

2021년 1학기

7

제어지능SW개발

openCV를 활용한 지능형시스템 개발

교수 도경민

컴퓨터전자공학과



7차시 내용

- 6주차 자기평가 질문
- 5주차 과제 해결
- 영상 이해
- 배열
- numpy

- 문자를 list로 받는것은 어느정도 익숙해졌지만 문자열로 바로 받는 방법을 잘 모르겠습니다. 예를 들어 반복문 실행후 나온 결과값이 a,b,c,d 일때 문자열x에 이를 대입하면 출력시 마지막에 입력된 d 한개만 찍힘.
- 진도보다 저희들의 이해도를 우선시해주셔서 감사합니다.
- 없습니다. 이번주 화상강의가 많이도움이 됐습니다. 자신감을 얻었습니다!
- 처음에는 엄청 귀찮았던 노트정리를 함으로 못풀고 포기했던 문제도 차근차근 정리하면서 하나씩 해결해갈수있었습니다.
- 6주차 수업을 들으면서 문제를 프로그래밍 하기전 나름 정리를 한다고 생각하였는데 턱 없이 부족한 정리와 정보,해결방법을 제대로 생각해보지 않고있었다고 느꼈습니다. 프로그래밍을 할때 노트에 연습하는 습관을 가져야 할것 같습니다.
- 확실히 연습장에 문제에 대한 생각을 코드로 나타낼수 있을 때까지 정리하고 나니 프로그래밍을 하는것이 훨씬 수월해졌습니다. 프로그래밍이 더 재밌게 느껴집니다.

- 6주차는 실시간 화상강의로 이루어 졌는데 6주차 내용인 영상,배열,numpy에 대한 설명 영상이 있으면 좋겠습니다. 아무런 영상없이 pdf파일과 인터넷만 보고 그 내용들을 이해하는데에는 문제가 있고 그에 따라 시간이 많이 소요되어 과제, 자기평가의 기간 또한 조정이 필요할 것 같습니다.
- 질문이 없습니다! 오늘 줌수업 고생하셨습니다! 얼굴뵈고 수업을 들으니깐 집중이 더 잘 됩니다!
- 이번 강의로 노트에 생각을 구체적이고 체계적으로 정리하는 방법을 찾았습니다. 앞으로 정리하는 과정을 불편하게 생각하지 않을 것 같습니다.
- 이번 수업은 아직 진도를 못나가셔서 질문 없습니다!
- 이번 6주차 강의를 통해 프로그래밍을 하는 것에 대한 새로운 마음가짐을 갖게 된 것 같습니다. 주어진 상황을 해결할 때 우선 필요한 주 요소가 무엇인 지 명확하게 정리하고, 혹여나 오류 또는 문제가 발생하지 않도록 세세하게 점검하여 차후 모든 과정이 완료되었을 때 '피드백이 생기지 않도록 만드는 것이 가장 중요하다' 라는 생각을 다시 하게 되었습니다.
- 과제 풀이를 해주시는 과정을 보고 문제를 단계적으로 해결한다는 것이 큰 도움이 되었습니다.
- 구체적인 프로그래밍 공부방법을 알려주셔서 도움이 많이 되었습니다.

- 실강에서 교수님의 방법으로 4주차 과제를 다시 풀어보니 처음이라 시간은 오래걸려도 과제 해결이 조금 더 수월했습니다 감사합니다
- 6주차 실시간 강의는 4주차 복습을 했기에 4주차 과제를 제출한 저로써는 이미 한번 이해했고 한번 제출한 과제이기에 어려움 없는 것 같습니다만 교수님이 말씀해주신 과제를 해결할 때에 연습장에 자신의 생각을 정리하는 것이 쉽지 않습니다.

특히나 자신만의 생각을 정리하는 부분에 있어, 처음 단계인 예를 들어가며 진행하는 부분은 쉽습니다만 그것을 풀어서 프로그래밍 언어로 나타내는 부분이 어렵습니다. 하지만 이는 많은 연습을 통해 해결할 수 있을 것 같기에 많은 연습을 할 예정입니다.

* python에서 제공되는 함수를 사용하지 않고 프로그램을 작성하시오.

