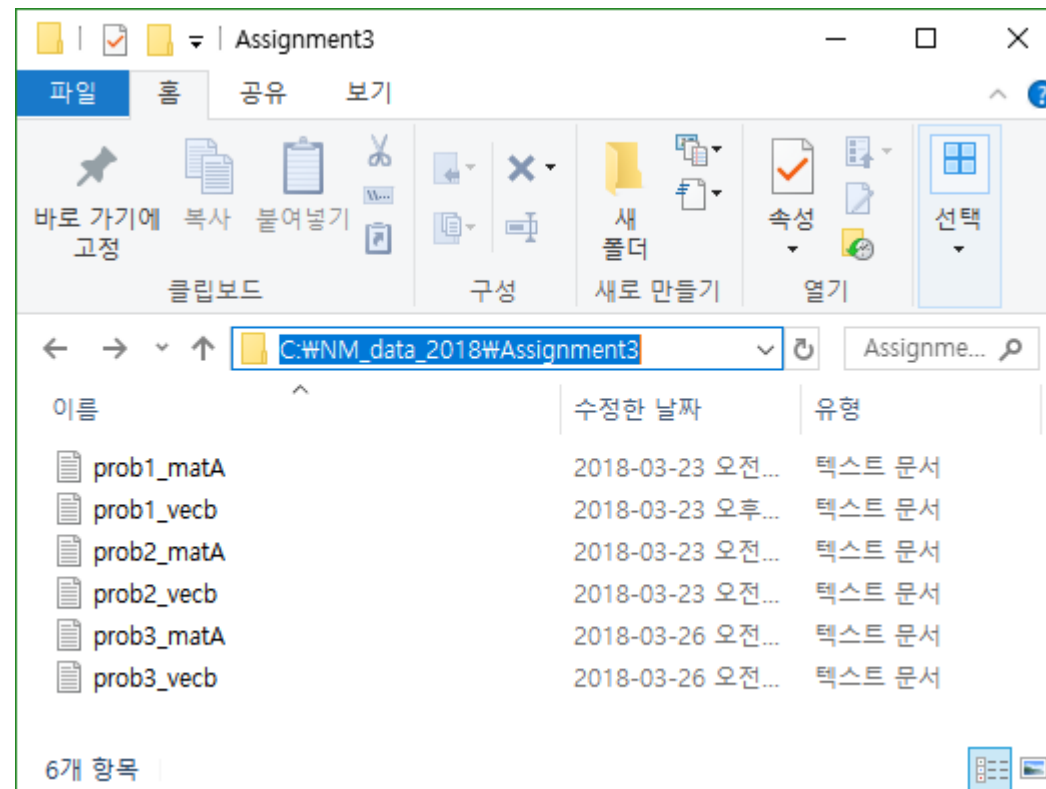
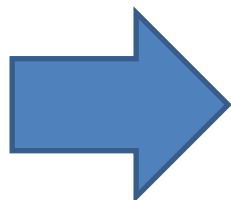
템플릿 다운로드 및 샘플 구동

