

# P04

## Linear Algebra

Self-Scoring table

	P01	P02	P03	E01	E02
Score	1	1	1	1	1

# Practice

## -P01

Part1	Part2
<pre>vin@gwonbin-ui-MacBookAir EXE % ./P01 3x1 vectors a = vec3(0.000000, 0.000000, 0.000000) b = vec3(3.000000, 2.000000, 1.000000) a = vec3(1.000000, 2.000000, 3.000000) a[0] = 1 a.z = 3</pre>	<pre>vin@gwonbin-ui-MacBookAir EXE % ./P01 2 Vector operations a = vec3(1.000000, 2.000000, 3.000000) b = vec3(3.000000, 2.000000, 1.000000) a + b = vec3(4.000000, 4.000000, 4.000000) a - b = vec3(-2.000000, 0.000000, 2.000000) -a = vec3(-1.000000, -2.000000, -3.000000) 1.5*a = vec3(1.500000, 3.000000, 4.500000) dot(a,b) = 10 cross(a,b) = vec3(-4.000000, 8.000000, -4.000000) length(a) = 3.74166</pre>

## -P02

Part3
<pre>vin@gwonbin-ui-MacBookAir EXE % ./P01 4 Matrix operations A = mat3x3((1.000000, 0.000000, 0.000000), (0.000000, 1.000000, 0.000000), (0.000000, 0.000000, 1.000000)) B = mat3x3((1.000000, 0.000000, 0.000000), (2.000000, 1.000000, 0.000000), (3.000000, 0.000000, 1.000000)) A + B = mat3x3((2.000000, 0.000000, 0.000000), (2.000000, 2.000000, 0.000000), (3.000000, 0.000000, 2.000000)) A - B = mat3x3((0.000000, 0.000000, 0.000000), (-2.000000, 0.000000, 0.000000), (-3.000000, 0.000000, 0.000000)) -A = mat3x3((-1.000000, -0.000000, -0.000000), (-0.000000, -1.000000, -0.000000), (-0.000000, -0.000000, -1.000000)) 1.5*A = mat3x3((1.500000, 0.000000, 0.000000), (0.000000, 1.500000, 0.000000), (0.000000, 0.000000, 1.500000)) A x B = mat3x3((1.000000, 0.000000, 0.000000), (2.000000, 1.000000, 0.000000), (3.000000, 0.000000, 1.000000)) transpose(B) = mat3x3((1.000000, 2.000000, 3.000000), (0.000000, 1.000000, 0.000000), (0.000000, 0.000000, 1.000000)) inverse(B) = mat3x3((1.000000, 0.000000, 0.000000), (-2.000000, 1.000000, 0.000000), (-3.000000, 0.000000, 1.000000))  inverse(B) * B = mat3x3((1.000000, 0.000000, 0.000000), (0.000000, 1.000000, 0.000000), (0.000000, 0.000000, 1.000000))</pre>
Part4
<pre>vin@gwonbin-ui-MacBookAir EXE % ./P01 3 3x3 matrices A = mat3x3((0.000000, 0.000000, 0.000000), (0.000000, 0.000000, 0.000000), (0.000000, 0.000000, 0.000000)) A = mat3x3((1.000000, 0.000000, 0.000000), (0.000000, 1.000000, 0.000000), (0.000000, 0.000000, 1.000000)) B = mat3x3((1.000000, 0.000000, 0.000000), (2.000000, 1.000000, 0.000000), (3.000000, 0.000000, 1.000000)) B = mat3x3((1.000000, 0.000000, 0.000000), (2.000000, 1.000000, 0.000000), (3.000000, 0.000000, 1.000000)) 3rd col of B = vec3(3.000000, 0.000000, 1.000000) 3rd col B = vec3(3.000000, 0.000000, 1.000000) 3rd row of B = vec3(0.000000, 0.000000, 1.000000) 1st row 3rd col of B = 3 1st row 3rd col of B = 3</pre>

