

COSE213-01 Assignment #2

2018320192 컴퓨터학과 손재민

1. 구현 환경

- OS – Windows 10
- TOOL – Microsoft Visual Studio 2017

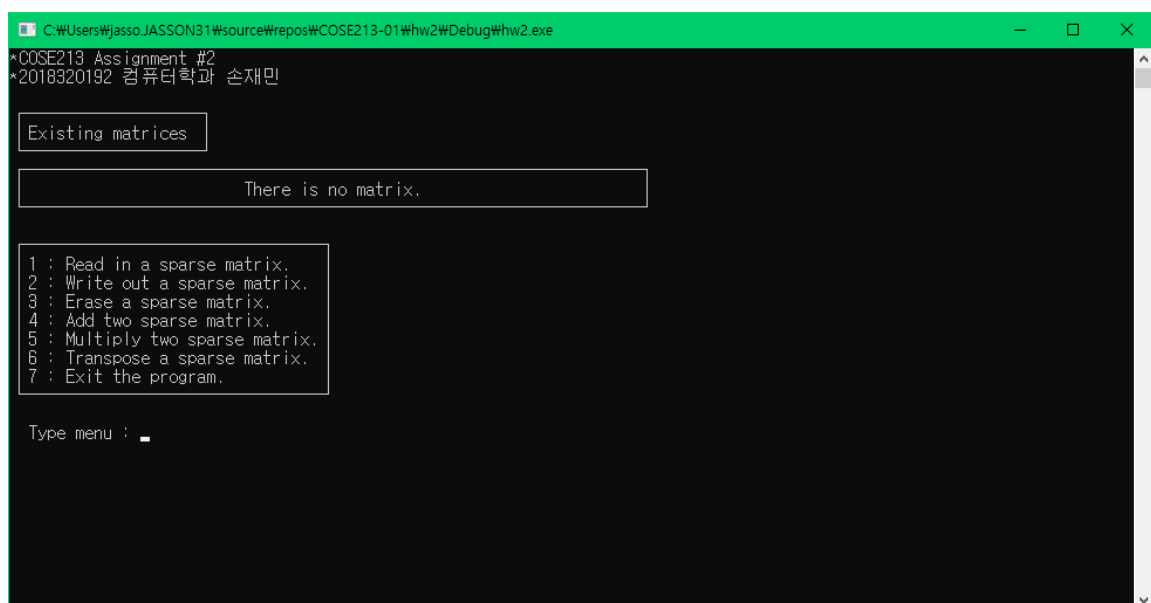
2. 프로그램 사용 방법 및 실행 화면

- 개요 : 본 프로그램은 희소 행렬과 관련된 몇가지 간단한 연산을 수행하는 프로그램입니다. 희소 행렬의 저장, 출력 및 삭제가 가능하고, 행렬 간의 합, 곱 및 전치행렬을 구하는 기능을 갖추고 있습니다. 최대 저장가능한 희소 행렬의 수는 50개로, 0번부터 49번까지 저장이 가능하고, 최대 수를 넘어서면 더 이상 저장을 하지 않습니다.

*프로그램의 설명에서 예시로 입력할 행렬들은 다음과 같습니다.

	1번 : 2/4/4	2번 : 2/4/4	3번 : 4/2/3
0이 아닌 값의 정보 (행/열/값)	0/0/1	0/1/2	0/0/2
	1/0/3	0/2/3	2/0/1
	1/2/8	1/0/1	2/1/2
	1/3/2	1/3/7	

