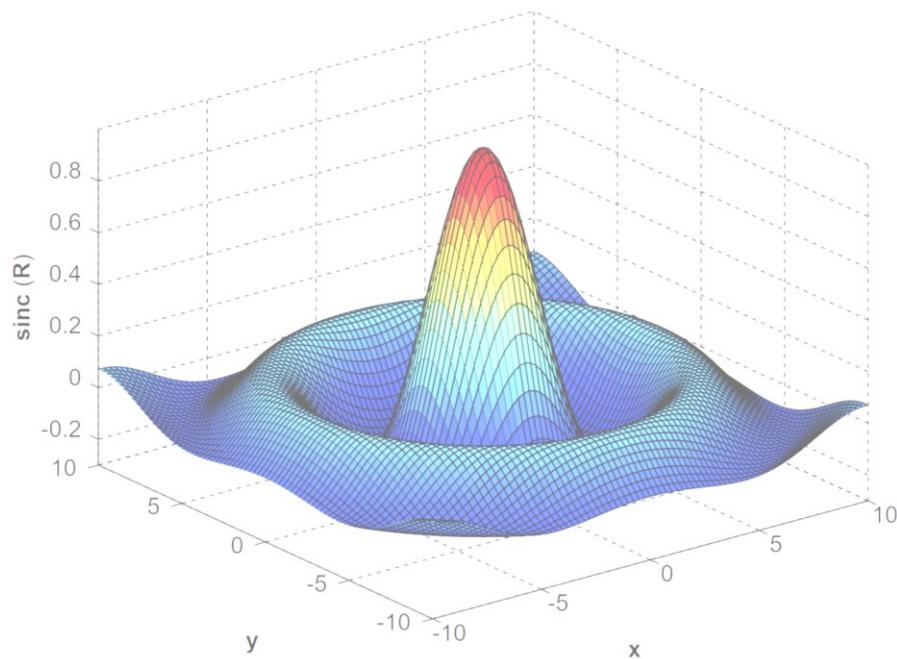


# 응용전산및실습 I

(강의자료 #1)



교과목명 : 응용전산 및 실습 I

담당교수 : 이수형

E-mail : [soohyong@uu.ac.kr](mailto:soohyong@uu.ac.kr)

교재명 : 유인물

# 수업 계획

- 교과목 : 응용전산 및 실습
- 담당교수 : 이 수 형 (010-2521-3227), [soohyong@uu.ac.kr](mailto:soohyong@uu.ac.kr)
- 수업개요
  - MATLAB은 전기, 전자 외 다양한 공학 분야에서 사용되는 소프트웨어로 범용성이 높고 사용하기 편리하여 많은 분야에서 각광받고 있으며 그 정확성 또한 널리 검증되어있는 상태이다. 기본적인 수치해석 문제를 손쉽게 해결할 수 있고 고차원의 수학적 문제 또한 MATLAB을 사용하여 간단히 해를 구할 수 있다. 이와 같이 유용한 수학적 도구를 사용하여 복잡한 수치해석 문제와 전기해석에 관련된 문제를 학생들이 쉽게 해결할 수 있는 능력을 함양하는 데에 본 교과목은 목표를 두고 있다.
- 수업 목표
  - MATLAB의 기본적인 사용법을 숙지한다.
  - MATLAB의 기본 연산인 행렬 연산을 익힌다.
  - MATLAB을 활용하여 그래프 작성하는 방법을 익힌다.
  - 스크립트 파일의 작성법과 프로그램을 작성하는 방법을 익힌다.

# Matlab 시작

- Matlab이란 무엇인가?

- Matlab은 Matworks사의 테크니컬 컴퓨팅 언어로서, 공학계열에서 가장 많이 사용되는 계산용 언어이다. [<https://kr.mathworks.com/>]
- MATLAB = MATrix + LABoratory : Mat이라는 용어가 Matrix에서 나왔으며 기본적인 계산방법이 행렬연산을 손쉽게 할 수 있게 한 일종의 계산용 언어이다.
- 행렬연산을 이용하는 다양한 알고리즘들을 내장하고 있으므로 1줄로 간단하게 계산할 수 있다. (Matlab을 사용하지 않고 이러한 알고리즘을 C/C++, FORTRAN 등의 언어로 구현하려면 많은 노력이 필요하다.)
- M-file이라는 형태로 일반적인 프로그래밍 언어로서의 사용이 가능하다. 즉, C/C++과 같이 프로그램을 작성하여 저장한 것을 불러서 수행할 수 있다.
- 프로그래밍 언어로서 윈도우 환경에서 그래프, 버튼, 메뉴 등의 GUI(Graphics User Interface) 프로그램을 작성하는 것이 가능하다.
- 다양한 툴박스(toolbox)들을 통해서 다양한 분야에서의 응용이 가능하다.
  - 신호처리, 통계학, 영상처리, 제어, 재정, 화학 등
- 심볼로 이루어진 수식을 계산하는 기호 계산이 가능하다.

# Matlab 시작

- Matlab의 장점

- 설치가 쉽다.
- 사용하는 방법이 쉽다. 프로그래밍 언어로서의 문법도 쉬운 편이어서 처음 접하는 사람들도 쉽게 접할 수 있다.
- 인터프리터 방식이므로 명령어 또는 계산식을 한 줄 입력하면 바로 결과를 확인할 수 있기 때문에 사용이 쉽다.
- 다양한 그래프를 지원하므로 계산 결과를 시각화할 수 있는 기능이 우수하다.
- CMEX라는 이름으로 C로 작성된 함수들을 불러와서 사용할 수 있다.
- Matlab으로 작성된 것을 C언어로 변환하는 것이 가능하다. 단, 추가적인 라이브러리 등이 필요하기도 하며 Matlab 버전 및 컴파일러 버전에 따라 내용이 달라지기도 하므로 주의해야 한다.

