

디지털신호처리



강 의 노트

그래프와 흐름 제어

학습내용

- ❖ 기본적인 그래프 명령어
- ❖ 흐름 제어 명령어
- ❖ M-파일 프로그래밍

학습목표

- ❖ 기본적인 그래프 함수 명령어로 그래프를 그릴 수 있다.
- ❖ 다양한 흐름 제어 명령어를 학습하고, 흐름 제어 프로그램에 활용할 수 있다.
- ❖ 2가지 모드의 M-파일 프로그래밍을 할 수 있다.

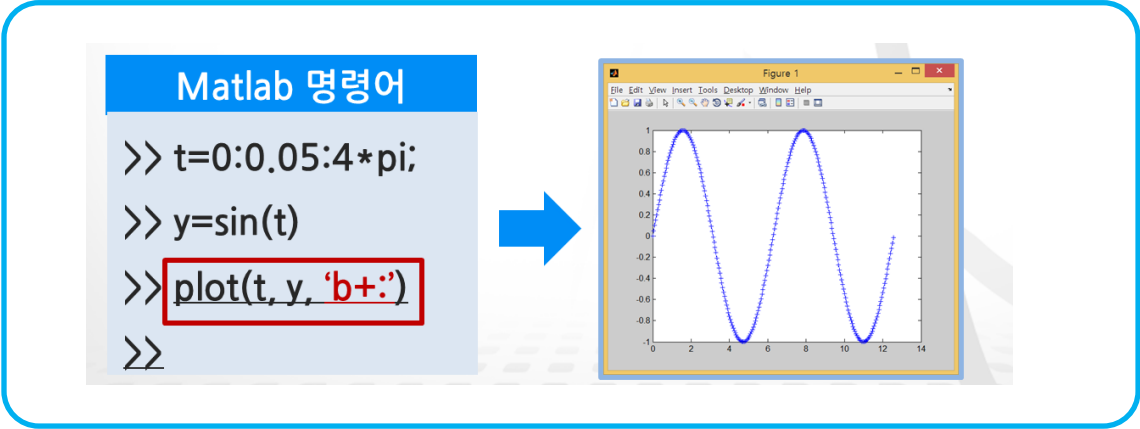


기본적인 그래프 명령어

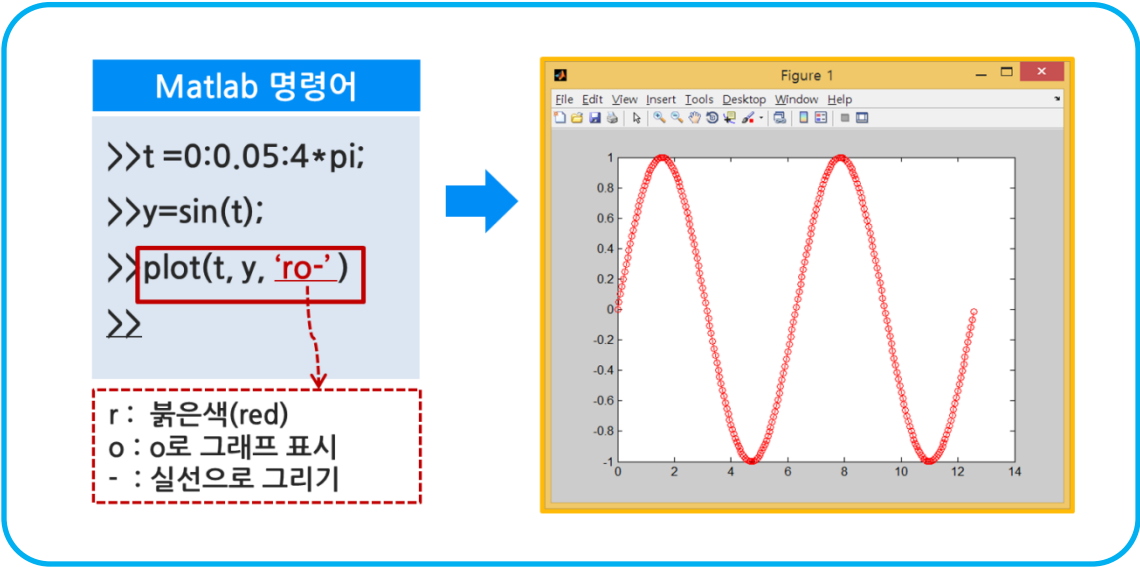
1. 그래프 명령어 plot

1) 정의

- `plot(X,Y)`: 벡터 X에 대한 벡터 Y 값에 대응한 값을 선형 축을 사용하여 그래프로 출력하는 함수
- [예] `plot(y)`, `plot(x,y)`, `plot(x,y,s)`,