- 메인 메뉴



```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe
*COSE213 Assignment #2
*2018320192 컴퓨터학과 손재민

Existing matrices

There is no matrix.

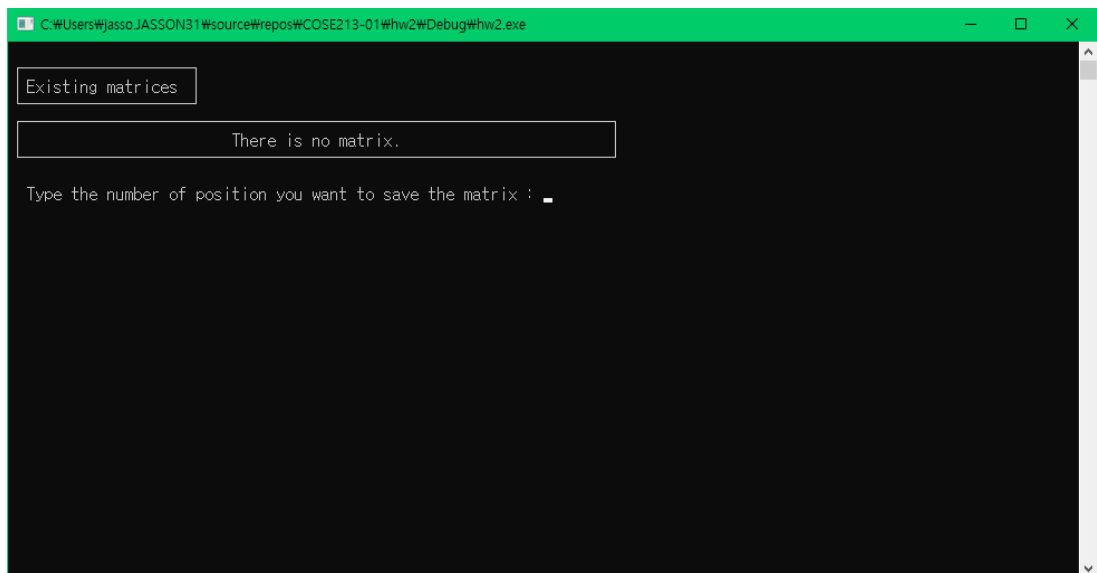
1 : Read in a sparse matrix.
2 : Write out a sparse matrix.
3 : Erase a sparse matrix.
4 : Add two sparse matrix.
5 : Multiply two sparse matrix.
6 : Transpose a sparse matrix.
7 : Exit the program.

Type menu : _
```

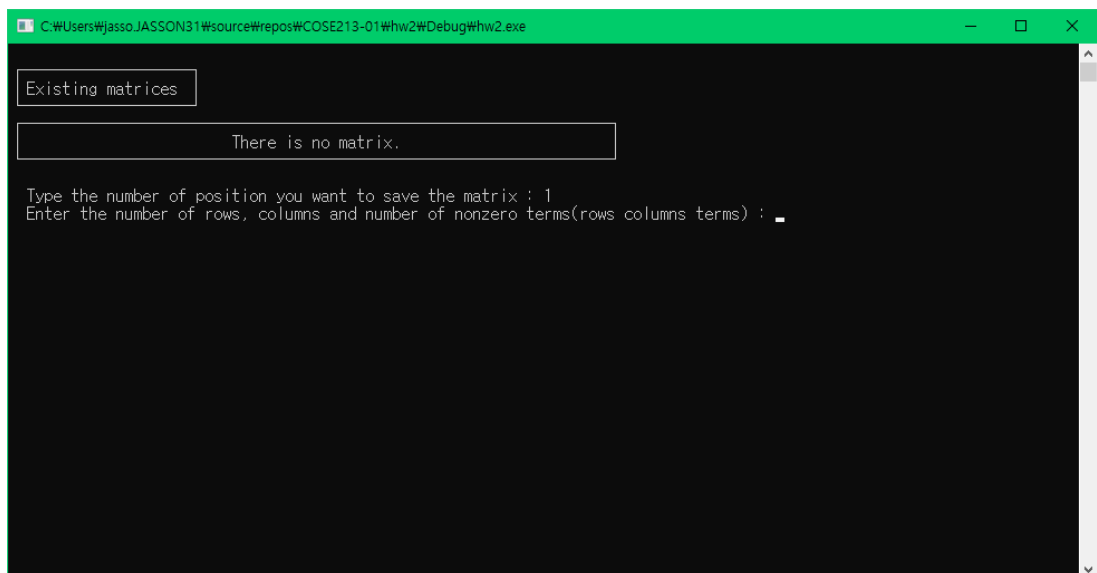
메인 메뉴는 다음과 같이 구성되어 있습니다. 우선 현재 저장된 모든 행렬들의 번호와, 열,

행, 그리고 0이 아닌 값의 개수들을 출력한 다음, 선택 가능한 메뉴들을 출력하고 메뉴를 입력 받습니다. 이후, 입력 받은 메뉴에 해당하는 기능을 실행합니다. 위 사진은 현재 저장된 행렬이 없어 아무 행렬도 출력되지 않은 모습입니다.