- 1) 입력 문자열을 알파벳 순으로 포함된 알파벳 수를 출력하시오.

ex) morning

```
g : 1
i : 1
m : 1
n : 2
o : 1
r : 1
```

- 2) 입력 문자열을 알파벳 순으로 포함된 알파벳 수를 오른쪽과 같이 출력하시오.
빈칸은 제외한다.

ex) nice morning

```
          n
        i  n
c e g i m n o r
```

과제 5-1

7

* python에서 제공되는 함수를 사용하지 않고 프로그램을 작성하시오.

2) 입력 문자열을 알파벳 순으로 포함된
알파벳 수를 오른쪽과 같이 출력하시오.
빈칸은 제외한다.

ex) nice morning

```
      n
    i  n
c e g i m n o r
```



- 프로그램 실행 중 발생하는 오류 처리
 - try
 - except
 - else
 - finally
 - raise



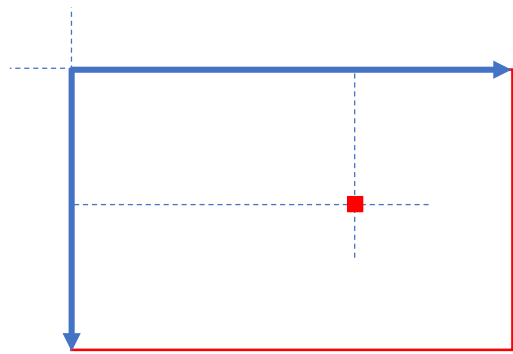
제어지능SW개발

영상 처리

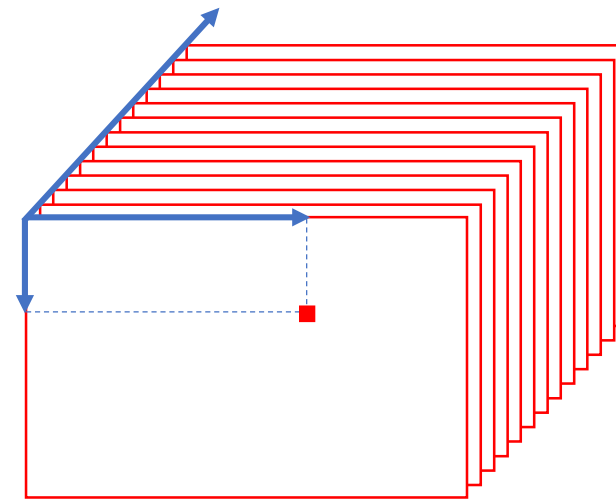




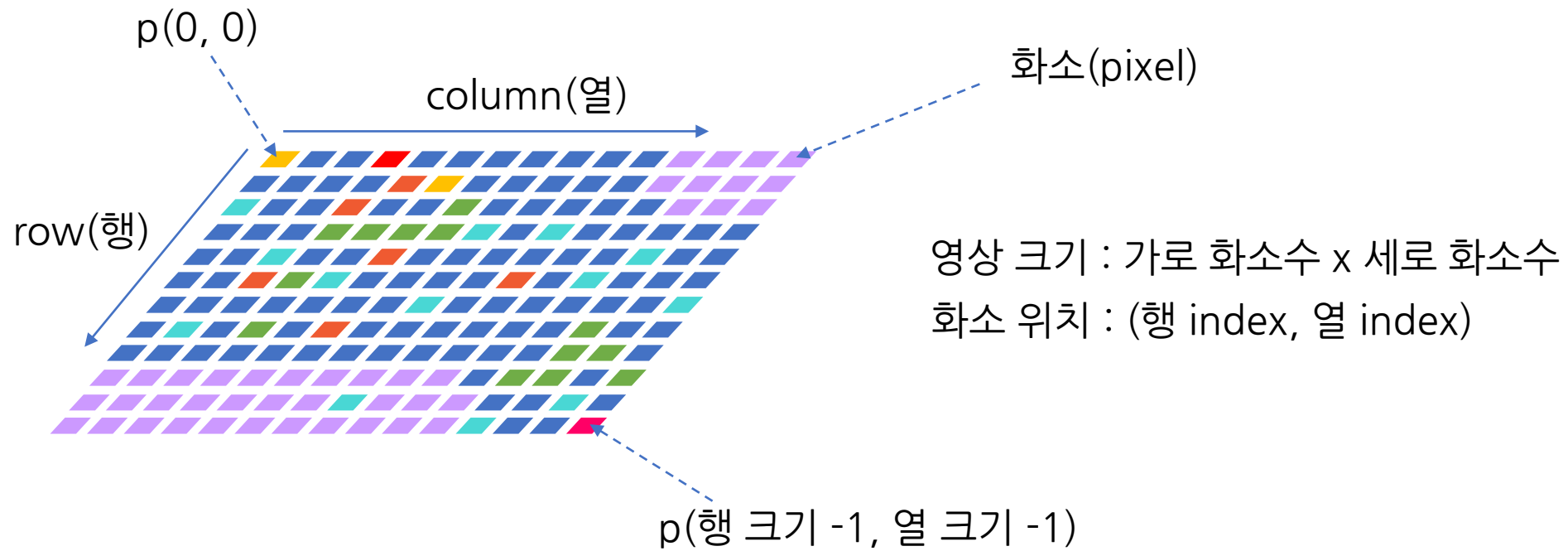
- Image Digitization