# Matlab 시작

- Matlab의 단점

- 인터프리터 방식을 사용하므로 계산식 또는 프로그램을 읽으면서 해석한 다음 실행하기 때문에 C/C++과 같은 컴파일하는 언어에 비해서 느리다.
  - 최근 버전에서는 병렬처리 및 해석/컴파일 후 실행하기 때문에 처음 해석할 때를 제외하고는 실행 성능이 많이 향상되었다.
- 상업용 소프트웨어이며 기업용은 매우 비싸다. 기본 패키지도 싼 편은 아니지만 각각의 툴박스들이 따로 구매하도록 하기 때문에 가격이 비싼 편이다.
  - 상업용은 기본 패키지가 260만원이며 필요한 툴박스들은 별도 구매이다.
  - 교육용은 기본적으로 사용하는 툴박스가 몇가지 포함되어 \$55로 판매되고 있다.
- 다른 프로그래밍 언어에 비해서 사용법이 쉽지만, 최근에 많이 사용되는 프로그래밍 언어로서의 방식에 비하면 기초적이다.

# Matlab 시작

- Matlab 의 대안

- Matlab이 공학용 계산 소프트웨어로 고가의 소프트웨어이므로 Matlab의 대안이 될 수 있는 비슷한 기능을 하는 다양한 무료 소프트웨어가 존재한다.
- Matlab을 사용함으로써 사용하고자 하는 여러가지 기능들을 라이브러리 형태로 Python 등의 프로그래밍 언어로서 쉽게 사용할 수 있도록 만들어놓은 형태가 있고, Matlab처럼 별도의 소프트웨어로서 Matlab과 비슷하게 사용해놓은 S/W도 존재한다.
- 프로그래밍 언어 (무료)
  - Python 라이브러리 (SymPy, Numpy) : 파이썬 프로그래밍 언어를 위한 수학, 과학, 공학용 계산 라이브러리 (아주 많은 기능을 제공한다.)
  - Julia 프로그래밍 언어 : Matlab과 비슷하게 행렬연산 및 다양한 과학/공학용 계산을 위한 프로그램 언어이며 속도 또한 매우 빠르다. 단, 아직 초기 개발 버전이다.
- 무료 소프트웨어 패키지
  - Scilab, Octave : 둘 다 Matlab과 유사하게 사용할 수 있다. 벡터/행렬 연산을 기본으로 제공하며 Matlab과 유사하게 사용할 수 있다. 다만 Scilab 보다 Octave가 Matlab과 거의 동일한 명령어 및 계산식을 사용하므로 Octave가 여러가지 대안 중에서 가장 호환성이 뛰어나므로 대부분의 경우 동일하게 사용할 수 있다.

# Octave 시작

- Octave

- GNU재단에서 만든 수치해석용 소프트웨어로 누구나 무료로 사용이 가능하다.
- 기본적으로 Matlab과 동일한 문법을 사용하므로 한 S/W에 익숙하면 다른 곳에 적용이 아주 쉽다.
- Gnuplot 등의 그래프 툴과 연계해서 표, 그래프, 차트 등을 만들 수 있으며 Matlab과 동일한 명령어로 같은 내용의 그래프 등을 만들 수 있다. (다만, 그래프의 자세한 설정 방법은 Matlab과 다소 차이가 난다.)
- Matlab이 사용 소프트웨어로 최적화 등에 많은 투자를 하기 때문에, Octave의 수행 속도는 Matlab에 비해서 느리다.