2) 2차원 그래프 명령어





기본적인 그래프 명령어

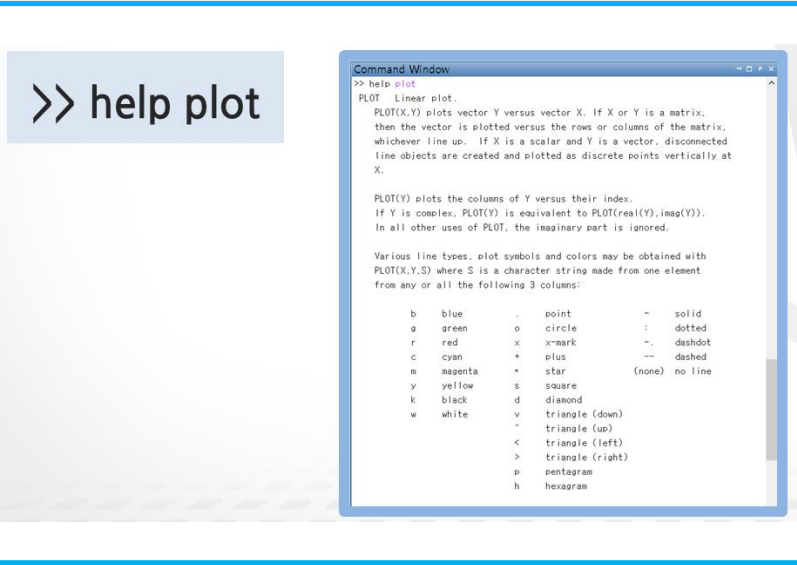
2. 그래프 선 종류/색상

1) 다양한 명령어

기호	선의 종류	기호	색상
.	점	y	노랑색(yellow)
o	원	m	자홍색(magenta)
x	x	c	하늘색(cyan)
+	덧셈	r	빨간색(red)
*	별표	g	녹색(green)
-	실선	b	파란색(blue)
:	점선	w	흰색(white)
-.	일점쇄선	k	검정색(black)
--	쇄선		