image

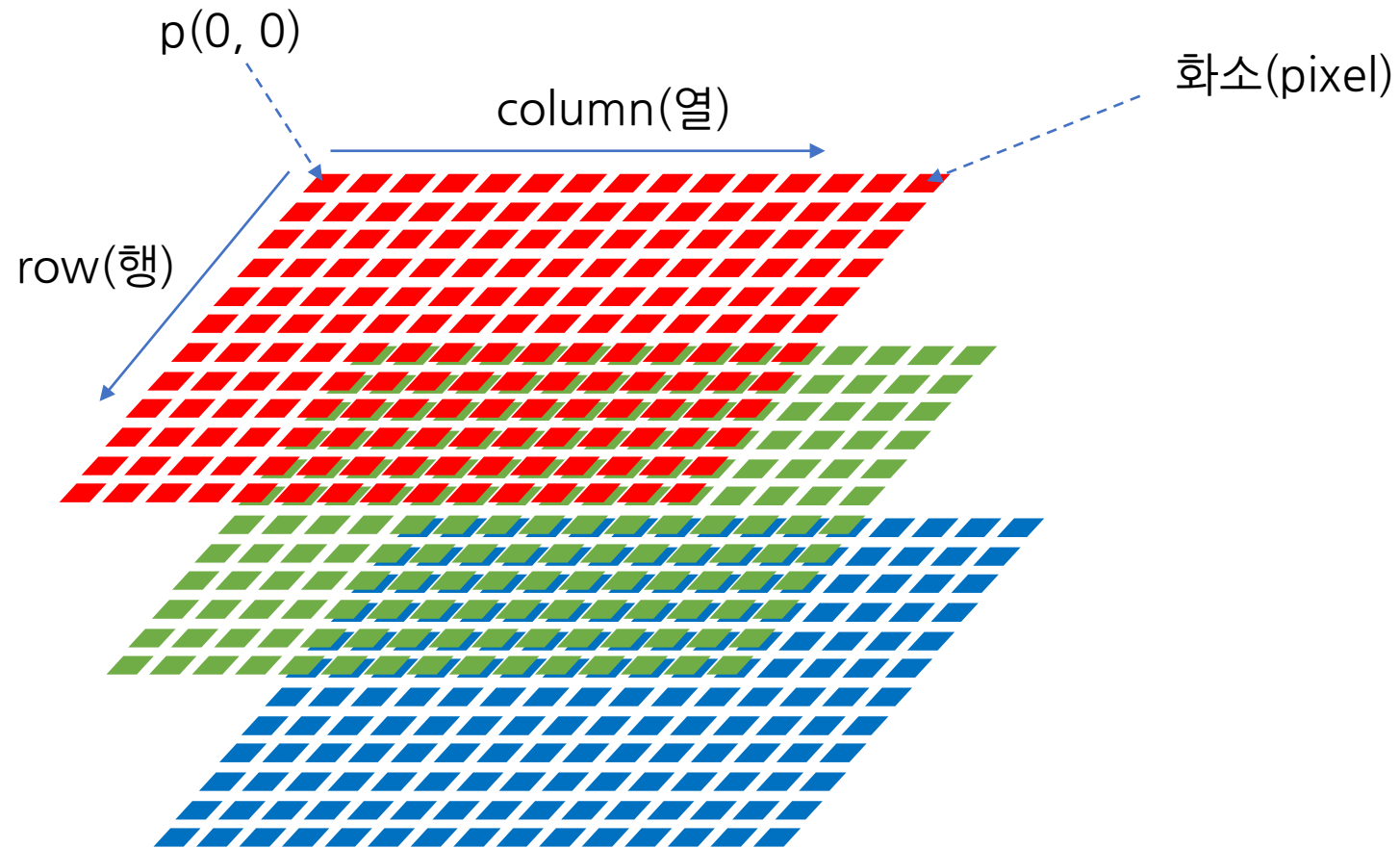


video

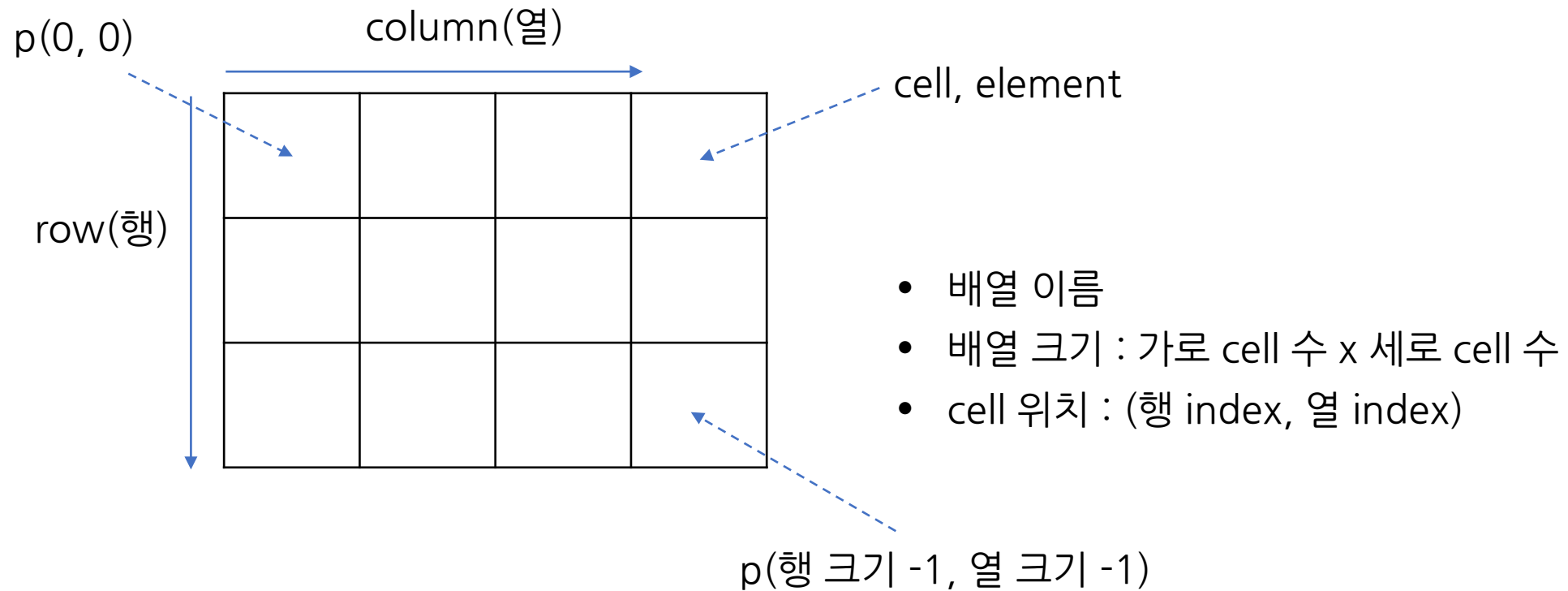


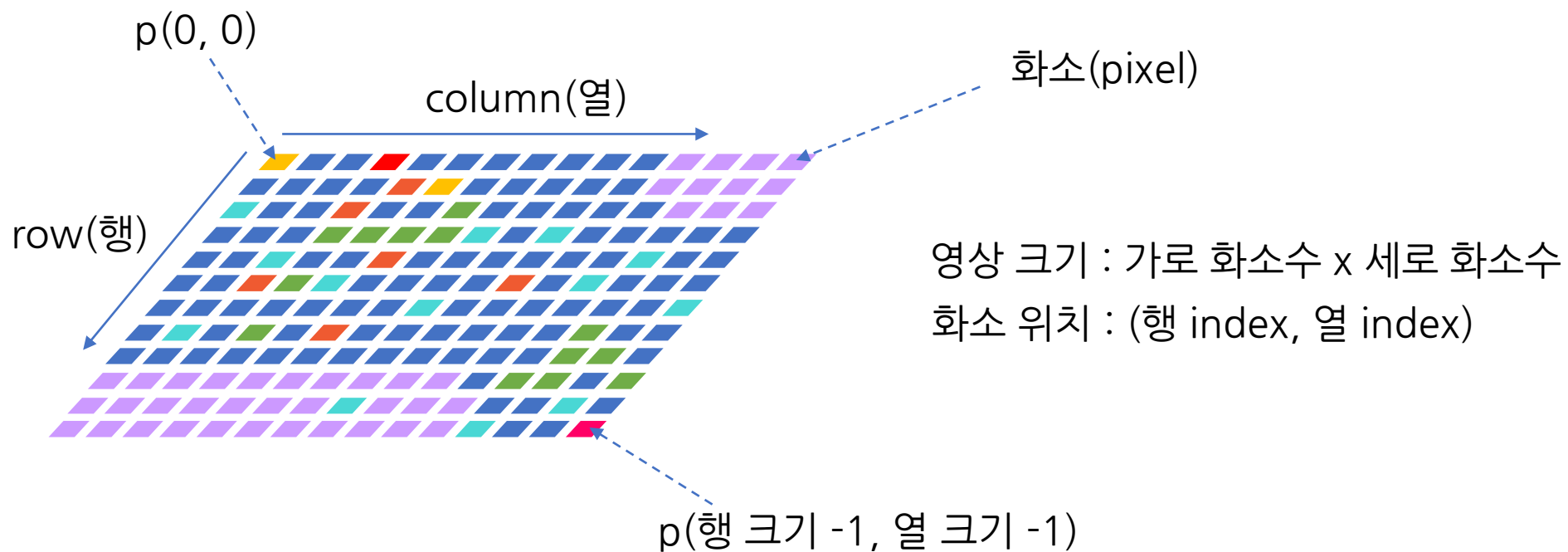


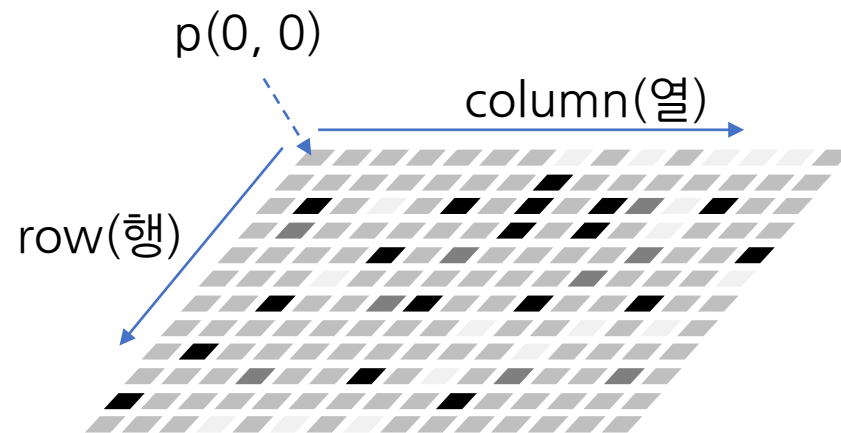
칼라 평면(plane)