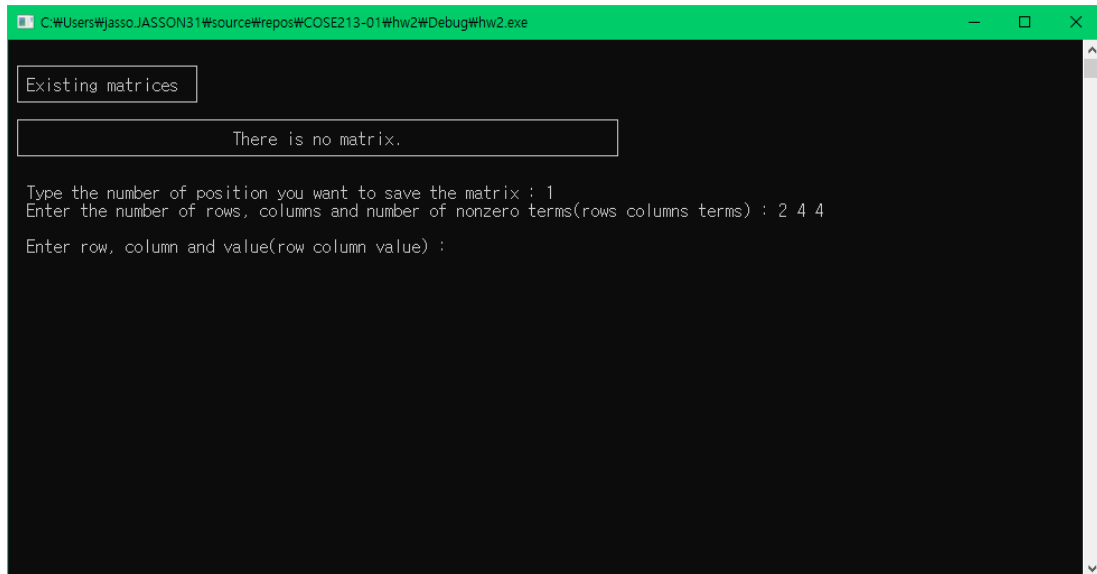
- 희소 행렬 입력 : 메인 메뉴에서 1번을 입력하여 실행합니다.



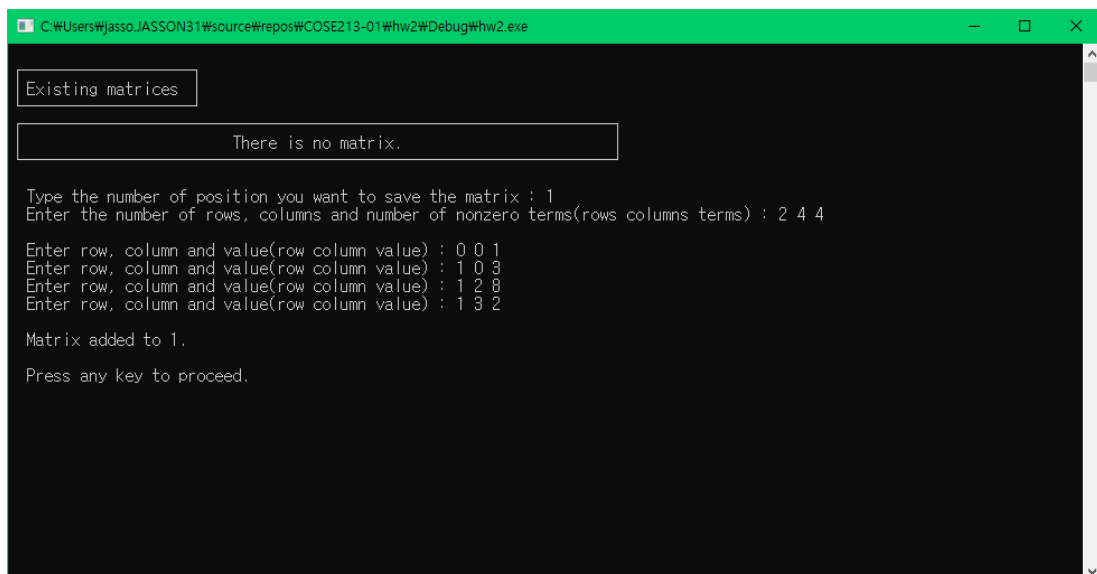
실행 시 희소 행렬을 저장할 번호를 입력 받습니다. 0에서 49 사이의 정수를 입력 받으며, 그 밖의 수나 이미 행렬이 저장되어 있는 번호를 입력 받으면 경고 메시지를 출력하고 다시 메인 메뉴로 되돌아갑니다.