- Octave 사용 이유

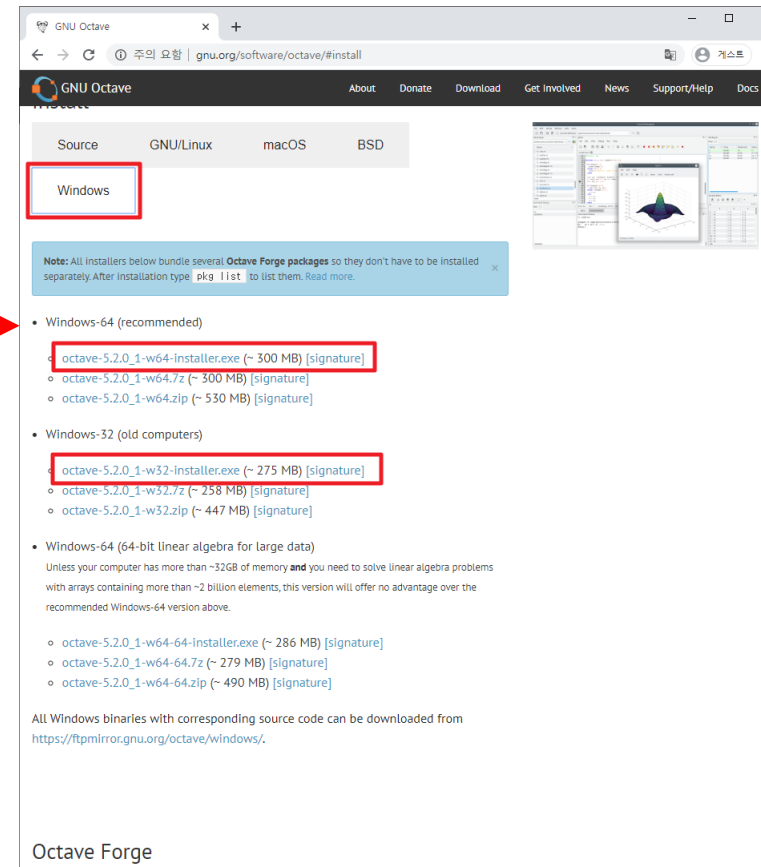
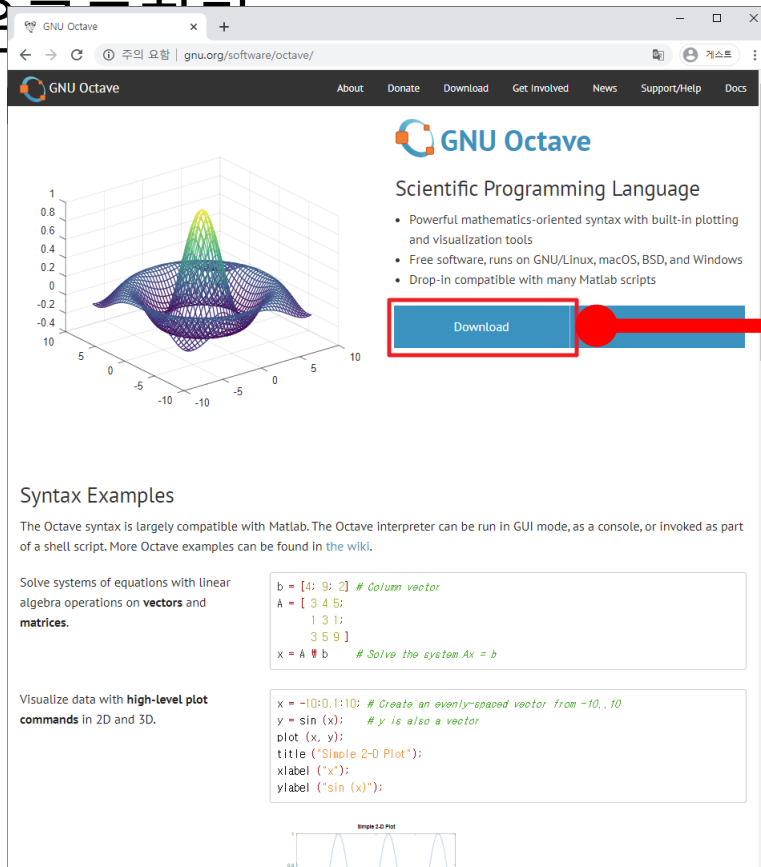
- 상용 소프트웨어이므로 구매해서 설치해 놓은 학교의 실험실에서는 사용이 가능하나, 연습/복습 등의 이유로 집에서 따로 구매해서 사용하여야 한다.
- 따라서 수강하는 학생들이 별도로 실습을 하기 위해서 사용할 수 있으므로 Octave를 설치하여 사용한다. (추후 실습 시간에 Matlab과 Octave를 비교하면서 실습할 예정)

# Octave 설치

- Octave 설치 방법

- 홈페이지 [<http://www.gnu.org/software/octave/>]에 접속한다.

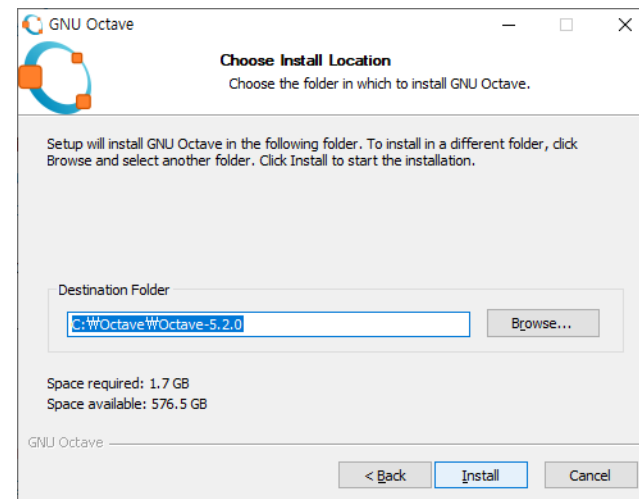
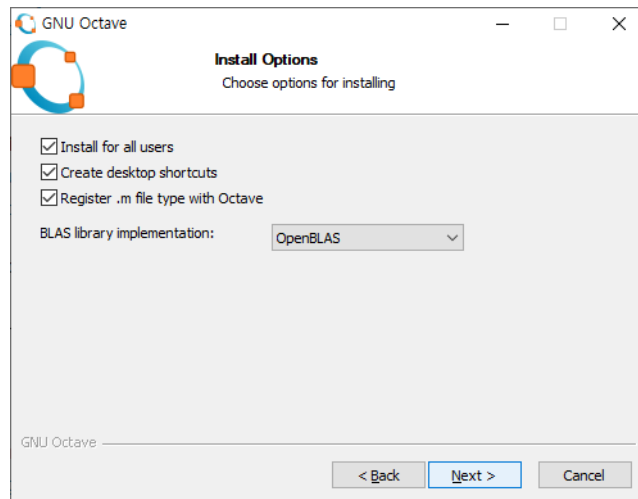
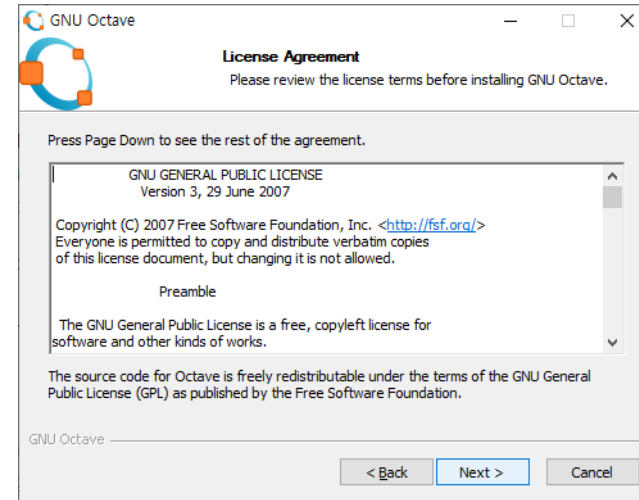
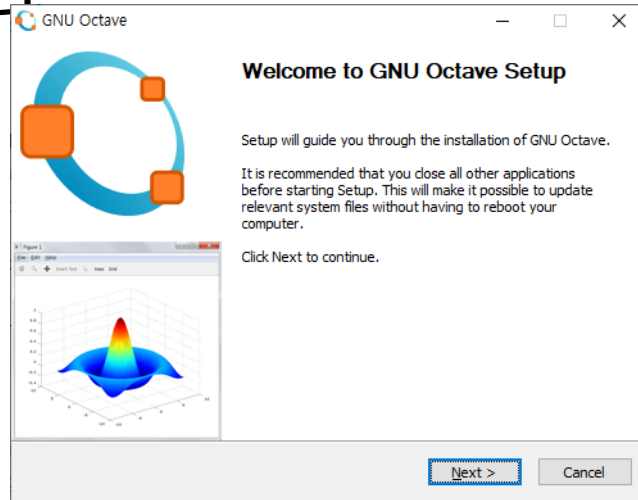
- 다운로드를 클릭하고, Windows를 선택한 후에 윈도우 버전에 알맞은 설치파일을 다운로드한다.