사용할 행렬 데이터 다운로드 및 지정경로에 복사

LMS 자료실에 첨부된
NP_data_2021.zip 파일을

C:\W

에 압축해제



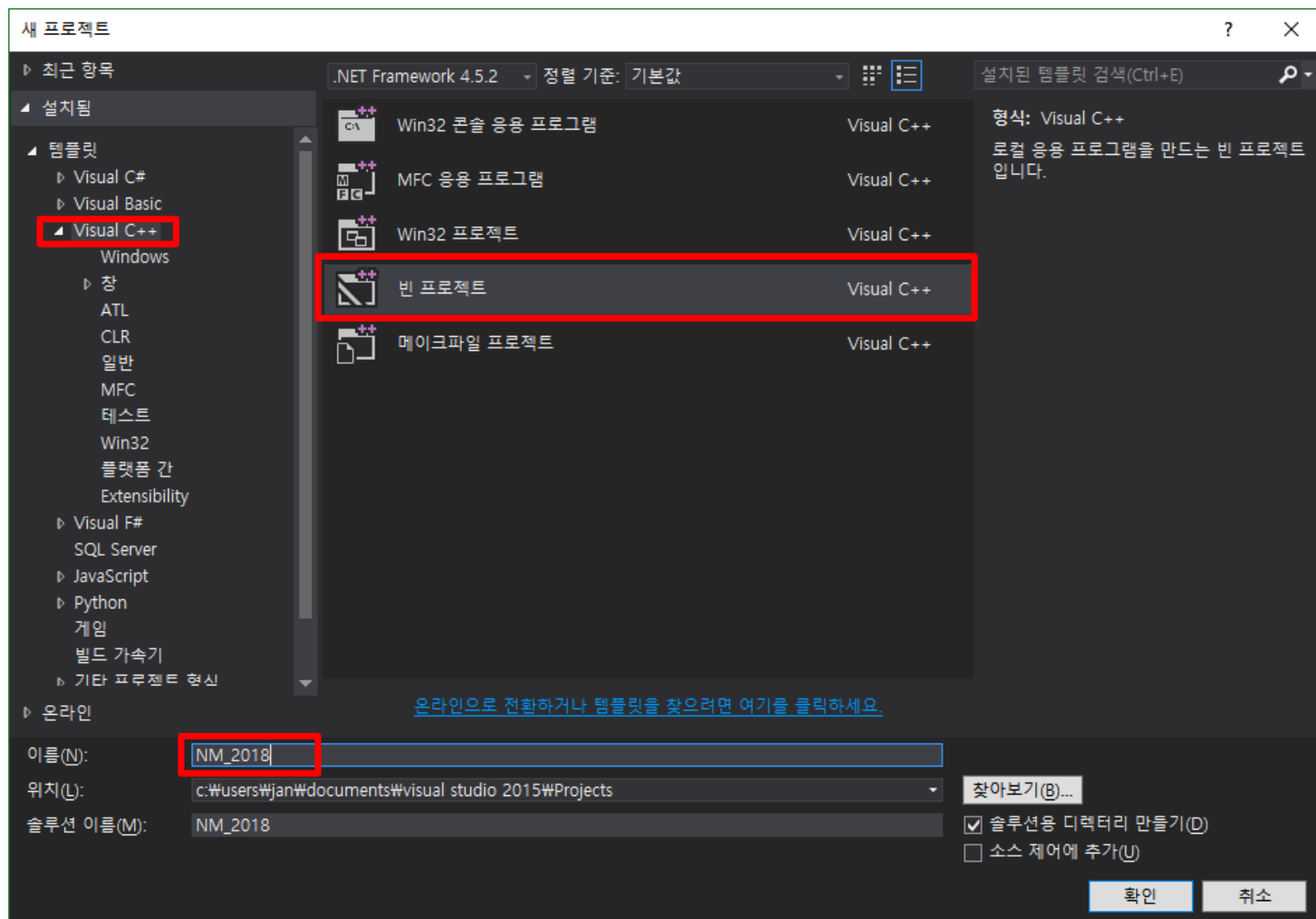
텍스트 문서의 경로가 아래와 같은지 필히 확인

C:\WNP_data_2021\Assignment2

아래와 같은 경우가 발생하지 않도록 주의
C:\WNP_data_2021\NP_data_2021\Assignment2

새 프로젝트 생성

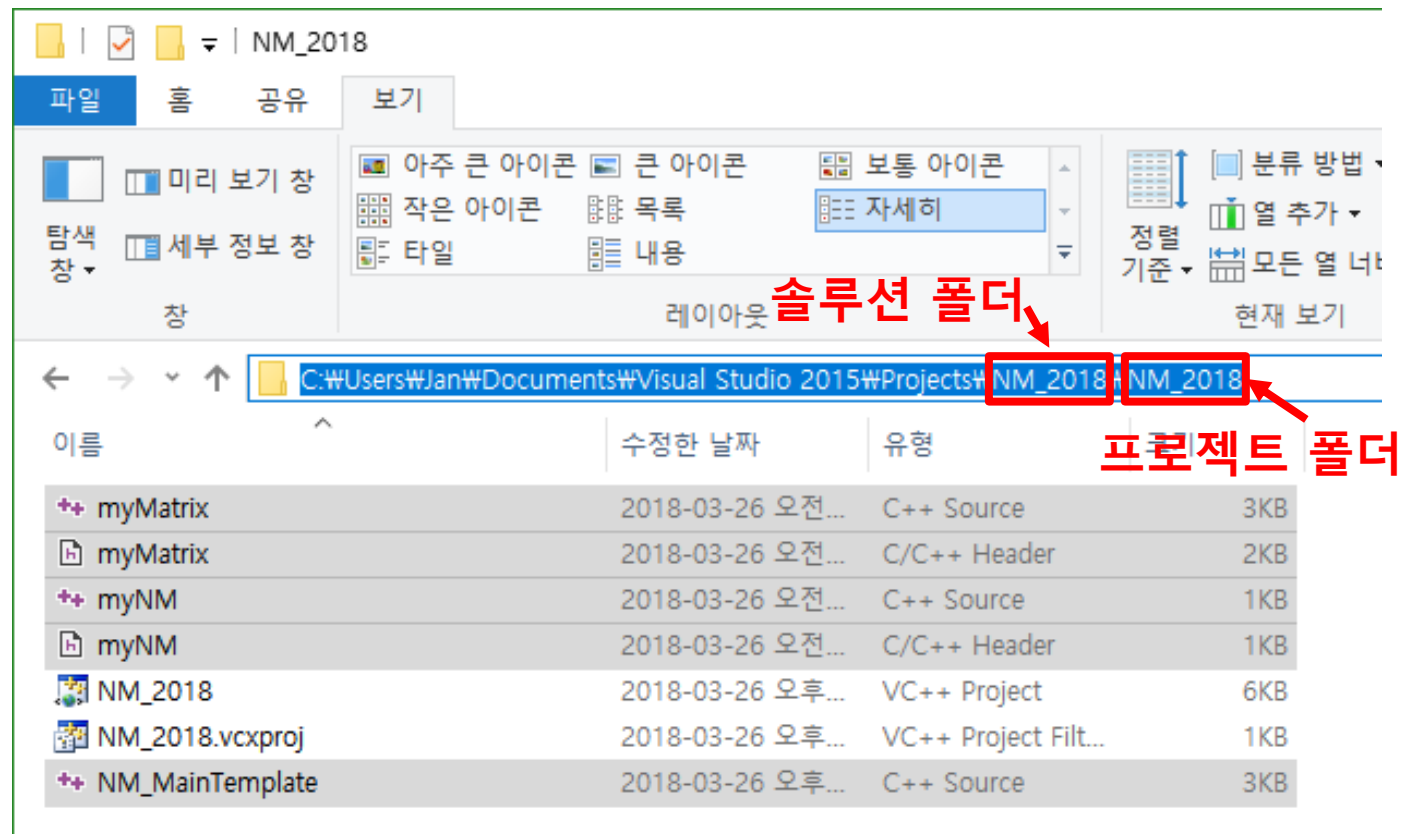
- 1) Visual studio 실행
- 2) 파일→새로 만들기→프로젝트 클릭
(or Ctrl+Shift+N)
- 3) 우측과 같이 설정 후 확인 클릭
(프로젝트 명 자유롭게 변경)