이 후 순서대로 희소 행렬의 행 개수, 열 개수, 0이 아닌 값들의 개수를 입력 받습니다. 행 개수와 열 개수는 최대 50을 넘을 수 없으며, 50이 넘는 수를 입력하면 경고 메시지를 출력하고 다시 메인 메뉴로 되돌아갑니다.



그 다음은 각 값들을 순서에 맞게 입력합니다. 주의할 점으로 각 값들을 입력할 때 반드시 우선 행 내에서 작은 열에서 큰 열 순서대로 입력해주고, 행들도 작은 행에서 큰 행 순서대로 값을 입력해야 하며, 만약 이 순서가 제대로 지켜지지 않을 경우 프로그램이 비정상적으로 작동하게 됩니다.



예시 행렬 중 1번 행렬을 입력한 모습입니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe
*COSE213 Assignment #2
*2018320192 컴퓨터학과 손재민

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

1 : Read in a sparse matrix.
2 : Write out a sparse matrix.
3 : Erase a sparse matrix.
4 : Add two sparse matrix.
5 : Multiply two sparse matrix.
6 : Transpose a sparse matrix.
7 : Exit the program.

Type menu :
```

1, 2, 3번 행렬 모두 입력하고 난 후 메인 메뉴로 되돌아오자, 현재 존재하는 행렬들이 간단하게 출력되는 모습을 볼 수 있습니다.

- 희소 행렬 출력 : 메인 메뉴에서 2번을 입력하여 실행합니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to print :
```

실행 시 출력하고자 하는 행렬의 번호를 입력 받습니다. 현재 저장되어 있지 않은 행렬의 번호를 입력 받을 경우 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to print : 1

numRows = 2, numCols = 4
The matrix by row, column, and value :

0 0 1
1 0 3
1 2 8
1 3 2

Press any key to proceed.
```

행렬 번호를 입력하고 나면 차례대로 행렬의 행 개수, 열 개수를 출력한 뒤 0이 아닌 값들을 각각 행, 열, 그리고 값의 순서로 하나씩 출력합니다.

- 희소 행렬 삭제 : 메인 메뉴에서 3번을 입력하여 실행합니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to erase :
```

실행 시 출력하고자 하는 행렬의 번호를 입력 받습니다. 현재 저장되어 있지 않은 행렬의 번호를 입력 받을 경우 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to erase : 1
Matrix erased.
Press any key to proceed.
```

이 후 입력된 번호의 행렬을 삭제하고, 행렬의 메모리를 모두 되돌립니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe
*COSE213 Assignment #2
*2018320192 컴퓨터학과 손재민

Existing matrices

At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

1 : Read in a sparse matrix.
2 : Write out a sparse matrix.
3 : Erase a sparse matrix.
4 : Add two sparse matrix.
5 : Multiply two sparse matrix.
6 : Transpose a sparse matrix.
7 : Exit the program.

Type menu : _
```

예시로 1번 행렬을 지우자 메인 메뉴에서 보여주는 현재 저장된 행렬 목록에서도 1번 행렬이 삭제되었음을 확인할 수 있습니다.

- 희소 행렬 합 : 메인 메뉴에서 4번을 입력하여 실행합니다.

```

C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe
Existing matrices
At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

Type the number of two matrices you want to add : 1 2
Type the number of position you want to save the result matrix : 4
  
```

실행 시 우선 더하고자 하는 두 행렬의 번호와 계산 결과 행렬을 저장할 번호를 입력 받습니다. 더하고자 하는 두 행렬의 번호가 현재 행렬이 저장되어 있지 않은 번호이거나 결과를 저장하고자 하는 번호가 현재 행렬이 저장되어 있는 번호라면 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다. 더하고자 하는 두 행렬의 열 개수와 행 개수는 반드시 같아야 하며, 만약 크기가 다른 두 행렬이 입력되었다면 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다.

```

C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe
Existing matrices
At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3

Type the number of two matrices you want to add : 1 2
Type the number of position you want to save the result matrix : 4

Result matrix added to 4.
Press any key to proceed.
  