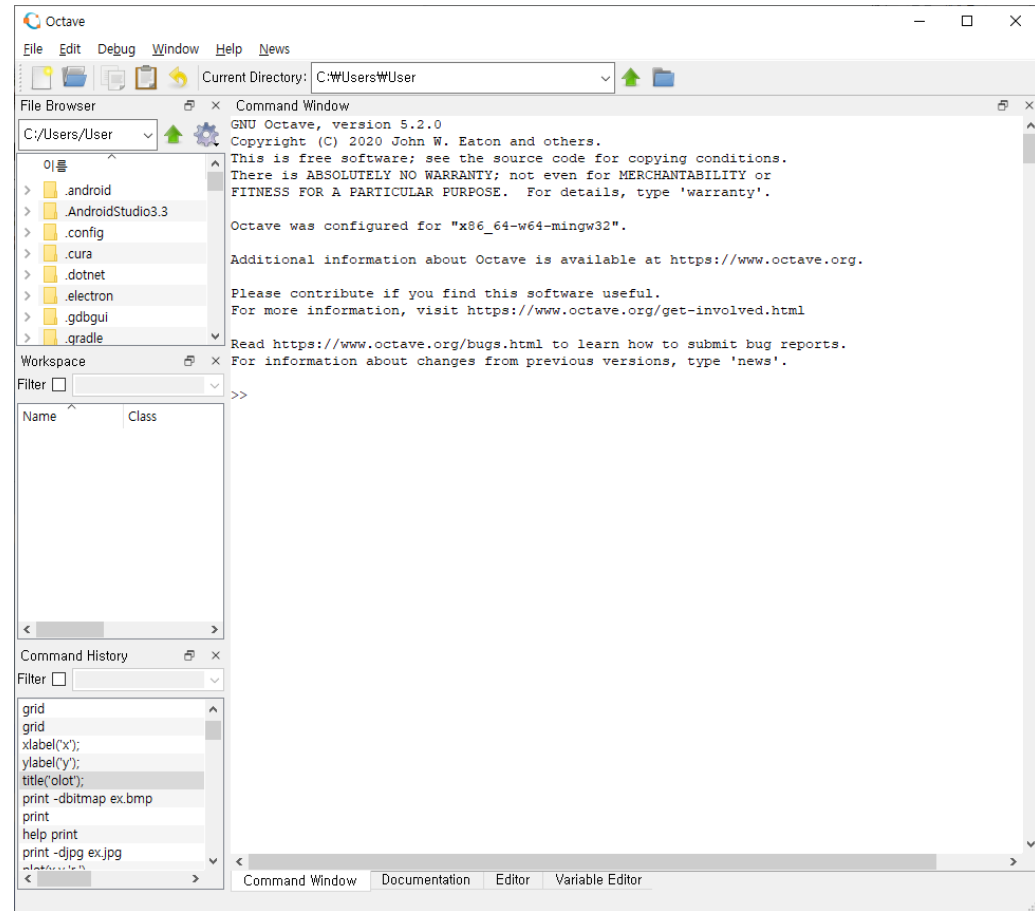
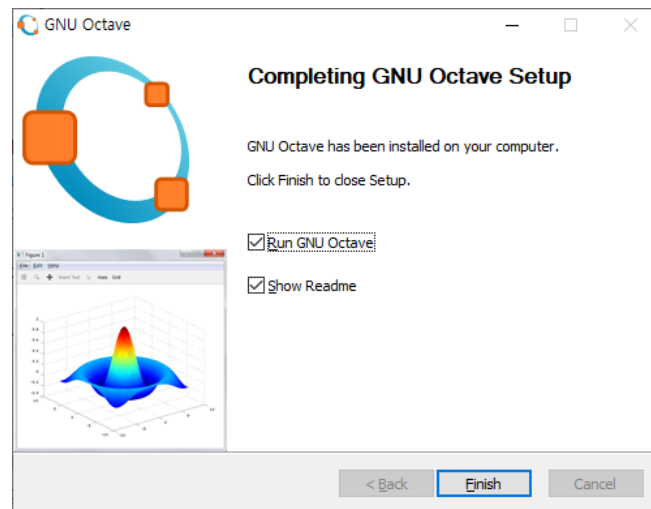
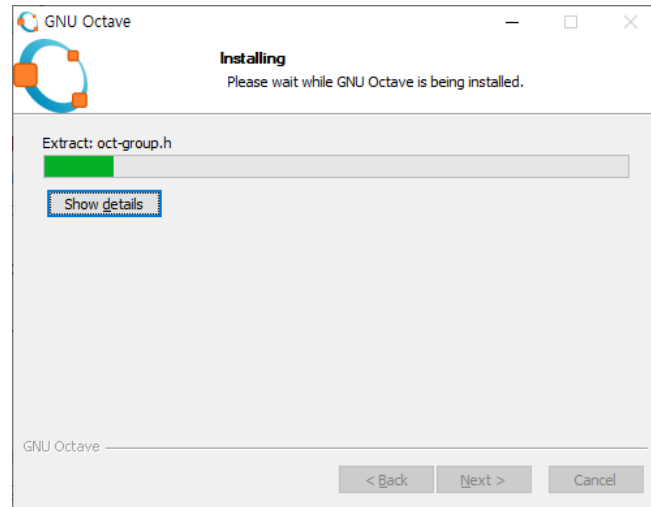
# Octave 설치