템플릿 다운로드 및 샘플 구동

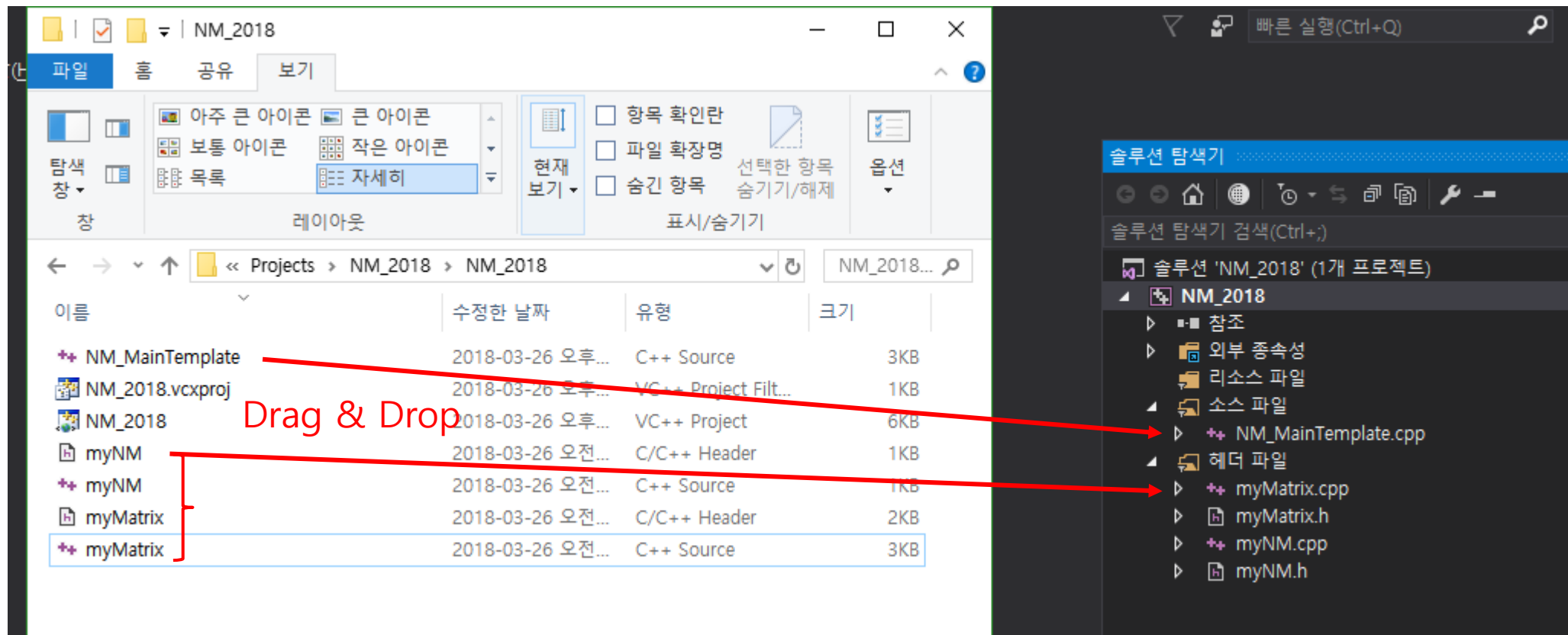
템플릿 코드 다운로드 및 지정 경로에 복사

- 1) LMS 자료실에 첨부된 **NP_templateCode.zip** 파일을 압축해제
- 2) 해당 5개의 파일을
방금 생성한 프로젝트 폴더에 복사&붙여넣기
(※ 솔루션 폴더가 아닌 프로젝트 폴더임)



