デジタルメディア処理1

担当: 井尻 敬

提出方法: 共有フォルダに 『**dm1学籍番号**』というフォルダを作成し，その中にソースコードの入ったファイルを置く．フォルダ名は全て半角.

フォルダ名の例: dm2AL150999

課題雛形: http://takashiijiri.com/classes/dm2018_1/dm1exer.zip

入出力 ： 課題では入力画像を受け取り，画像またはファイルを保存するプログラムを作る．入出力ファイル名は，以下の例のようにコマンドライン引数より与えるものとする．（※各課題の指定に従うこと）

```
$python   exer*.py   fname_in.png   fname_out.png
```

注意 ：

採点は自動化されています．フォルダ名・ファイル名やプログラムの仕様は指示に厳密に従ってください．入出力の仕様を満たさないコードは評価できず0点扱いとなることがあります．

今回は計算速度を重視しませんが，60秒以上の計算時間がかかるものは，自動採点の都合上0点とします．

各課題について，入出力例を用意するので，作成したプログラムのテストに利用してください．

締め切りと配点

date ：	課題番号	締め切り	配点
• 11/09：	課題01~05	11/13 23:59	各2.0点
• 11/13：	課題06~09	12/08 23:59	各2.5点
• 11/20：	課題10~13	12/08 23:59	各2.5点
• 11/27：	課題14~16	12/08 23:59	各3.3点
• 12/04：	課題17~20	12/08 23:59	各2.5点 (未定)

※ 全問正解で合計 50点

※ 配点/締め切りは，上方に/後ろに修正する可能性があります

注意

- **この課題は，知人同士で相談しながら取り組んでよいです**
  - 教える立場の方は理解が補強でき，教わる方は難しい課題も解けるようになり，メリットは大きいです
  - ただし，教わる人はただ他人のコードをコピーするだけではなく，その部分にどのような意味があるかを確実に理解するようにしてください
- **この課題の解答となるコードを，『この課題の解答と分かる形』でWeb上（GitHub，SNS，個人web page)に公開することは避けてください**
- **知恵袋やteratailなどのナリッジコミュニティサイトにて，問題文をそのまま掲載し，解答を得ることは行なわないでください**
  - 上記のような活動を井尻が発見した場合は，しかるべき処理をとります
  - 分からない部分がある場合は『どこがどう分からないかを自分の言葉で明確に説明し』他者から知識を受け取ってください
- **プログラミングが不得意な方：**課題を読むと難しく感じるかもしれませんが，各課題のひな形にヒントを多く記載してあります．まずはそこを読んでみてください．
- **プログラミング得意な方:** それなりに歯応えのあるものを用意しようと思ってはいるのですが，まだ簡単なものも多いです．簡単すぎたらごめん。

10/09分 pythonの基本と画像の入出力 課題1~5

課題1. 標準出力 - 雛形 exer1.py

コマンドライン引数から2つの整数を受け取り，その積の回数だけ『hello, world』と標準出力に表示するプログラムを作成せよ

雛形(exer1.py)に途中まで作製したコードがあるので参考にすること

- 実行コマンドと実行例は以下の通り

```
$ python  exer1.py  1 4
hello, world
hello, world
hello, world
hello, world
```

```
$ python  exer1.py  2 3
hello, world
hello, world
hello, world
hello, world
hello, world
hello, world
```

※自動採点の都合上，関係ないものは標準出力に出さないで下さい

課題2. 標準出力 - 雛形 exer2.py

***1*から*N*までの整数を順番に標準出力に表示するプログラムを作成せよ．ただし，以下の仕様を満たすこと.**

- 表示する数字が4の倍数の時には ”hoge” と表示する
- 表示する数字が5の倍数の時には ”fuga” と表示する
- 表示する数字が，4と5，両方の倍数の時には ”hogefuga”と表示する
- 最大の数 *N* は，コマンドライン引数より与える

実行コマンドは以下の通り.

```
$ python  exer2.py  20
```

実行例は，後述

課題3. ファイル入力と配列 - 雛形 exer3.py

数値データをファイルから読み込み，その最大値・最小値をファイルに出力するプログラムを作成せよ

- 入力ファイル名，出力ファイル名は，コマンドライン引数より与える
- ファイルには１行に１つ数値が記載されている
- 出力ファイルの一行目に，最大値と最小値を記載する.
- 最大値と最小値の間には，スペースを一つだけ指定する

実行コマンドは以下の通り.（file_in.txt, file_out.txt は，適当なファイル名）

```
$ python  exer3.py  file_in.txt  file_out.txt
```

実行例は，後述.

ファイル入出力のやり方は，雛形に書いてあるので，参考にしてください

課題4. 画像のグレースケール化 - 雛形 exer4.py

画像データを読み込み，グレースケール化して保存せよ

- 入力ファイル名，出力ファイル名は，コマンドライン引数より与えられることとする
- グレースケール値は，赤・緑・青チャンネルの平均を利用することとする

  $I = (r+g+b)/3$

実行コマンドは以下の通り．(file_in.txt, file_out.txt は，適当なファイル名)

```
$ python   exer4.py   file_in.png  file_out.png
```

実行例は，後述．

画像の入出力・画素値へのアクセス方法は雛形に書いてあるので，参考にしてください

課題5. 画像の2値化 - 雛形 exer5.py

画像データを読み込み，グレースケール化した後，与えられた閾値により二値化し，その画像を保存せよ

- 入力ファイル名，出力ファイル名，閾値は，コマンドライン引数により与えられるとする
- 画素値が閾値以上のなら，その画素値を255にする
- 画素値が閾値より小さいなら，その画素値を 0 とする

実行コマンドは以下の通り．(file_in.txt/file_out.txt は適当なファイル名，200は閾値)

```
$ python   exer5.py   file_in.png  200  file_out.png
```

実行例は，後述．

画像の入出力・画素値へのアクセス方法は雛形に書いてあるので，参考にしてください

実行例(テストに利用してください)

課題2

```
$ python   exer2.py   20
1
2
3
hoge
fuga
6
7
hoge
9
fuga
11
hoge
13
14
fuga
hoge
17
18
19
hogefuga
```

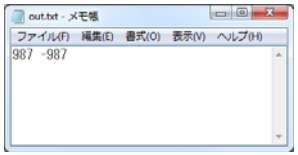
課題3

basicフォルダ内のexer2test.txt に対して実行すると以下のファイルが出力されます

実行コマンド

```
$ python   exer3.py   basic¥exer2.txt basic¥exer2out.txt
```

出力



実行例

課題4

basicフォルダ内のimg.pngに対して実行すると，同一フォルダ内のimg_gray.pngが出力されます

実行コマンド

```
$ python   exer4.py   basic¥img.png basic¥img_gray.png
```

basic¥img.png
入力

basic¥img_gray.png
出力

課題5

basicフォルダ内のimg.pngに対して閾値を150として実行すると，同一フォルダ内のimg_bin.pngが出力されます

実行コマンド

```
$ python   exer4.py   basic¥img.png 150 basic¥img_bin.png
```

basic¥img.png
入力

basic¥img_bin.png
出力

11/13分 フィルタ処理
課題6~9

- 作成中

11/20 フィルタ処理/ハーフトーン処理
課題10~13

- 作成中

11/27 周波数フィルタ
課題14~16

- 作成中

11/27 周波数フィルタ
課題14~16

- 作成中