- 설치 파일을 수행한 후에 기본으로 선택되어진대로 [Next]를 눌러서 설치한다

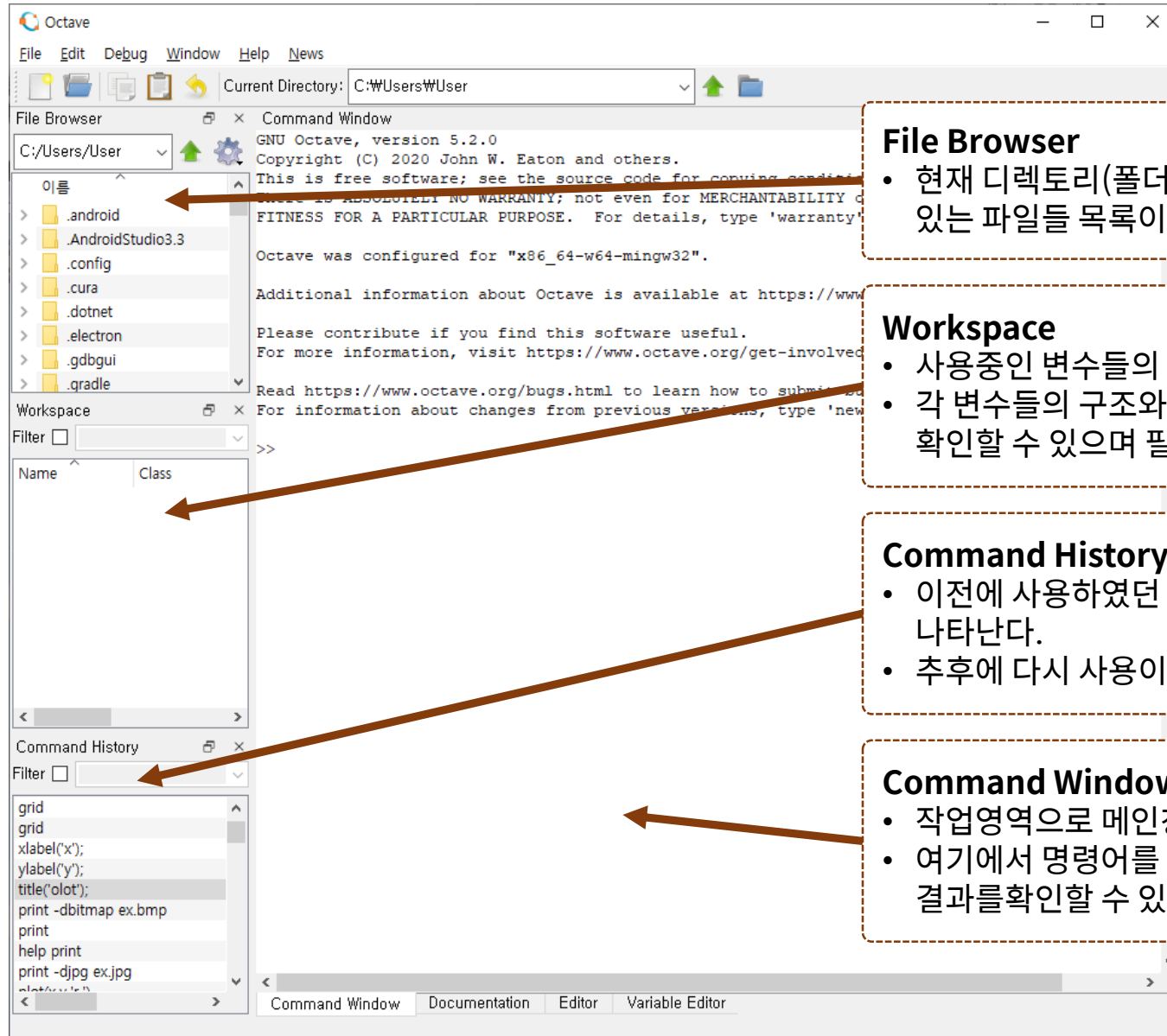


# Octave 설치

- 설치 후 수행하면 다음과 같은 화면이 나타난다.