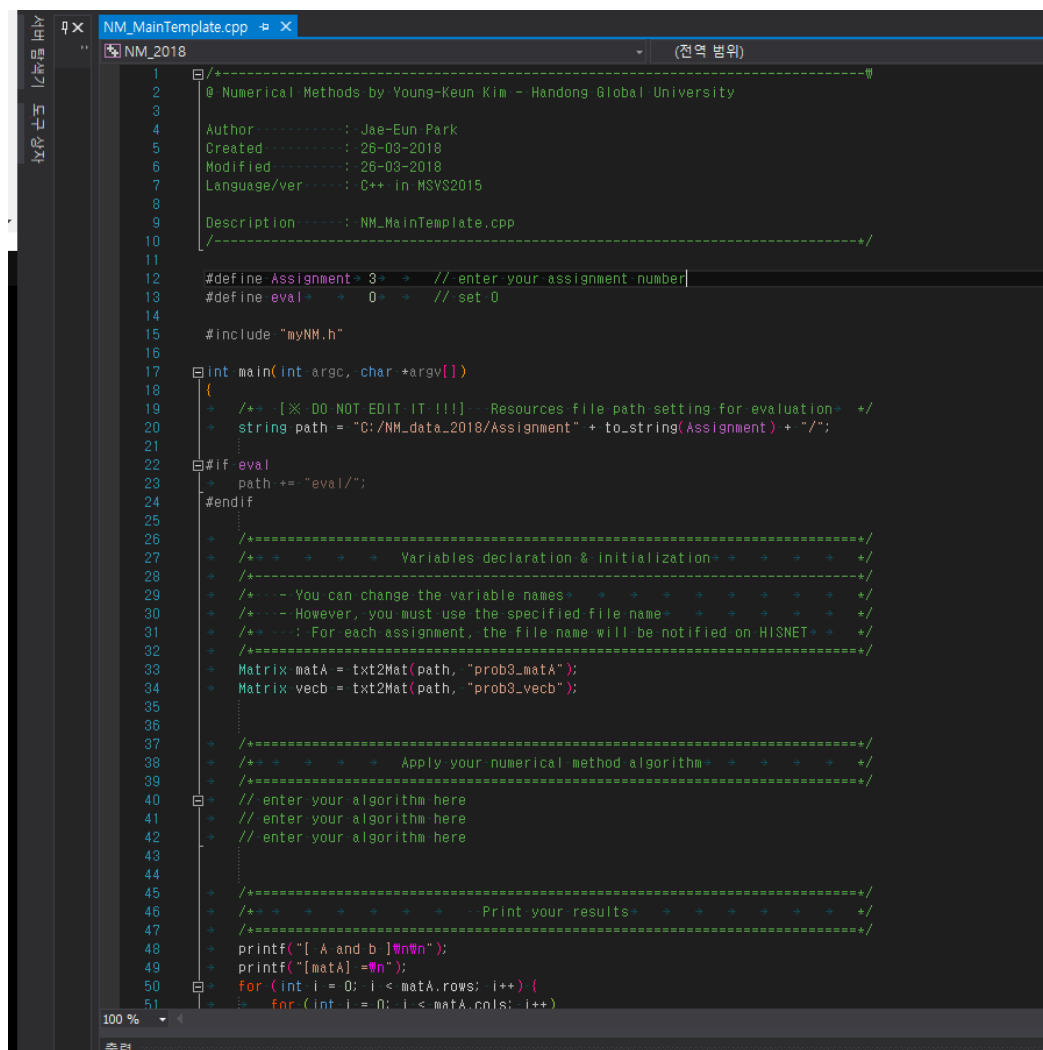
템플릿 다운로드 및 샘플 구동

템플릿 코드를 Visual studio 프로젝트에 드래그 & 드랍

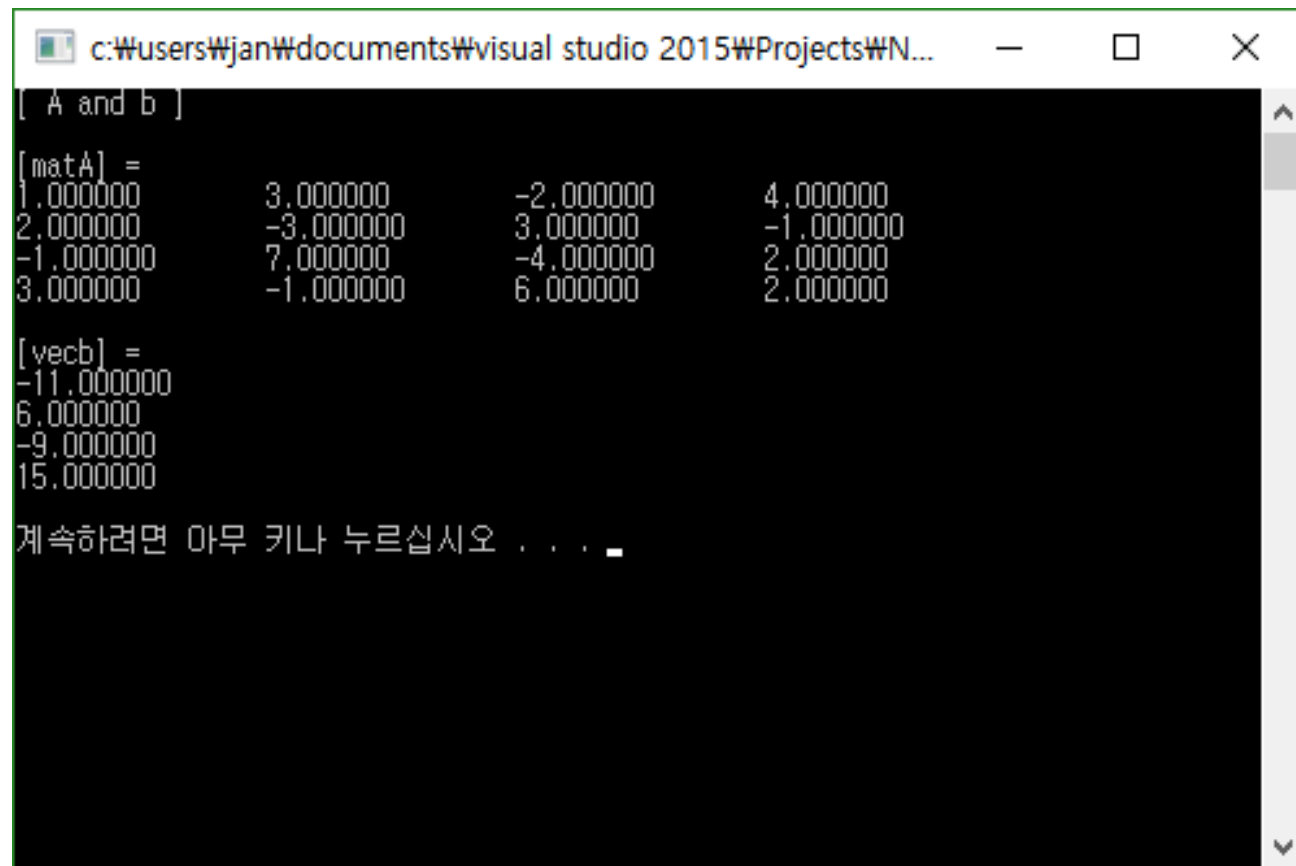


템플릿 다운로드 및 샘플 구동

메인함수(NP_MainTemplate.cpp) 확인 및 빌드



```
1  /*-----*/
2  @ Numerical Methods by Young-Keun Kim - Handong Global University
3
4  Author:-----: Jae-Eun Park
5  Created:-----: 26-03-2018
6  Modified:-----: 26-03-2018
7  Language/version: C++ in MSVS2015
8
9  Description:-----: NP_MainTemplate.cpp
10 /*-----*/
11
12 #define Assignment 3 // enter your assignment number
13 #define eval 0 // set 0
14
15 #include "myNM.h"
16
17 int main(int argc, char *argv[])
18 {
19     /*----- [X DO NOT EDIT IT !!!] Resources file path setting for evaluation -----*/
20     string path = "C:/NM_data_2018/Assignment" + to_string(Assignment) + "/";
21
22     #if eval
23     path += "eval/";
24     #endif
25
26     /*----- Variables declaration & initialization -----*/
27     /*----- You can change the variable names -----*/
28     /*----- However, you must use the specified file name -----*/
29     /*----- For each assignment, the file name will be notified on HISNET -----*/
30
31     Matrix matA = txt2Mat(path, "prob3_matA");