```

예시로 1번과 2번 행렬을 더해 4번에 결과를 저장한 모습입니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6

Type the number of a matrix you want to print : 4

numRows = 2, numCols = 4
The matrix by row, column, and value :
  0  0  1
  0  1  2
  0  2  3
  1  0  4
  1  2  8
  1  3  9

Press any key to proceed.
```

이후 4번 행렬을 출력해보면 정상적으로 행렬의 합이 진행되었음을 확인할 수 있습니다.

- 희소 행렬 곱 : 메인 메뉴에서 5번을 입력하여 실행합니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6

Type the number of two matrices you want to multiply : 1 3
Type the number of position you want to save the result matrix : 5_
```

실행 시 우선 곱하고자 하는 두 행렬의 번호와 계산 결과 행렬을 저장할 번호를 입력 받습니다. 곱하고자 하는 두 행렬의 번호가 현재 행렬이 저장되어 있지 않은 번호이거나 결과를 저장하고자 하는 번호가 현재 행렬이 저장되어 있는 번호라면 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다. 곱하고자 하는 행렬 중 앞서 입력한 행렬의 열 개수와 나중에 입력한 행렬의 행 개수는 반드시 같아야 하며, 만약 이 조건을 만족하지 않는 두 행렬이 입력되었다면 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다.


```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6

Type the number of two matrices you want to multiply : 1 3
Type the number of position you want to save the result matrix : 5

Result matrix added to 5.

Press any key to proceed. _
```

예시로 1번과 3번 행렬을 곱해 5번에 결과를 저장한 모습입니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6
At position [ 5], Rows : 2, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to print : 5

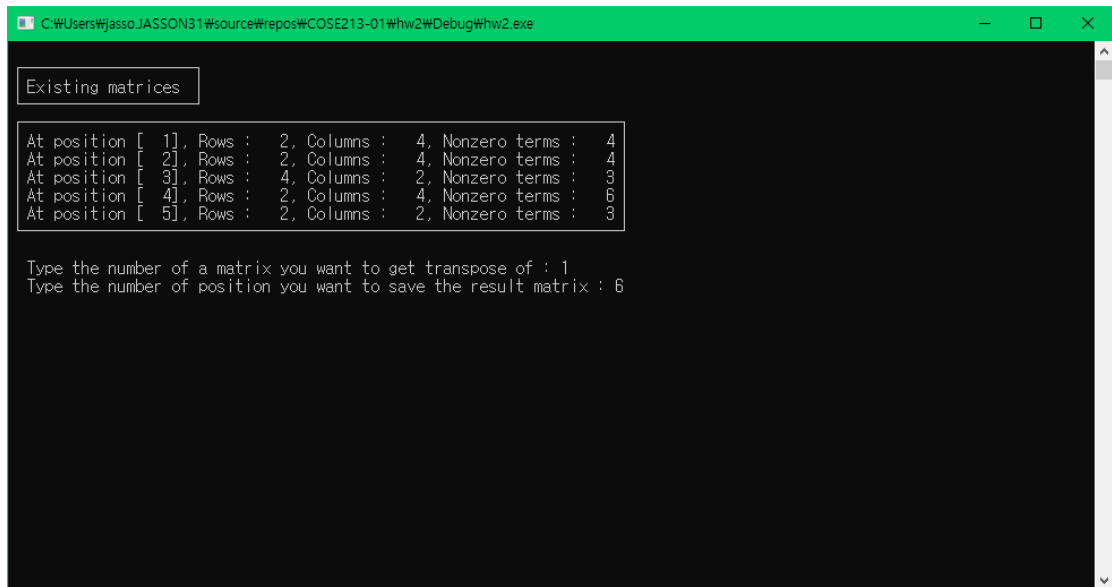
numRows = 2, numCols = 2
The matrix by row, column, and value :