# Octave 화면



## File Browser

- 현재 디렉토리(폴더)와 여기에 포함되어 있는 파일들 목록이 나타난다.

## Workspace

- 사용중인 변수들의 목록이 나타난다.
- 각 변수들의 구조와 값을 포함한 정보들을 확인할 수 있으며 필터를 사용할 수 있다.

## Command History

- 이전에 사용하였던 명령어/계산식들이 나타난다.
- 추후에 다시 사용이 가능하다.

## Command Window

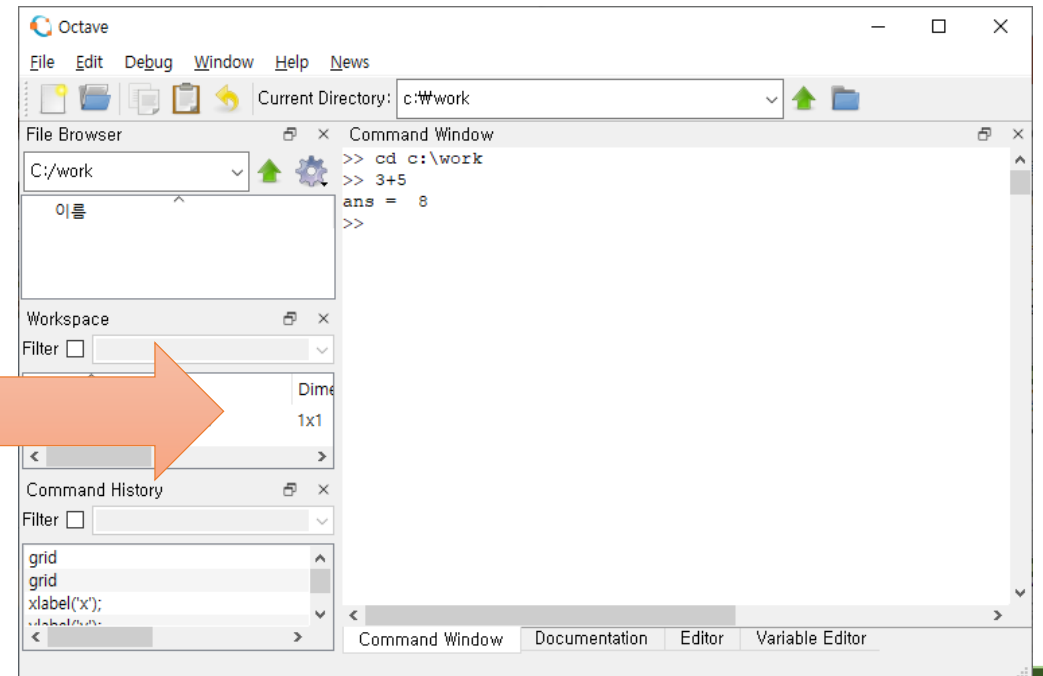
- 작업영역으로 메인창에 해당한다.
- 여기에서 명령어를 입력하고 수행 결과를 확인할 수 있다.

# Octave 시작

- 실습시에 작업영역은 C:\work로 지정한다.
  - File Browser에서도 이에 맞추어서 사용한다.
  - Matlab 또는 Octave의 명령어에서 cd 명령어를 사용하면 디렉토리 위치를 변경할 수 있다.
    - `cd c:\work` → c:\work 디렉토리(폴더)로 현재의 폴더를 변경한다.

- 명령창(Command Window)에서 명령어 입력

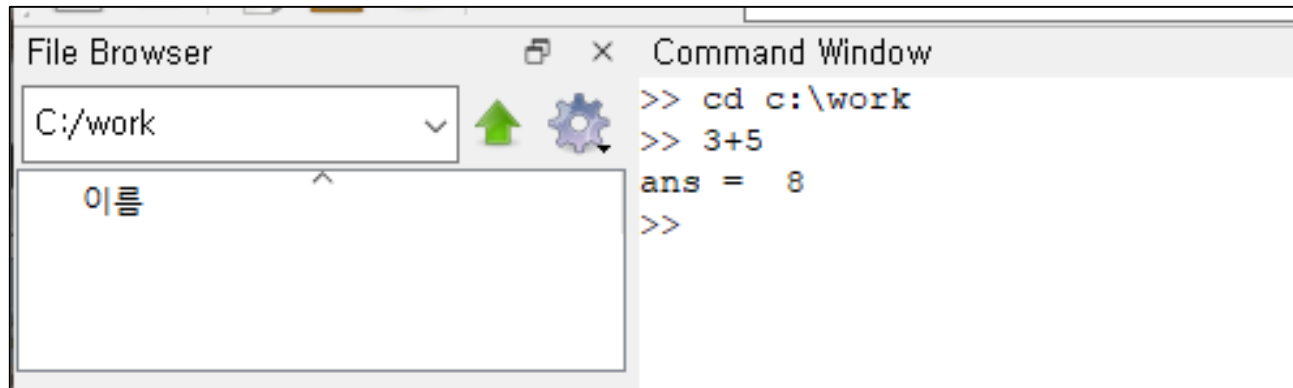
```
>> clc
>> cd c:\work
>> 3 + 5
ans = 8
>>
```



# Octave 시작

- Octave 명령창의 사용 방법

- [>>] 기호가 명령어를 입력할 수 있는 대기상태라는 것을 알린다.
- [>>] 기호가 나타나면 앞의 명령어를 입력한다.
- `clc` : `clear screen`으로 화면을 지우는 명령어이다.
  - 수행 후에는 그림과 같이 화면이 지워지며 정리된다.
- `3+5` : 수식을 입력하면 수식을 계산한다.
  - 다른 변수에 입력하지 않는 계산 결과는 `ans`라는 변수에 자동으로 저장된다.
  - 즉, `ans`는 가장 최근의 계산결과를 의미한다.