32     Matrix vecb = txt2Mat(path, "prob3_vecb");
33
34     /*----- Apply your numerical method algorithm -----*/
35
36     // enter your algorithm here
37     // enter your algorithm here
38     // enter your algorithm here
39
40     /*----- Print your results -----*/
41     printf("[ A and b ]\n");
42     printf("[matA] =\n");
43     for (int i = 0; i < matA.rows; i++) {
44         for (int j = 0; j < matA.cols; j++) {
```



```
c:\Users\Wjan\documents\visual studio 2015\Projects\WN...
[ A and b ]

[matA] =
1.000000      3.000000     -2.000000      4.000000
2.000000     -3.000000      3.000000     -1.000000
-1.000000      7.000000     -4.000000      2.000000
3.000000     -1.000000      6.000000      2.000000

[vecb] =
-11.000000
6.000000
-9.000000
15.000000

계속하려면 아무 키나 누르십시오 . . .
```

빌드 결과

Main 함수

템플릿 설명 및 이용법

템플릿 설명 및 이용법

행렬 데이터 텍스트 파일 생성법

과제 업로드 이후 수일 내로 LMS에 필요한 행렬 텍스트 파일 및 파일명 공지 → 과제에 반영

Assignment3 데이터 파일 경로/파일명을 아래와 같이 공지합니다.

[파일 경로]

C:\NM_data_2018\Assignment3

[파일명]

Q1.
matrix A : prob1_mata
vector b : prob1_vecb

Q2.
matrix A : prob1_mata
vector b : prob1_vecb

Q2.
matrix A : prob1_mata
vector b : prob1_vecb

(※ 모든 파일의 확장자는 ".txt" 입니다.)

