

1 課題1

1.1

以下の命令を実行し, 実行結果について説明せよ.

- `ones(4, 3)`

```
--> ones(4, 3)
ans =
```

```
1.    1.    1.
1.    1.    1.
1.    1.    1.
1.    1.    1.
```

すべての要素が1の4行3列の行列を作成する。

- `eye(5, 3)`

```
--> eye(5, 3)
ans =
```

```
1.    0.    0.
0.    1.    0.
0.    0.    1.
0.    0.    0.
0.    0.    0.
```

対角要素が1の5行3列の行列を作成する。

- `zeros(3, 4)`

```
--> zeros(3,4)
ans =
```

```
0.    0.    0.    0.
0.    0.    0.    0.
0.    0.    0.    0.
```

すべての要素が 0 の 3 行 4 列の行列を作成する。

- `linspace(-1, 1, 10)`

```
--> linspace(-1,1,10)
ans =
```

```
column 1 to 3
```

```
-1. -0.7777778 -0.5555556
```

```
column 4 to 5
```

```
-0.3333333 -0.1111111
```

```
column 6 to 7
```

```
0.1111111 0.3333333
```

```
column 8 to 10
```

```
0.5555556 0.7777778 1.
```

-1 から 1 へ向かう等間隔に並んだ 10 個の点を含んだ一次元配列を作成する。

1.2

`v = [1; -2]; w = [1 2]; A = [1 2; 3 4];` を実行し, 以下の問いに答えよ.

1.2.1 1-2-1

`size` 関数について以下の命令を実行し, 実行結果を比較せよ.

- `size(v)`

```
--> size(v)
ans =
```

```
2. 1.
```

2行1列であることを示している。

- size(w)

```
--> size(w)
ans =
```

```
1. 2.
```

1行2列であることを示している。

- size(A)

```
--> size(A)
ans =
```

```
2. 2.
```

2行2列であることを示している。

1.2.2 1-2-2

max 関数について以下の命令を実行し、実行結果を比較せよ。

- max(A)

```
--> max(A)
ans =
```

```
4.
```

A のすべての要素に対して最大の要素を返している。

- `max(A, 'c')`

```
--> max(A, 'c')  
ans =
```

```
2.  
4.
```

Aの各行に対して最大の要素を返している。

- `max(A, 'r')`

```
--> max(A, 'r')  
ans =
```

```
3. 4.
```

Aの各列に対して最大の要素を返している。

1.2.3 1-2-3

`sum` 関数について以下の命令を実行し, 実行結果を比較せよ.

- `sum(A)`

```
--> sum(A)  
ans =
```

```
10.
```

Aのすべての要素を加算した値を返している。

- `sum(A, 1)`

```
--> sum(A, 1)  
ans =
```

```
4. 6.
```

A の各行に対して加算した値を返している。

- `sum(A, 2)`

```
--> sum(A, 2)
ans =
```

```
3.
7.
```

A の各列に対して加算した値を返している。

1.2.4 1-2-4

以下の命令を実行し, 実行結果について説明せよ.

- `norm(v)`

```
--> norm(v)
ans =
```

```
2.236068
```

そのベクトルのノルムを返している。

- `gsort(v)`

```
--> gsort(v)
ans =
```

```
1.
-2.
```

要素に対してソートを行って、大きい値順に返している。

- `abs(v)`

```
--> abs(v)
ans =
```

```
1.
2.
```

絶対値を取った値を返している。

- $\text{inv}(A)$

```
--> inv(A)
ans =
```

```
-2.    1.
1.5 -0.5
```

逆行列を返している。

2 課題2

2.1

A と v を変数 A と v へ代入せよ.

```
--> A = [4, -2, 0; -1, 4, -2; 0, -1, 4]
A =
```

```
4. -2. 0.
-1. 4. -2.
0. -1. 4.
-> v = [3; 0; 1.5]
v =
```

```
3.
0.
1.5
```

2.2

Av の計算結果を示せ.

```
--> A * v
ans =

    12.
   -6.
    6.
```

2.3

ベクトル v の 2 ノルム $\|v\|_2$ を求めよ. Scilab の norm 関数を用いてもよい.

```
-> norm(v)
ans =

    3.354102
```

2.4

線形方程式 $Ax = v$ の解 x を求めよ.

```
--> A \ v
ans =

    1.
    0.5
    0.5
```

3 課題 3

公式の左辺と右辺を計算せよ.

- 値を代入する。

```
--> A = [1, -1, 0; -1, 2, -1; 0, -1, 2], x = [1;2;3], y = [-2; 2; 1]
A =
```

```
1. -1. 0.
-1. 2. -1.
0. -1. 2.
```

```
x =
```

```
1.
2.
3.
```

```
y =
```

```
-2.
2.
1.
```

- 左辺を計算する。

```
--> inv(A + x * y')
ans =
```

```
5. 0. -1.
3.8 0.2 -0.8
2.2 -0.2 -0.2
```

- 右辺の計算する。

```
--> inv(A) - (1 / (1 + y' * inv(A) * x))*(inv(A)*x)*(y'*inv(A))
ans =
```

```
5. 0. -1.
3.8 0.2 -0.8
2.2 -0.2 -0.2
```


4 課題 4

4.1

gsort 関数と abs 関数を用いて, 1 次元配列の絶対値最小の要素と絶対値が 2 番目に小さい要素を返す関数を作成せよ. ただし, 返す値は絶対値ではなく, もとの要素の値とすること。

```
function [val1, val2] = myfunc(vec)
[vals, idxs] = gsort(abs(vec))
idxs = flipdim(idxs, 2)
val1 = vec(idxs(1))
val2 = vec(idxs(2))
endfunction
```

4.2

```
--> [val1, val2] = myfunc(datas)
val2 =

-0.048493

val1 =

-0.0278533
```