

## 파이썬프로그래밍 2차 레포트

공고 일자: 2020년 11월 25일

### 문제 1: 지정된 형식으로 숫자를 출력하는 함수, print\_num() 설계

1,000 단위 쉼표(.) 넣기와 10,000 이하의 자릿수에 대한 아라비아 숫자와 한글이 아래 사례와 같이 표현되는 함수 print\_num()을 제작하시오.

```
12,334,567,891(1233456만7천8백9십1)
20,500(2만5백)
1,100,007(110만7)
900,001(90만1)
9,028(9천2십8)
100(백)
15(십5)
8(8)
0()
```

- 1) 1,000 단위로 comma(,)를 넣은 숫자를 표현한 후에는 연이어 괄호 안에는 지정한 형식의 한글과 섞어 표현해야 한다.
- 2) '만' 이상 단위는 숫자로 표현한다. 하지만 이때는 ','로 자릿수를 구분하지 않는다.
- 3) 하위 4자리, 0~9999는 한글과 숫자를 섞어서 표현한다.
- 4) 한글 단위 앞에 0이 오면 그 한글 단위는 출력되지 않는다.
- 5) 한글 단위 앞에 1이 오면 '1'자는 표시되지 않지만 그 한글 단위는 출력된다.

인터넷 자료를 그대로 쓰지 못하게 하기 위해서 다소 관습적이지 않은 표현을 사용하였습니다. 양해 바랍니다.

할 일 1: 검증을 위해 아래의 루틴을 소스 상단에 작성해 넣기 바랍니다.

```
int_list = [12334567891, 20500, 1100007, 900001, 9028, 100, 15, 8, 0]
for i in int_list:
    print_num(i)
# num_lines = xxx    print_num()의 작성에 소요된 line의 수
```

## 문제 2: 영상 파일 검색 및 출력 프로그램

다음의 기능을 수행하는 프로그램을 제작한다. 각 기능은 단계별로 수행하여 보일 수도 있고, 종합적으로 완결하여 보일 수도 있다.

1. 디렉토리 안에 존재하는 영상 파일의 수와 총 용량을 출력하기  
스트링 변수, `src_folder`로 지정된 폴더 안에(서브 폴더 포함) 존재하는 모든 파일 중에서 영상 파일을 찾아 그 파일들의 개수와 이름과 파일의 크기의 총합을 파악하여 아래 사례와 같이 출력한다.

Total number of picture files: 32(jpg: 25, png:5, tif:2)  
Total size of files: 31,408,029(3140만5천2십9)

문제의 가정 혹은 제한 조건:

- 1) 소스의 상단에 다음과 같이 폴더의 위치를 지정하는 변수를 선언해야 한다.

`src_folder = 'd:/work/mission/'` or `src_folder = 'mission/'`

- 2) 영상 파일은 “jpg, png, tif, gif” 4 종류로 한정하고, 오류없는 파일로 가정한다.
- 3) 서브 폴더 안에 있는 서브 폴더가 있을 수 있고 그 아래 또 서브폴더가 있을 수 있다. 폴더의 계층 한계는 제한하지 않는다.
- 4) 파일 크기는 1,000 단위로 comma(,)를 넣어 출력해야 한다.
- 5) 본 미션은 DIR 명령어를 사용하는 것은 인정하지 않습니다.

2. 그림 파일 중에서 1) 용량이 제일 큰 파일과, 2) 용량이 제일 적은 파일, 3) 영상의 높이가 높은 파일, 4) 영상의 가로가 제일 넓은 파일 총 4개의 사진을 2x2 서브화면에 출력한다.