경로 확인

파일명(복사&붙여넣기)

파일 유형 확인

이름	수정한 날짜	유형
prob1_mata	2018-03-23 오전...	텍스트 문서
prob1_vecb	2018-03-23 오후...	텍스트 문서
prob2_mata	2018-03-23 오전...	텍스트 문서
prob2_vecb	2018-03-23 오전...	텍스트 문서
prob3_mata	2018-03-26 오전...	텍스트 문서
prob3_vecb	2018-03-26 오전...	텍스트 문서

LMS 공지사항 확인 (매 과제마다 새로 공지)

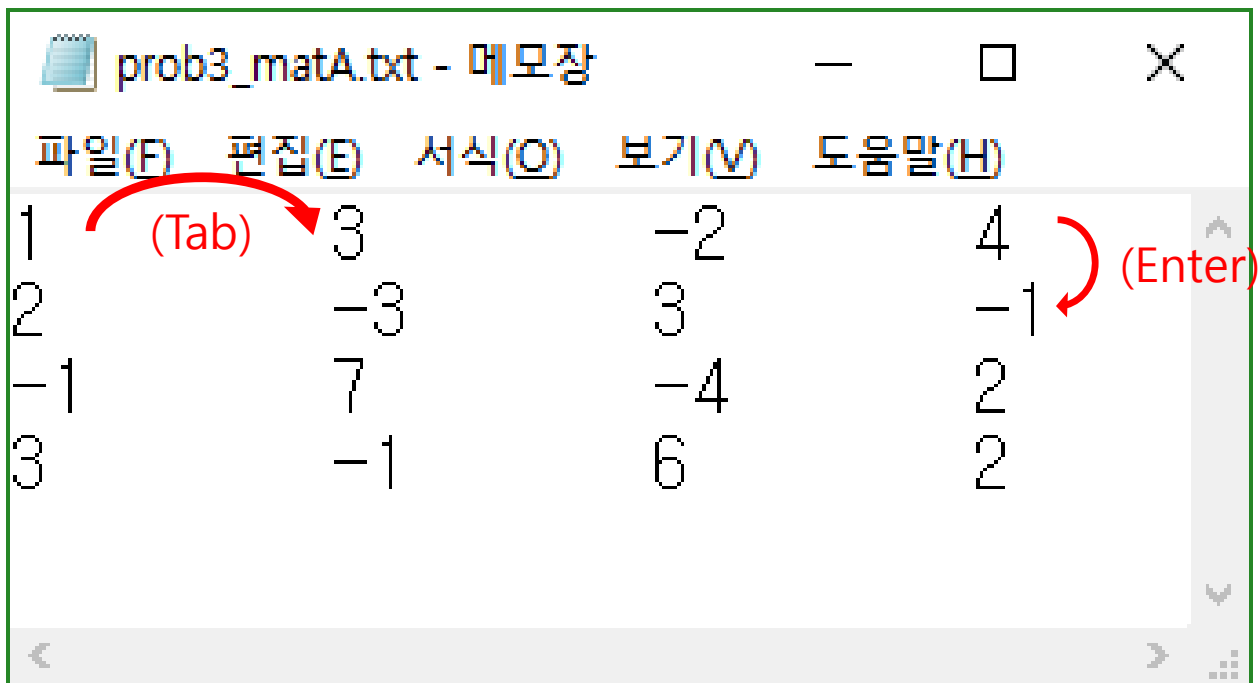
공지사항 참조하여 위와 같이

- 경로 설정 및 확인
- 텍스트 파일 생성 및 이름설정

템플릿 설명 및 이용법

텍스트 파일 내 행렬 원소 입력 방법

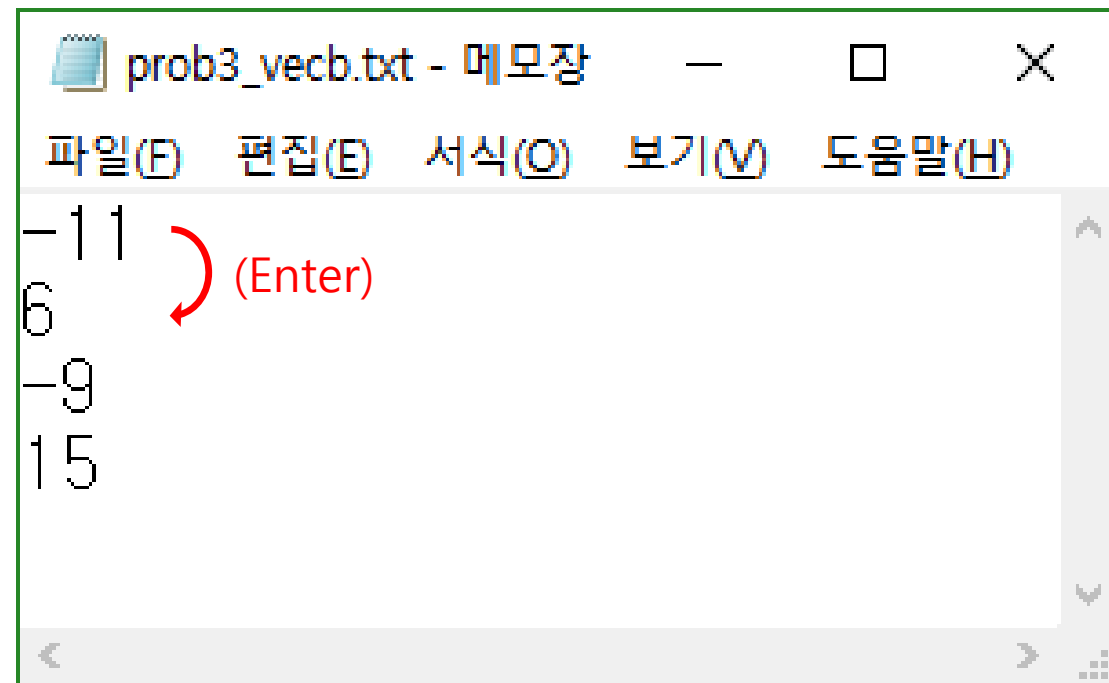
숫자의 길이와는 무관하게 행단위 변경시 : enter 키 / 열단위 변경시 : Tab 키 이용 , 완료 후 저장