2) [참고] Help 명령어 이용 방법

- 명령어 창에서 help 명령어를 입력하면 MATLAB에 관한 도움말을 제공



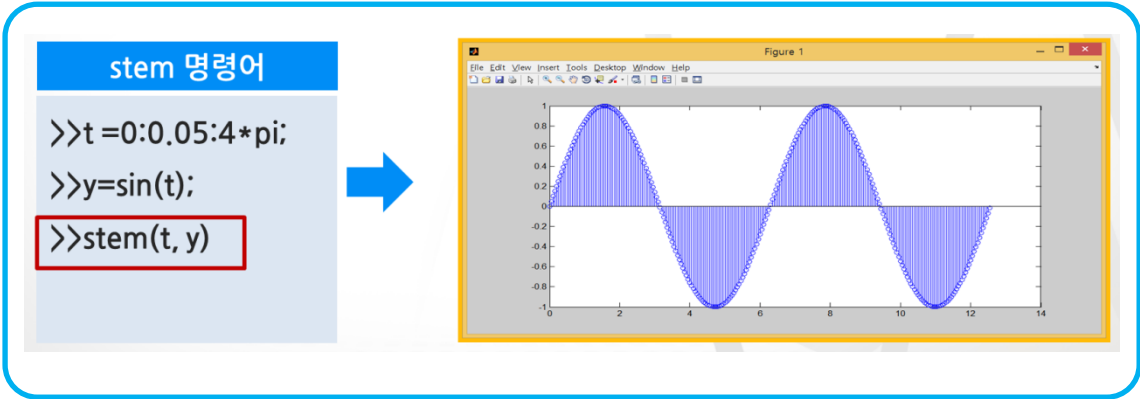


기본적인 그래프 명령어

3. 기타 그래프 명령어

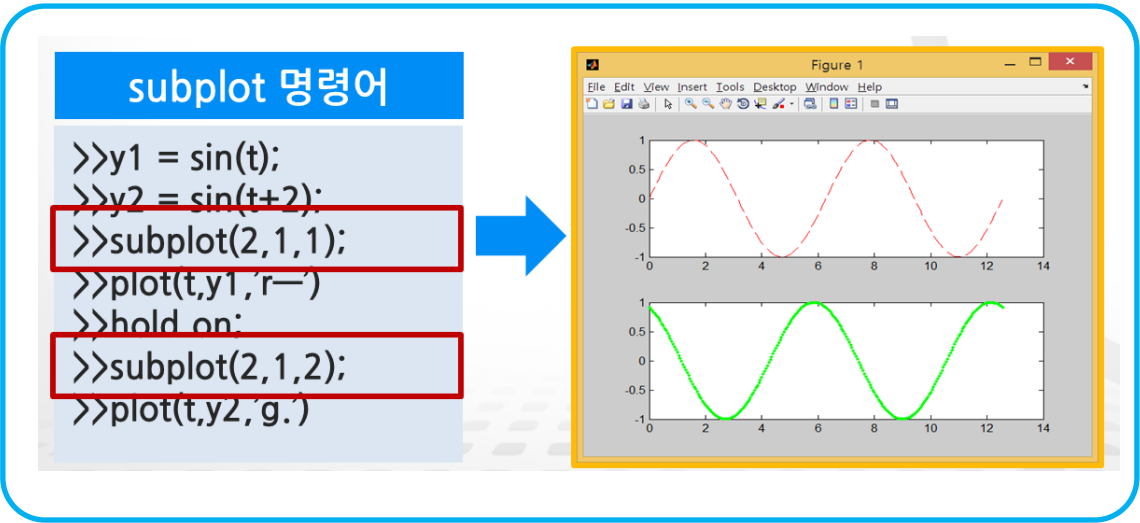
1) stem

- 이산적인 축을 사용하여 이산신호에 대한 그래프를 출력하는 그리기 명령어



2) subplot

- 부그래프를 그릴 수 있도록 하는 명령어
- 그림창 수를 m x n 행렬로 나누고, p번째 위치에 p번째 그림을 그릴 수 있도록 하는 명령어





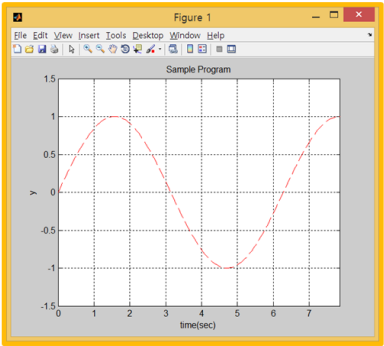
기본적인 그래프 명령어

3) axis, xlabel, ylabel, title, grid

기호	기능
axis	x축과 y축의 크기를 조정
xlabel	x축에 축의 이름을 추가
ylabel	y축에 축의 이름을 추가
grid	격자선을 그래프에 추가
title	그래프의 제목을 추가

기타 명령어

```
>>t=0:0.05:4*pi;  
>>y = sin(t);  
>>plot(t,y,'r--');  
>>axis([0 2.5*pi -1.5 1.5]);  
>>xlabel('time(sec)');  
>>ylabel('y');  
>>title('Sample Program');  
>>grid;
```





흐름 제어(Flow Control) 명령어

1. if, elseif, else, end 명령어

if 논리적인 식

.....

명령어 문장

.....

elseif 논리적인 식

.....

명령어 문장

.....

else 논리적인 식

.....

명령어 문장

.....

end

apples = 6;

if apples < 5
cost = apples*50;

elseif apples > 5 && apples < 10;
cost = apples*30;

else
cost = apples*20;

end

5개 미만이면
1개당 50원

5~10개는
30원

2. for 루프 명령어

- 명령어들이 고정, 미리 결정된 횟수 동안 반복

for 변수 = 시작값:증가분:최종값

.....

명령어 문장

.....

end

for i=1:1:10
x(i)=i;
end

x=

12345678910

3. while 명령어

- 국부적인 조건에 따라 하나 이상의 문장들을 불확정적인 횟수만큼 반복

while 국부적 조건

.....