The screenshot shows the Octave interface. On the left is a 'File Browser' window with the current directory set to 'C:/work'. On the right is the 'Command Window' where the following commands and output are visible:

```
>> cd c:\work
>> 3+5
ans = 8
>>
```

# 사칙연산

- Matlab을 사용하면 계산식의 입력이 가능하며, 가장 간단한 계산식으로 사칙 연산을 들 수 있다.

Operation	Symbol	Example
Addition, $a + b$	+	$3 + 22$
Subtraction, $a - b$	-	$90 - 54$
Multiplication, $a \cdot b$	*	$3.14 * 0.85$
Division, $a \div b$	/ or \	$56/8 = 8 \setminus 56$
Exponentiation, $a^b$	^	$2^8$

- 일반적인 프로그래밍 언어와 비슷하게 더하기와 빼기는 수학 기호인 +, - 를 그냥 사용하고, 곱하기와 나누기는 \*, / 로 사용한다.
- 특이하게 Matlab에서는 일반적인 나누기인 / 기호 외에 \ 기호를 사용하기도 하는데 / 기호는 right division 이라고 하고 \ 는 오른쪽의 값을 왼쪽으로 나누는 나눗셈을 수행하며 left division 이라고 한다. 일반적인 숫자의 경우에는 순서만 주의하면 결과는 차이가 없으며, 행렬 연산에서 차이가 발생한다.

# 사칙연산

- 다음의 연산을 따라해 보도록 하자.
  - 먼저 +, -, \*, / 는 C 언어에서 사용하던 연산자이므로 별 무리없이 이해가 될 것이다.
  - 왼쪽 나눗셈(left division)의 경우에  $56 / 8$ 과  $8 \setminus 56$ 의 결과가 동일한 것을 확인하도록 한다.
  - 거듭제곱(exponentiation)인  $^$  연산은 C 언어에서는 없지만 Matlab에서는 지원하며 FORTRAN 언어의 경우  $**$ 의 연산자로 지원한다.
    - $2^8 \Rightarrow 2^8$
  - 잘 알다시피 괄호() 는 연산의 우선순위를 변경하는 용도로 사용된다. 오른쪽 아래 두 가지 경우를 비교해 보도록 하자.
- 수행결과는 화면을 캡처해서 과제물에 포함하여 제출한다.

```
>> 3 + 22
ans = 25

>> 90 - 54
ans = 36

>> 3.14 * 0.85
ans = 2.6690

>> 56 / 8
ans = 7

>> 8 \ 56
ans = 7

>> 2 ^ 8
ans = 256

>> ((1.043 + 2.01) * 3)^2
ans = 83.887

>> (1.043 + 2.01) * 3^2
ans = 27.477
```

# 사칙연산

- 계산식이 주어질 때 연산자들은 다음의 우선순위를 가지고 계산된다.

우선순위	연산자
1	괄호, 중첩되어 있는 경우에는 안쪽 괄호부터 수행된다.
2	거듭제곱 (^)
3	곱셈, 나눗셈
4	덧셈, 뺄셈

- 동일한 우선순위의 연산자들은 동등한 순위를 가진다. 즉 곱셈과 나눗셈 기호는 서로 동등한 순위이다. 따라서 먼저 나오는 연산자를 먼저 수행한다.
- 위와 같이 복잡한 계산식의 경우에도 식만 입력하면 바로 계산되므로 탁상용 계산기의 용도로 사용해도 무방하다.



# 변수

- 변수(variable)란?

- 변수는 하나의 이름을 가지는 기억장소로써 수치를 기억할 수 있으며, 이를 변경할 수 있다. 기억된 변수는 그 자체로 기억된 숫자로서 수식, 함수 등에 바로 사용할 수 있다.
- Matlab에서는 숫자 하나(스칼라: scalar)를 기억할 수도 있고, 벡터 또는 배열을 저장할 수도 있다.

- 변수 이름 규칙

- 대소문자를 구별한다.
  - Cost, cost, CoST, COST 는 서로 다른 변수의 이름이다.
- 최대 31글자까지 허용한다. 단, 최근 Matlab과 Octave에서는 63글자까지 허용한다.
  - 참고로 namelengthname() 함수를 수행하면 최대 가능한 길이를 알 수 있다.
- 변수 이름은 문자, 숫자, 밑줄(\_)이 사용된 하나의 단어로 구성되며, 문자로 시작해야 한다. 다른 기호는 사용할 수 없다.
  - Octave에서는 '\_' 로 시작할 수도 있다. (Matlab에서는 안된다.)
- 키워드는 변수로 사용할 수 없으며, 함수 이름은 변수로 사용하지 않도록 주의한다.
  - 함수 이름을 변수로 사용해도 되나, 사용한 후에 해당하는 함수를 사용하지 못한다.

```
>> namelengthmax()  
ans = 63  
>>
```

# 변수

- 변수의 사용법

- 변수에 값을 지정하고자 할 때는 = 기호를 사용한다.

- 변수 a를 만들고 5를 대입한다.
    - 변수 b를 만들고 4를 대입한다.

```
>> a = 5
a =
    5
>> b = 4
b =
    4
```

- 수식에 변수를 적으면 변수에 저장되어 있는 값이 사용된다.