1	3	-2	4
2	-3	3	-1
-1	7	-4	2
3	-1	6	2

입력 :

1 → tab → 3 → tab → -2 → tab → 4 → enter
→ 2 → tab → -3 → tab → 3 → tab → -1 → enter
→ -1 → tab → 7 → tab → -4 → tab → 2 → enter
→ 3 → tab → -1 → tab → 6 → tab → 2 → ctrl+s (저장)



-11
6
-9
15

입력 :

-11 → enter → 6 → enter
→ -9 → enter → 15 → ctrl+s (저장)

Main함수 구간별 속지사항 1

코드 개요
(적절하게 변경 가능)

과제(assignment) 번호 입력

'0'으로 고정
(평가 시 활용되는 파라미터)

행렬을 불러들일
텍스트 파일 경로
(변경 불허)

```
1  /*-----*/
2  @-Numerical-Methods-by-Young-Keun-Kim--Handong-Global-University
3
4  Author-----:Jae-Eun-Park
5  Created-----:26-03-2018
6  Modified-----:26-03-2018
7  Language/ver-----:C++-in-MSVS2015
8
9  Description-----:NM_MainTemplate.cpp
10 /*-----*/
11
12 #define Assignment 3 //enter your assignment number
13 #define eval 0 //set 0
14
15 #include "myNM.h"
16
17 int main(int argc, char*argv[])
18 {
19     /*- [※-DO-NOT-EDIT-IT-!!!]-Resources-file-path-setting-for-evaluation-*/
20     string path="C:/NM_data_2018/Assignment"+to_string(Assignment)+"/";
21     ...
22     #if eval
23     path+="eval/";
24     #endif
25     ...
26 }
```

템플릿 설명 및 이용법

Main함수 구간별 속지사항 2

텍스트파일로부터 행렬 생성
(파일명은 공지사항 참조하여 기재)

수치해석 알고리즘 함수 적용

결과 출력
(무엇에 대한 결과인지
명확히 표기해야 함)

선언된 행렬들의 메모리 할당 해제

```
26  → /*=====*/
27  → /*→ → → → Variables declaration & initialization→ → → → */
28  → /*-----*/
29  → /*---- You can change the variable names→ → → → → */
30  → /*---- However, you must use the specified file name→ → → → → */
31  → /*→ ----: For each assignment, the file name will be notified on HISNET→ → */
32  → /*=====*/
33  → Matrix matA = txt2Mat(path, "prob3_matA");
34  → Matrix vecb = txt2Mat(path, "prob3_vecb");
35
36
37  → /*=====*/
38  → /*→ → → → → Apply your numerical method algorithm→ → → → → */
39  → /*=====*/
40  → //enter your algorithm here
41  → //enter your algorithm here
42  → //enter your algorithm here
43
44
45  → /*=====*/
46  → /*→ → → → → Print your results→ → → → → */
47  → /*=====*/
48  → printf("[ A and b ]\n\n");
49  → printf("[ matA ] =\n");
50  → for (int i = 0; i < matA.rows; i++) {
51  →     for (int j = 0; j < matA.cols; j++)
52  →         printf("%f\t", matA.at[i][j]);
53  →     printf("\n");
54  → }
55  → printf("\n[ vecb ] =\n");
56  → for (int i = 0; i < vecb.rows; i++)
57  →     printf("%f\n", vecb.at[i][0]);
58  → printf("\n");
59
60
61  → /*=====*/
62  → /*→ → → → → Deallocate memory→ → → → → */
63  → /*=====*/
64  → freeMat(matA); → freeMat(vecb);
65
```

템플릿 설명 및 이용법

작성된 Main함수 예시

텍스트파일로부터
행렬 생성

수치해석 알고리즘
함수 적용

결과 출력
(무엇에 대한 결과인지
명확히 표기해야 함)