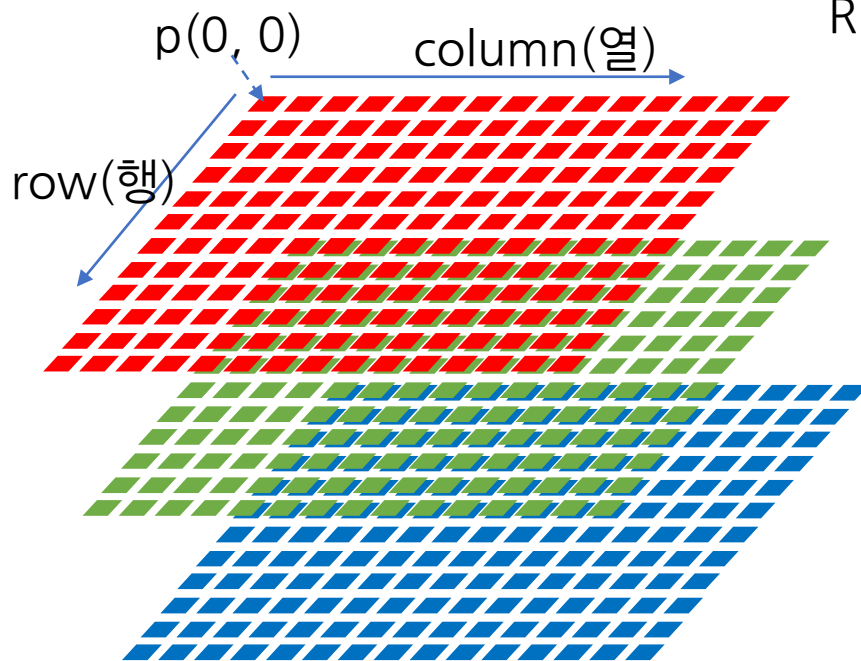
영상 크기	pixel resolution (bit)	영상 종류	영상 데이터 용량 (byte)
128 x 128	1	black & white	
256 x 256	1	black & white	
1024 x 768	1	black & white	
128 x 128	8	gray	
256 x 256	8	gray	
1024 x 768	8	gray	
128 x 128	24	color(RGB 각 8bit)	
256 x 256	24	color(RGB 각 8bit)	
1024 x 768	24	color(RGB 각 8bit)	







24	12	128	255	
25	0	0	255	
102	67	86	220	



R

87	126	42	90
1	24	2	15
43	168	126	240

G

32	124	2	34
55	12	145	200
0	3	2	58

B

24	12	128	255
25	0	0	255
102	67	86	220

- 다차원의 배열 사용 가능
- “ndarray” class 사용
- 2차원 이상에서 axis사용
 - axis = 0 : 열
 - axis = 1 : 행
 - axis = -1 : 층
- `python.exe -m pip install --upgrade pip`
- `python -m pip install --user numpy scipy matplotlib ipython jupyter pandas sympy nose`



- numpy.dtype

Numpy type	C type
np.bool_	bool
np.byte	signed char
np.ubyte	unsigned char
np.short	short
np.ushort	unsigned short
np.intc	int
np.uintc	unsigned int
np.int_	long
np.uint	unsigned long
np.longlong	long long
np.ulonglong	unsigned long long
np.half / np.float16	
np.single	float
np.double	double
np.longdouble	long double
np.csingle	float complex
np.cdouble	double complex
np.clongdouble	long double complex

Numpy type	C type
np.int8	int8_t
np.int16	int16_t
np.int32	int32_t
np.int64	int64_t
np.uint8	uint8_t
np.uint16	uint16_t
np.uint32	uint32_t
np.uint64	uint64_t
np.intp	intptr_t
np.uintp	uintptr_t
np.float32	float
np.float64 / np.float_	double
np.complex64	float complex
np.complex128 / np.complex_	double complex