- 기억되어 있는 a, b 값이 계산에 사용되어  
a + b 는 5 + 4로 계산된다.

```
>> a + b
ans =
    9
>> a * b
ans =
   20
```

# 변수

- 미리정의된 특별한 변수들

- Matlab에서는 시작할 때 미리 정의되어 있는 특별한 변수들이 있다. 이 변수들은 값의 변경이 가능하나, 변경 후에는 원래의 의미가 사라지므로 가능하면 변경하지 않도록 한다.

Special Variable	Description
ans	결과를 나타내는 기본 변수, 가장 최근에 계산된 결과를 저장하고 있음
pi	원주율 ( $\pi$ )
eps	인접한 두 수 사이의 최소값으로 이 숫자 이하로 가까이 있는 숫자들은 구분하지 못함 : $2.2204 \times 10^{-16}$
inf	무한대 ( $1 / 0$ )
i, j	허수를 표현하는데 사용되는 상수, $i = j = \sqrt{-1}$
realmin	저장할 수 있는 최소의 + 숫자 : $2.2251 \times 10^{-308}$
realmax	저장할 수 있는 최대의 + 숫자 : $1.7977 \times 10^{308}$

```
>> pi
ans =
    3.1416
>> eps
ans =
    2.2204e-16
>> inf
ans =
    Inf
>> i
ans =
    0.0000 + 1.0000i
>> j
ans =
    0.0000 + 1.0000i
>> realmin
ans =
    2.2251e-308
>> realmax
ans =
    1.7977e+308
```

# 변수

- 미리 정의된 키워드

- 키워드(keyword)란 Matlab에서 특정한 명령어로 사용되므로 변수의 이름으로 사용될 수 없다.
- 후에 프로그램을 작성하는 경우 제어문으로 사용되는 명령어가 대부분이다.
- Matlab과 Octave의 경우 미리 정의된 키워드가 조금 다르나, 제어문과 같은 것들은 공통이므로 아래의 키워드를 보면서 구분해보자.
- Matlab keywords
  - break, case, catch, classdef, continue, do, else, elseif, end, for, function, global, if, otherwise, parfor, persistent, return, spmd, switch
- Octave keywords
  - \_\_FILE\_\_, \_\_LINE\_\_, break, case, catch, classdef, continue, do, else, elseif, end, end\_try\_catch, end\_unwind\_protect, endclassdef, endenumeration, endevents, endfor, endfunction, endif, endmethods, endparfor, endproperties, endswitch, endwhile, enumeration, events, for, function, global, if, methods, otherwise, parfor, persistent, properties, return, switch, try, until, unwind\_protect, unwind\_protect\_cleanup, while
  - 특히 end로 시작되는 키워드들은 Matlab에서 사용하는 것처럼 end로 대신할 수 있다.

# 자주 사용하는 명령어

- 앞에서 본 화면을 지우는 `clc` 이외에도 자주 사용하는 명령어들이 있으므로, 반복 연습하여 익히도록 한다.
  - `help` : 함수 이름이나 명령어의 사용법을 익히는 명령어이다.
  - `who` : 현재 사용중인 변수들의 목록을 출력한다.
  - `whos` : 변수들의 목록 및 크기, 데이터 형 등의 정보를 같이 출력한다.
  - `clear` : 사용중인 변수를 지운다. 변수명을 지정하지 않으면 모든 변수를 제거한다.

```
>> help cos
cos    Cosine of argument in radians.
      cos(X) is the cosine of the elements of X.

      See also acos, cosd.
      .... (생략)

>> a = 5
A =
     5
>> b = 6
B =
     6
>> who

사용자의 변수:
A  b
>> whos
      Name      Size      Bytes  Class  Attributes

      a         1x1         8  double
      b         1x1         8  double

>> clear a
>> a
'a' 은(는) 정의되지 않은 함수 또는 변수입니다.
>> b
B =
     6
>> clear
```

# 자주 사용하는 명령어

- 아래의 명령어 중에서 pwd, cd 등은 자주 사용하므로 반드시 익히도록 한다.
  - pwd (print working directory)
    - 현재 디렉토리(폴더)를 보여준다.
  - cd (change directory)
    - 현재 디렉토리를 변경한다.
  - dir : 현재 디렉토리의 파일들을 보여준다.
  - quit : Matlab를 종료한다.
  - version : 현재 버전을 출력한다.

```
>> pwd
ans = c:\users\user
>> cd c:\work
>> pwd
ans = c:\work
>> dir
.   ..
>> version
ans = 5.2.0
```

[ Octave 수행 결과]

```
>> pwd
ans =
C:\WINDOWS\system32
>> cd c:\work
>> pwd
ans =
c:\work
>> dir
.   ..
>> version
ans =
8.6.0.267246 (R2015b)
>>
```

[ Matlab 수행 결과]

# 과제물

- 아래의 과제물을 풀고 과제물 사이트를 통해서 제출하라.

- 과제물 사이트 : <http://energy.uu.ac.kr/report>

- 과제물

1. p. 15의 점선으로 둘러싸인 내용을 직접 수행하고 그 결과 화면을 캡처하여 제출하라.

2. 다음의 식을 계산하고 그 결과 화면을 캡처하여 제출하라.

$$\frac{22+51}{30-2.2^3}$$

3. 변수  $x = 1.7$ 로 지정하고 다음의 식을 계산하도록 한 후 결과화면을 제출하라.

$$5x^3 - 2.3x^2 + 3x - 10$$