메모리 할당 해제

```
/*=====*/
/*--> --> --> --> Variables declaration & initialization--> --> --> -->*/
/*=====*/
/*----- You can change the variable names--> --> --> --> --> --> -->*/
/*----- However, you must use the specified file name--> --> --> --> -->*/
/*----- For each assignment, the file name will be notified on HISNET--> --> -->*/
/*=====*/
Matrix matA = txt2Mat(path, "Prob3_matA");
Matrix vecb = txt2Mat(path, "Prob3_vecb");

/*=====*/
/*--> --> --> --> Apply your numerical method algorithm--> --> --> -->*/
/*=====*/
Matrix x_g = solveLinear(matA, vecb, "g");
Matrix x_gj = solveLinear(matA, vecb, "gj");
Matrix x_lu = solveLinear(matA, vecb, "LU");

Matrix invA_gj = inv(matA, "gj");
Matrix invA_lu = inv(matA, "LU");

/*=====*/
/*--> --> --> --> Print your results--> --> --> -->*/
/*=====*/
printf("[ A and b ]\n\n");
PrintMat(matA); --> PrintMat(vecb);

printf("\n\n[ solution of Ax=b ]\n\n");
printMat(x_g, "x, using Gauss elimination");
printMat(x_gj, "x, using Gauss-Jordan elimination");
printMat(x_lu, "x, using LU decomposition");

printf("\n\n[ inverse matrix of A ]\n\n");
printMat(invA_gj, "inv(A), using Gauss-Jordan elimination");
printMat(invA_lu, "inv(A), using LU decomposition");

/*=====*/
/*--> --> --> --> Deallocate memory--> --> --> -->*/
/*=====*/
freeMat(matA); --> freeMat(vecb);
freeMat(x_g); --> freeMat(x_gj); --> freeMat(x_lu);
freeMat(invA_gj); --> freeMat(invA_lu);
```

```
[ A and b ]
[ matA ] =
1.000000    3.000000   -2.000000    4.000000
2.000000   -3.000000    3.000000   -1.000000
-1.000000    7.000000   -4.000000    2.000000
3.000000   -1.000000    6.000000    2.000000

[ vecb ] =
-11.000000
6.000000
-9.000000
15.000000

[ solution of Ax=b ]
[ x, using Gauss elimination ] =
-2.000000
1.000000
4.000000
-1.000000

[ x, using Gauss-Jordan elimination ] =
-2.000000
1.000000
4.000000
-1.000000

[ x, using LU decomposition ] =
-2.000000
1.000000
4.000000
-1.000000

[ inverse matrix of A ]
[ inv(A), using Gauss-Jordan elimination ] =
0.171053    0.815789    0.250000   -0.184211
-0.144737    0.078947    0.250000    0.078947
-0.184211   -0.263158    0.000000    0.236842
0.223684   -0.394737   -0.250000    0.105263

[ inv(A), using LU decomposition ] =
0.171053    0.815789    0.250000   -0.184211
-0.144737    0.078947    0.250000    0.078947
-0.184211   -0.263158    0.000000    0.236842
0.223684   -0.394737   -0.250000    0.105263

계속하려면 아무 키나 누르십시오 . . .
```

Assignment #3, Problem3 활용 예시

출력 결과

기타 안내사항

지정된 형식의 구조체와 다른 형태로 코드작성을 희망할 경우

- 대부분의 함수는 본인의 것을 사용해도 무방
- 그러나 아래 사항은 반드시 따라야 함

main 함수 템플릿 & txt2Mat() 의 사용

- txt2Mat의 경우 구조체 형식이 다르면 구동 안됨
이 경우 우측의 빨간 밑줄 변경을 통해
본인이 원하는 구조체 형식으로 변경 허용

```
// Create a matrix from a text file
Matrix txt2Mat(string _filePath, string _fileName)
{
    ifstream file;
    string temp_string, objFile = _filePath + _fileName + ".txt";
    int temp_int = 0, nRows = 0;

    file.open(objFile);
    if (!file.is_open()) {
        printf("\n*****");
        printf("\n Could not access file: 'txt2Mat' function");
        printf("\n*****\n");
        return createMat(0, 0);
    }

    while (getline(file, temp_string, '\t'))
        temp_int++;
    file.close();

    file.open(objFile);
    while (getline(file, temp_string, '\n'))
        nRows++;
    file.close();

    int nCols = (temp_int - 1) / nRows + 1;
    Matrix Output = createMat(nRows, nCols);

    file.open(objFile);
    for (int i = 0; i < nRows; i++)
        for (int j = 0; j < nCols; j++) {
            file >> temp_string;
            Output.at[i][j] = stof(temp_string);
        }
    file.close();

    return Output;
}
```

수행 기능 그룹별로 헤더파일 분류(권장)

myMatrix.h :

- 연산자가 요구되지 않는, 행렬 자체 기능 수행에 관한 헤더파일
- 예시) 행렬생성, 행렬출력, 행렬복사, 행렬원소 복사, 원소값 일괄 초기화, 행렬 메모리 할당 해제, 영행렬/단위행렬/전치행렬 생성 등

myNM.h :

- 연산자가 요구되는 행렬연산 및 수치해석 알고리즘 수행 헤더파일
- 행렬 활용이 요구되므로 **#include "myMatrix.h"** 명령 필요
- 예시) 행렬합, 행렬곱, Gauss elimination, LU decomposition 등