- `np.array([원소 값, 원소 값 ...])`
- `np.array([원소 값, 원소 값 ...], (원소 값, 원소 값 ...), ..., dtype=데이터 유형)`
- `np.arange(갯수)`
- `np.arange(시작, 불포함 끝), np.arange(시작, 불포함 끝, 증감)`
- `np.arange(시작, 불포함 끝, dtype=데이터 유형)`
- `np.reshape(총수, 행수, 열수)`
- `np.linspace(시작, 포함 끝, 갯수)`
- `np.zeros((행수, 열수))`

- 1차원 : 배열명[위치]
- 2차원 : 배열명[행위치, 열위치], 배열명[행위치][열위치]
- “시작:불포함끝” 사용 가능
- “시작:불포함끝:증감” 사용 가능
- - 값 사용 가능
- 위치 지정을 배열로 사용 가능
 - ex) 배열명[np.array([2,3,4])]
- Boolean 값으로 항목 선택 가능
 - 해당 위치가 True이면 값 선택

- 1) 7부터 26까지의 값을 가지는 4x5 배열을 만들어 배열값을 출력하시오.
- 2) 1부터 24까지의 값을 가지는 1차원 배열을 만들어 출력하고, 이 1차원 배열로 만들 수 있는 2차원 배열을 모두 만들어 배열 크기와 배열값을 출력하시오.

1) 7부터 26까지의 값을 가지는 4x5 배열을 만들어 배열값을 출력하시오.

		column(열)				
		0	1	2	3	4
row (행)	0	7	8	9	10	11
	1	12	13	14	15	16
	2	17	18	19	20	21
	3	22	23	24	25	26

2) 1부터 24까지의 값을 가지는 1차원 배열을 만들어 출력하고, 이 1차원 배열로 만들 수 있는 2차원 배열을 모두 만들어 배열 크기와 배열값을 출력하시오.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24

1부터 30까지의 1차원 배열을 만들고 다음 각 물음을 처리하는 프로그램을 각각 작성하시오.

- 1) 짝수번 항목의 합을 구하여 출력하시오.
- 2) 홀수번 항목의 합과 짝수번 항목의 합의 차를 출력하시오.
- 3) 배열의 정가운데를 기준으로 서로 마주보는 항목을 곱하여 새로운 배열로 저장하여 출력하시오.

1부터 30까지의 1차원 배열을 만들고 다음 각 물음을 처리하는 프로그램을 각각 작성하시오.

1) 짝수번 항목의 합을 구하여 출력하시오.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1부터 30까지의 1차원 배열을 만들고 다음 각 물음을 처리하는 프로그램을 각각 작성하시오.

2) 홀수번 항목의 합과 짝수번 항목의 합의 차를 출력하시오.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

1부터 30까지의 1차원 배열을 만들고 다음 각 물음을 처리하는 프로그램을 각각 작성하시오.

3) 배열의 정가운데를 기준으로 서로 마주보는 항목을 곱하여 새로운 배열로 저장하여 출력하시오.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

- 1) 1부터 순서대로 값을 가지는 9x9 배열을 만들어 출력하시오.
그림은 5x5 배열을 예로 들어 설명하였으니, 프로그램은 9x9 배열로 작성합니다.
이후 문제는 이 배열을 사용하여 프로그래밍 합니다.

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

- 2) 1)의 배열에서 그림과 같은 위치의 항목의 합을 구하시오.
(그림은 5x5 배열을 예로 설명하였음.)

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

- 3) 1)의 배열 안쪽의 5x5 항목값을 제공한 배열을 만들어 출력하시오.
그림은 5x5 배열을 예로 들어 설명하였으니, 프로그램은 9x9 배열로 작성합니다.

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

→

1	6	11	16	21
2	49	144	289	22
3	64	169	324	23
4	81	196	361	24
5	10	15	20	25

- 4) 1)의 배열 안쪽의 5x5 항목값을 대칭한 배열을 만들어 출력하시오.

대각선 항목을 기준으로 대칭합니다.
그림은 5x5 배열을 예로 들어
설명하였으니, 프로그램은 9x9 배열로
작성합니다.

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

→

1	6	11	16	21
2	7	8	9	22
3	12	13	14	23
4	17	18	19	24
5	10	15	20	25