

처음 배우는 매트랩

[연습문제 공개용 답안 이용 안내]

- 본 연습문제 및 해답의 저작권은 방성완과 한빛아카데미(주)에 있습니다.
- 이 자료를 무단으로 전제하거나 배포할 경우 저작권법 136 조에 의거하여 최고 5 년 이하의 징역 또는 5 천만원 이하의 벌금에 처할 수 있고 이를 병과(併科)할 수도 있습니다.

Chapter 09 연습문제

9.1

```
complexroot.m
1 - x = input('Enter a value for x: ');
2 - y = input('Enter a value for y: ');
3 - z = x + 1i*y;
4 - r = sqrt(x^2+y^2);
5 - arg = atan(y/x);
6 - n = input('Enter a number of root n: ');
7 - for k = 0:n-1
8 -     w_k = (r)^(1/n)*(cos((arg+2*k*pi)/n)+1i*sin((arg+2*k*pi)/n));
9 -     disp('n roots: ');
10 -     w_k
11 - end
```

$k=0$ 이면 $w_0 = \cos \frac{\pi}{6} + i \sin \frac{\pi}{6} = \frac{\sqrt{3}}{2} + \frac{1}{2}i$, $k=1$ 이면 $w_1 = \cos \frac{5\pi}{6} + i \sin \frac{5\pi}{6} = -\frac{\sqrt{3}}{2} + \frac{1}{2}i$
 $k=2$ 이면 $w_2 = \cos \frac{3\pi}{2} + i \sin \frac{3\pi}{2} = -i$ 의 세 제곱근 3 개를 찾을 수 있다.

```
Command Window
>> complexroot
Enter a value for x: 0
Enter a value for y: 1
Enter a number of root n: 3
n roots:
w_k =
    0.8660 + 0.5000i
n roots:
w_k =
   -0.8660 + 0.5000i
n roots:
w_k =
   -0.0000 - 1.0000i
```

9.2

```
complexdiv.m x +
1 - x1 = input('Enter a value for x1: ');
2 - y1 = input('Enter a value for y1: ');
3 - r1 = sqrt(x1^2+y1^2);
4 - arg1 = 180 + atan(y1/x1)*(180/pi);
5 - x2 = input('Enter a value for x2: ');
6 - y2 = input('Enter a value for y2: ');
7 - r2 = sqrt(x2^2+y2^2);
8 - arg2 = atan(y2/x2)*(180/pi);
9 - r = r1/r2;
10 - arg = arg1-arg2;
11 - disp('magnitude:');
12 - disp(r)
13 - disp('argument:');
14 - disp(arg)
```

계산된 복소수의 나눗셈 극 형식은 $z=1\angle 210^\circ$ 가 된다.

```
Command Window
>> complexdiv
Enter a value for x1: -sqrt(3)
Enter a value for y1: 1
Enter a value for x2: 1
Enter a value for y2: -sqrt(3)
magnitude:
    1
argument:
    210
```

9.3

```
Cramer.m  x +
1 - A = input('Enter a 2-by-2 matrix A: ');
2 - c = input('Enter a 2-by-1 matrix c: ');
3 - D = A(1)*A(4)-A(2)*A(3);
4 - A1 = [c(:,1),A(:,2)];
5 - A2 = [A(:,1),c(:,1)];
6 - D_A1 = A1(1)*A1(4)-A1(2)*A1(3);
7 - D_A2 = A2(1)*A2(4)-A2(2)*A2(3);
8 - x1 = D_A1/D;
9 - x2 = D_A2/D;
10 - disp('x1 = ');
11 - disp(x1)
12 - disp('x2 = ');
13 - disp(x2)
```

크래머 법칙을 실행하여 구한 미지의 변수값은 $x_1 = 1$, $x_2 = 2$ 이다.

```
Command Window
>> Cramer
Enter a 2-by-2 matrix A: [4 1;3 2]
Enter a 2-by-1 matrix c: [6;7]
x1 =
    1
x2 =
    2
```

9.4

```
hyper_tri.m  x +
1 - x = input('Enter a value for x: ');
2 - y = input('Enter a value for y: ');
3 - z = x + 1i*y;
4 - sinh_z = (exp(z)-exp(-z))/2;
5 - cosh_z = (exp(z)+exp(-z))/2;
6 - disp('hyperbolic sin =');
7 - disp(sinh_z)
8 - disp('hyperbolic cos =');
9 - disp(cosh_z)
```

정의된 쌍곡선 함수를 입력한다.