## -P03

### Part5

```
vin@gwonbin-ui-MacBookAir EXE % ./P01 5
Matrix-vector multiplication and assembling
a = vec3(1.000000, 2.000000, 3.000000)
B = mat3x3((1.000000, 0.000000, 0.000000), (2.000000, 1.000000, 0.000000), (3.000000, 0.000000, 1.000000))
B x a = vec3(14.000000, 2.000000, 3.000000)
a x B = vec3(1.000000, 4.000000, 6.000000)
(a,1,0) = vec4(1.000000, 2.000000, 3.000000, 1.000000)
(1,0,a) = vec4(1.000000, 1.000000, 2.000000, 3.000000)
C = mat4x4((1.000000, 0.000000, 0.000000, 0.000000), (2.000000, 1.000000, 0.000000, 0.000000), (3.000000, 0.000000, 1.000000, 0.000000), (0.000000, 0.000000, 0.000000, 1.000000))
```

## Exercise

### -E01

#### Vector operations

```
vin@gwonbin-ui-MacBookAir EXE % ./E01
Compute the following vector operations
V1 + V2 = vec3(4.000000, 4.000000, 4.000000)
V1 - V2 = vec3(-2.000000, 0.000000, 2.000000)
-V1 = vec3(-1.000000, -2.000000, -3.000000)
V1 - 2*V2 = vec3(-5.000000, -2.000000, 1.000000)
V1 · V2 = 10.000000
V1 X V2 = vec3(-4.000000, 8.000000, -4.000000)
```

위에서  $V1=(1,2,3)$ ,  $V2=(3,2,1)$ 이다.

3차원 벡터의 합은 각 x,y,z값의 합으로  $V1+V2=(1+3,2+2,3+1)$ 이다.

3차원 벡터의 차는 앞벡터 x,y,z값 - 뒤벡터 x,y,z값 이므로

$V1-V2=(1-3,2-2,3-1)=(-2,0,2)$ 이다.

벡터의 스칼라배는 각 값에 모두 스칼라 배를 해주므로  $-V1=(-1,-2,-3)$ 이다.

$V1-2*V2$ 는 위의 방법들을 이용하면 구할 수 있고

$V1 \cdot V2$ 는 벡터의 내적으로 스칼라값 10이 나오고

$V1 \times V2$ 는 벡터의 외적으로  $(-4,8,-4)$ 가 나온다.

## -E02

### Matrix operations

```
vin@gwonbin-ui-MacBookAir EXE % ./E01 2
Compute the following matrix operations
A1 + A2 = mat3x3((3.000000, 4.000000, 2.000000), (3.000000, 5.000000, 2.000000), (5.000000, 3.000000, 3.000000))
A1 - A2 = mat3x3((-1.000000, 0.000000, 0.000000), (1.000000, 1.000000, 0.000000), (1.000000, 1.000000, 1.000000))
-A1 = mat3x3((-1.000000, -2.000000, -1.000000), (-2.000000, -3.000000, -1.000000), (-3.000000, -2.000000, -2.000000))
A1 - 2*A2 = mat3x3((-3.000000, -2.000000, -1.000000), (0.000000, -1.000000, -1.000000), (-1.000000, 0.000000, 0.000000))
)
A1*A2 = mat3x3((9.000000, 12.000000, 6.000000), (8.000000, 10.000000, 5.000000), (7.000000, 9.000000, 5.000000))
A2*A1 = mat3x3((6.000000, 7.000000, 4.000000), (9.000000, 11.000000, 6.000000), (12.000000, 12.000000, 7.000000))
A1*V1 = vec3(14.000000, 14.000000, 9.000000)
A2*V2 = vec3(10.000000, 11.000000, 6.000000)
```

행렬끼리의 합은 같은 위치에 있는 항들을 더하여 나오고

행렬의 차는 같은 위치에 있는 항들을 빼서 나온다.

행렬의 스칼라 곱은 각 항에 모두 스칼라 곱을 해주면 되고

행렬의 곱은 두행렬의 내적으로 구하는데 앞행렬의 i번째 행과 뒤행렬의 j번째 열의 내적을

계산하여서 새롭게 만들어지는 행렬의 i,j요소를 구한다.

행렬과 벡터의 곱에 경우 벡터를 행렬의 열로 간주하여 행렬의 각 행과 벡터의 내적을

계산하여 새로운 벡터를 생성한다.