0  0  2
1  0 14
1  1 16

Press any key to proceed. _
```

이후 5번 행렬을 출력해보면 정상적으로 행렬의 곱이 진행되었음을 확인할 수 있습니다.

- 희소 행렬의 전치 행렬 : 메인 메뉴에서 6번을 입력하여 실행합니다.



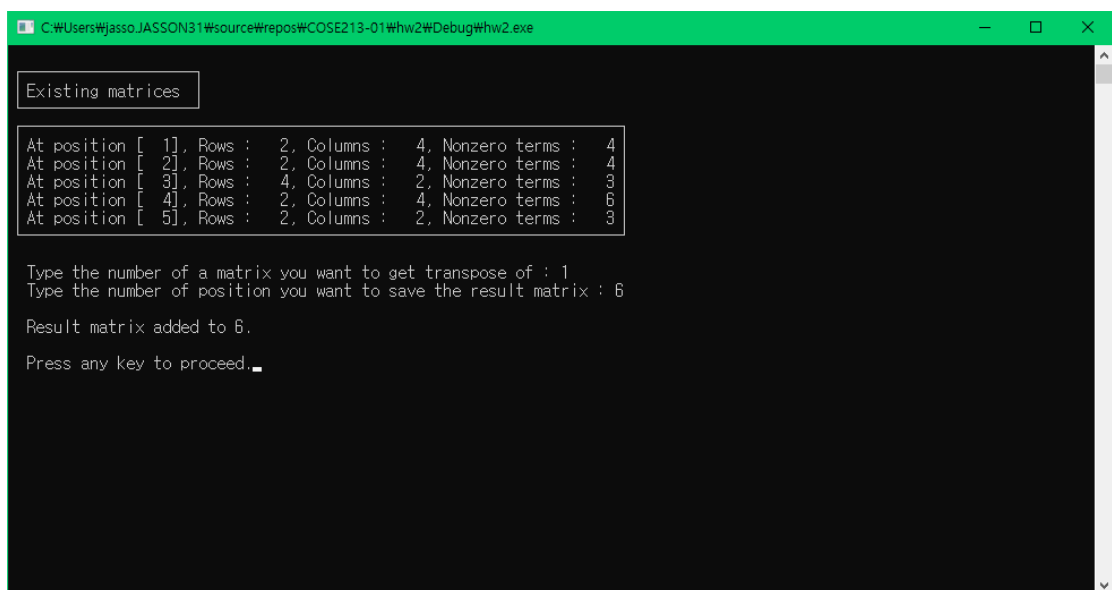
```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6
At position [ 5], Rows : 2, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to get transpose of : 1
Type the number of position you want to save the result matrix : 6
```

실행 시 전치 행렬을 구하고자 하는 행렬의 번호와 계산 결과 행렬을 저장할 번호를 입력 받습니다. 전치 행렬을 구하고자 하는 행렬의 번호가 현재 행렬이 저장되어 있지 않은 번호이거나 결과를 저장하고자 하는 번호가 현재 행렬이 저장되어 있는 번호라면 경고 메시지를 출력하고 메인 메뉴로 되돌아갑니다.



```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6
At position [ 5], Rows : 2, Columns : 2, Nonzero terms : 3

Type the number of a matrix you want to get transpose of : 1
Type the number of position you want to save the result matrix : 6

Result matrix added to 6.
Press any key to proceed.
```

예시로 1번 행렬의 전치 행렬을 구해 6번에 결과를 저장한 모습입니다.

```
C:\Users\jasso.JASSON31\source\repos\COSE213-01\hw2\Debug\hw2.exe

Existing matrices

At position [ 1], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 2], Rows : 2, Columns : 4, Nonzero terms : 4
At position [ 3], Rows : 4, Columns : 2, Nonzero terms : 3
At position [ 4], Rows : 2, Columns : 4, Nonzero terms : 6
At position [ 5], Rows : 2, Columns : 2, Nonzero terms : 3
At position [ 6], Rows : 4, Columns : 2, Nonzero terms : 4

Type the number of a matrix you want to print : 6

numRows = 4, numCols = 2
The matrix by row, column, and value :

0 0 1
0 1 3
2 1 8
3 1 2

Press any key to proceed.
```

이후 6번 행렬을 출력해보면 정상적으로 전치 행렬을 구했음을 확인할 수 있습니다.

- 종료 : 메인 메뉴에서 7번을 입력하여 프로그램을 종료할 수 있습니다.