명령어 문장

.....

end

i=1;
while i<=5
x(i)=i+2;
i=i+1;
end

x=

3 4 5 6 7



흐름 제어(Flow Control) 명령어

4. switch 명령어

- 변수나 표현에 의해서 임의의 실행문이 선택적으로 수행되는 명령어

```
switch expression(scalar or string)
```

```
case value1
```

```
case value2
```

```
otherwise
```

```
end
```



```
switch input_num
```

```
case -1
```

```
disp('negative one');
```

```
case 0
```

```
disp('zero');
```

```
case 1
```

```
disp('positive one');
```

```
otherwise
```

```
disp('other value');
```

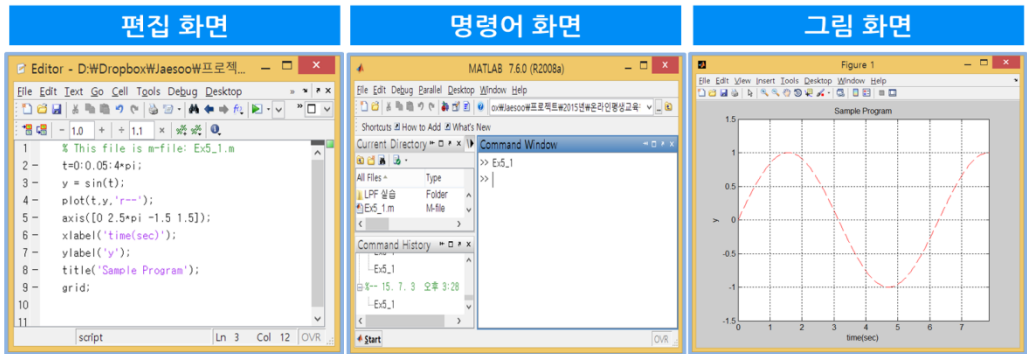
```
end
```




M-파일 프로그래밍

1. Script-mode

- 실행할 Matlab 명령어들을 입력한 전체 명령어들을 모아놓은 M-파일을 이용하는 프로그래밍 방법
- M-파일은 Matlab이 제공하는 Text Editor를 이용하여 작성
- 명령어창에서 M-파일을 부르거나, 다른 M-파일 내에서 M-파일을 호출해 M-파일 명령어들을 실행할 수 있음



2. Function-mode

- 입력매개변수와 출력 매개변수를 다루는 함수 양식으로 프로그래밍하는 것
- M-파일의 함수 이름과 파일의 첫 번째 줄에 있는 함수의 이름은 동일해야 함
- 첫 번째 줄은 **함수 선언줄**이라고 하며, 파일에 있는 마지막 줄이 실행되거나 도중에 return문 실행 시 종료

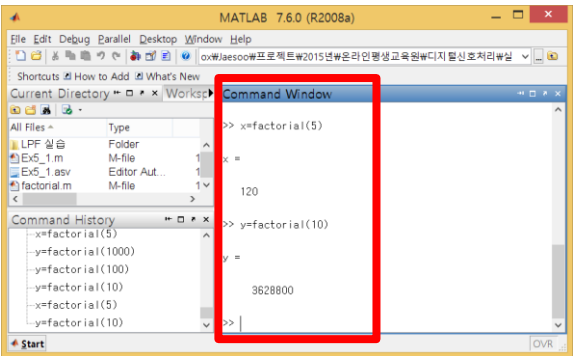
예제 05-01

n!를 구하는 함수를 M-파일로 작성해보자.

[예제풀이]

- Function mode M-파일을 작성하려면, “function 출력 변수 = 함수 이름(입력 변수)”의 형태로 함수를 정의함
- 기 작성된 factorial.m을 실행하는 방법

명령창에서 M-파일 함수명 factorial을 사용 후 전달인자로 5(또는 10)를 사용하여 5!값을 계산해서 리턴



핵심정리

기본적인 그래프 함수 명령어

- 기본적인 2차원 그래프 명령어 : plot, stem, subplot, axis, xlabel, ylabel, title, grid 등
- help 명령어: 특별한 명령어의 사용법에 대한 도움말

흐름 제어 명령어

- if, elseif, else 명령어, for, while, switch 명령어

M-파일 프로그래밍

- Script-mode M-파일: 수행해야 할 Matlab 명령어들을 모아놓은 명령어 파일
- Function mode M-파일: 수행하고자 하는 임의의 모듈을 함수 형태로 표현하며 입력 매개변수와 출력 매개변수를 다루는 함수 양식으로 프로그래밍하는 것을 의미