1번과 2번 서브 화면의 타이틀에는 그림과 같이 “파일이름:파일용량”을 보이고, 3번과 4번 서브 화면에는 “화면이름:가로x세로” 정보를 타이틀에 출력한다. 파일 이름에는 경로는 포함하지 않아야 하고, 확장자까지 보여야 합니다.

max="bk.PNG":551,177



min="j.png":458



tall="apt.JPG": (1004, 474)



wide="fighters.jpg": (317, 1038)



## 참고 사항

\*.py files with full path

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_1\test1\_dummy.py

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_1\layer\_3\dummy2.py

\*.txt files with full path

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\sample\_A.txt

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_1\ssase.txt

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\sample@1@2\_2@1.txt

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\layer\_3\sample@1@2\_2@3@1.txt

D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\layer\_3\sample@1@2\_2@3@2.txt

\*.jpg, gif, png, tif files with full path

17,950 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\barns.jpg

551,177 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_1\bk.PNG

267,835 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_1\layer\_3\flowers.6IF

458 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\j.png

110,181 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\layer\_3\fighters.jpg

23,298 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\layer\_3\layer\_4\bd.JPG

107,573 : D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_2\apt.JPG

fname\_lst: ['barns.jpg', 'bk.PNG', 'flowers.6IF', 'j.png', 'fighters.jpg', 'bd.JPG', 'apt.JPG']

max\_file=D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_1\bk.PNG

min\_file=D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\j.png

tall\_file=D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_2\apt.JPG

wide\_file=D:/Work/@@Python/LectureMaterials/report 공고/2차/Lab/mission\layer\_1\layer\_1\_2\layer\_3\fighters.jpg

### 문제 3: 영상 파일 암호화 프로그램

src\_folder 변수로 지정한 폴더 및 그 하부 폴더의 모든 파일을 검색하여 이중 “jpg, png, tif, gif” 4 종류 영상 파일에 대해 암호화를 행하여 해당 파일을 src 폴더의 하와 같은 이름으로 dst\_folder 변수로 지정하는 폴더에 저장하는 프로그램을 작성하시오.

1) 소스의 상단에 다음과 같이 폴더의 위치를 지정하는 변수를 선언해야 한다.

```
src_folder = 'd:/work/mission/' or src_folder = 'mission/'  
dst_folder = 'd:/work/mission/' or dst_folder = 'dst/'
```

2. 암호화하기: 파일의 각 바이트별로 주어진 변수 key8이 지정하는 8비트 값과 XOR 동작을 행한 결과를 상위 4비트(nibble), 하위 4비트의 값의 서로 교환하여 암호화한다. 예: XOR 8비트 값이 1011\_0110이라면 저장할 값은 0110\_1011이 됩니다.

3. 복호화하여 저장된 데이터가 올바르게 복원되는지 보이기

dst\_folder에 저장된 영상들을 하나씩 읽어서 상위, 하위 nibble를 교환한 후 여기에 Key8 변수의 값을 XOR 연산을 취한 값을 src\_folder에 있는 원본 영상의 데이터 값과 바이트 단위로 비교하여 모든 데이터의 값이 같으면 그 파일은 정상적으로 저장된 것으로 간주한다.

이 검증 과정에 대한 결과는 아래와 같은 예시 화면으로 출력해서 보인다.

```
#001: abcd.jpg(12,903), pass!!  
location: layer1/layer_1_2  
... (생략) ....  
#034: tmp.png(10,912,903), pass!!  
location: layer1/layer_1_2
```

**공통사항 : 소스의 크기 및 수행시간을 화면에 출력한다.**

아래 사항을 평가 자료로 활용될 수 있습니다.

적은 용량의 소스 프로그램과 고속 처리가 점수 획득에 유리합니다.

1. 프로그램 상단부에 본 프로그램의 총 용량을 다음과 같이 출력하게 합니다.

Program Size = 133,895,398

2. 본 프로그램의 상단부와 종료 직전 time() 함수를 호출하여 수행시간을 소수 4째자리까지 초단위로 출력합니다.

Total Execution Time = 24.3456[sec]

3. 기타 본 공고에서 지정하지 않은 조건, 가정에 대해서는 여러분 스스로 지정해서 문제를 해결하기 바랍니다. 단, 이러한 내용은 보고서와 소스의 도입부에서 미리 밝혀야 합니다.

지금까지 강의 중에 소개한 모듈 이외의 다른 모듈을 이용하는 것은 감점(-) 대상이 될 수 있습니다.

입니다.

## **제출 안내**

1) 제출 기한: 2020년 12월 10일(목) 오후 06:00까지

2) 제출물 - “2. 레포트 제출 안내.pdf” 참조

3) 제출방법

구글 클래스 과제제출

<https://classroom.google.com/c/MTk4OTEzODY2ODEx?cjc=6gnxc5r>