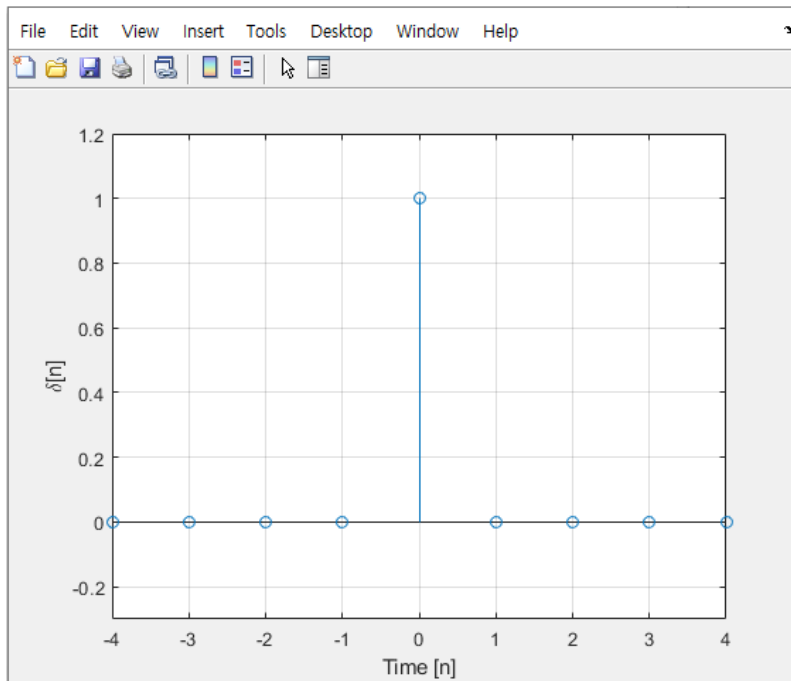
```
Command Window
>> hyper_tri
Enter a value for x: 1
Enter a value for y: pi/3
hyperbolic sin =
    0.5876 + 1.3363i
hyperbolic cos =
    0.7715 + 1.0178i
```

9.5

```
impulse_fun.m
1 - n = input('Enter time interval = ');
2 - u0 = stepfun(n,0);
3 - u1 = stepfun(n,1);
4 - delta_n = u0 - u1;
5 - stem(n, delta_n);
6 - axis([-4 4 -0.3 1.2]);
7 - grid on;
8 - xlabel('Time [n]');
9 - ylabel('#delta[n]');
```

다음과 같이 명령창에 작성한 스크립트 파일을 실행하면 임펄스 함수의 그래프를 나타낼 수 있다. 임펄스 함수는 시간 0에서만 그 값이 1이고 그 외의 모든 시간에서는 0이다.

```
Command Window
>> impulse_fun
Enter time interval = -4:1:4
```



9.6

```

rec2cyl.m  x  +
1 -  x = input('x value = ');
2 -  y = input('y value = ');
3 -  z = input('z value = ');
4 -  rho = sqrt(x^2 + y^2);
5 -  phi = atan2(y,x)+(180/pi);
6 -  disp('rho = ');
7 -  disp(rho)
8 -  disp('phi = ');
9 -  disp(phi)
10 - disp('z = ');
11 - disp(z)

```

직교좌표 (x, y, z) 를 이용하여 원통좌표의 ρ 와 ϕ 를 구하고 직교좌표 z 과 동일한 원통좌표 z 는 유지한다. 원통좌표의 ϕ 를 구하는 과정에서 atan2 함수를 이용하여도 단위의 각도를 구한다.

```

Command Window
>> rec2cyl
x value = 1
y value = 2
z value = -3
rho =
    2.2361
phi =
    63.4349
z =
    -3

```

9.7

```

sph2rec.m
1 - r = input('r value = ');
2 - theta = input('theta value = ');
3 - phi = input('phi value = ');
4 - x = r*sin(theta*pi/180)*cos(phi*pi/180);
5 - y = r*sin(theta*pi/180)*sin(phi*pi/180);
6 - z = r*cos(theta*pi/180);
7 - disp('x = ');
8 - disp(x)
9 - disp('y = ');
10 - disp(y)
11 - disp('z = ');
12 - disp(z)

```

구면좌표 (r, θ, ϕ) 를 이용하여 직교좌표의 (x, y, z) 를 구한다.

```

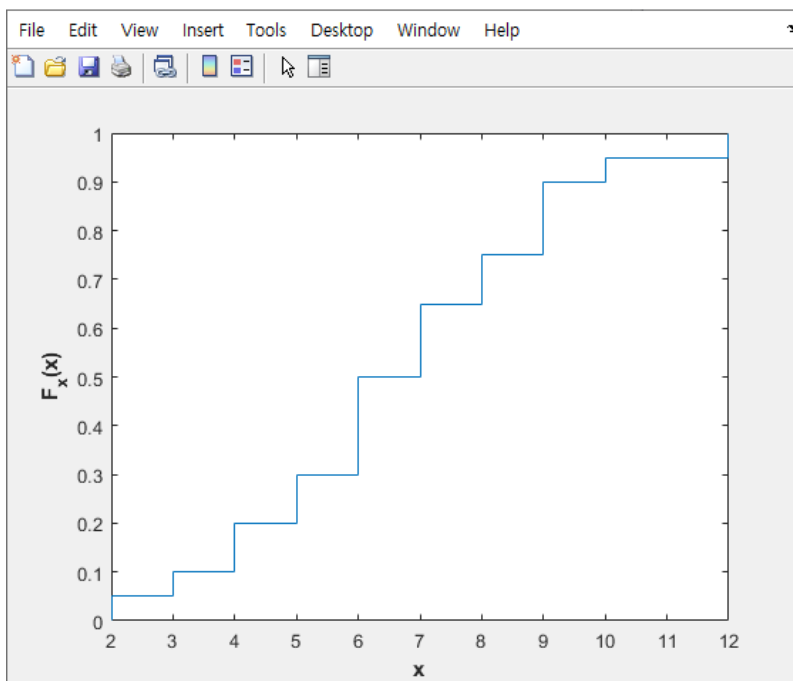
Command Window
>> sph2rec
r value = sqrt(14)
theta value = 74.4986
phi value = -56.3099
x =
    2.0000
y =
   -3.0000
z =
    1.0000

```

9.8

```
cumul_dist.m
1 - d = input('numbers of die: ');
2 - r = input('trial of roll a die: ');
3 - x = sum(ceil(6*rand(d,r)));
4 - stairs([min(x) sort(x)],(0:1/length(x):1));
5 - xlabel('#bf{x}');
6 - ylabel('#bf{F_x(x)}');
```

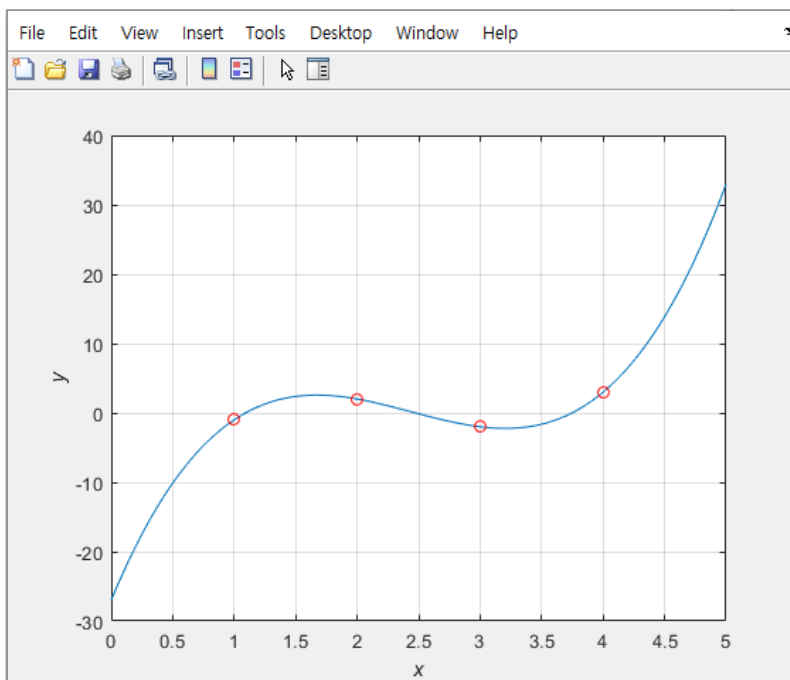
```
Command Window
>> cumul_dist
numbers of die: 2
trial of roll a die: 20
```



9.9

```
cubic_pol.m
1 - xi = input('Enter x_i vector: ');
2 - yi = input('Enter y_i vector: ');
3 - [N,X] = meshgrid(0:3, xi);
4 - A = X.^N;
5 - b = yi;
6 - c = A\b;
7 - f = @(x) c(1) + c(2)*x + c(3)*x.^2 + c(4)*x.^3;
8 - x = linspace(0,5,501);
9 - plot(x,f(x), xi,yi,'ro');
10 - grid on;
11 - xlabel('x');
12 - ylabel('y');
```

```
Command Window
>> cubic_pol
Enter x_i vector: [1 2 3 4]'
Enter y_i vector: [-1 2 -2 3]'
```



9.10

```
nn_AND.m
1 - p = input('Enter input vector: ');
2 - t = input('Enter target vector: ');
3 - net = perceptron;
4 - net = configure(net,p,t);
5 - net.iw{1,1} = rands(1,2);
6 - net.b{1} = 1;
7 - net = train(net,p,t);
8 - w = net.iw{1,1};
9 - b = net.b{1};
10 - plotpv(p,t);
11 - plotpc(w,b);
```

Command Window

```
>> nn_AND
Enter input vector: [0 0 1 1;0 1 0 1]
Enter target vector: [0 0 0 1]
```

