デジタルメディア処理1

担当: 井尻 敬

デジタルメディア処理1、2017（後期）

09/26　イントロダクション1：デジタル画像とは，量子化と標本化，Dynamic Range
10/03　イントロダクション2：デジタルカメラ，人間の視覚，表色系
10/10　フィルタ処理1：トーンカーブ，線形フィルタ
10/17　フィルタ処理2：非線形フィルタ，ハーフトーニング
10/24　フィルタ処理3：離散フーリエ変換と周波数フィルタリング
11/07　前半のまとめと中間試験
11/14　画像処理演習：python入門　（演習室）
11/21　画像処理演習：フィルタ処理 (演習室)
11/28　画像処理演習：フィルタ処理 (演習室)
12/05　画像処理演習：フィルタ処理 (演習室)
12/12　画像の幾何変換１：アファイン変換
12/19　画像の幾何変換２：画像の補間
01/16　画像復元：ConvolutionとDe-convolution（変更する可能性有り）
01/23　後半のまとめと期末試験

演習 11/28

締め切り: 12/08 23:59

提出方法: 共有フォルダに『**dm1excer**』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．

課題雛形: 準備中

入出力　: 課題では入力画像を受け取り，出力画像を保存するプログラムを作る．入力画像と出力画像のファイル名は以下の通りコマンドライン入力より与えよ

```
$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点扱いとなります．

注意 ： 今回は計算速度を重視しませんが，512x512程度の画像に対して20秒以上の計算時間がかかるものは0点とします．

課題5. モザイク画像作成（exer5.py）

カラー画像を読み込み，モザイク画像を作成するプログラムを作成せよ

- ファイル名はexer5.pyとし，ファイル名・モザイクサイズ*N*を以下の通りコマンドライン引数として取得せよ

```
$python   exer5.py   fname_in.png  r  fname_out.png
```

- モザイク画像生成の際，画像を*N* x *N*のブロックに分割し，各ブロックをその平均画素値で塗るものとする．
- 画像の右端・下端に割り切れないブロック（サイズが*N*に満たない領域）が発生する場合も，その領域を平均画素値で塗ること．

課題6. ハーフトーン – ティザ法（exer6.py）

画像を読み込み，グレースケール画像に変換後，ティザ法により濃淡を維持した二値画像を作成・保存せよ

- ファイル名は excer6.py とし，実行コマンドは以下の通り

```
$python  exer6.py  fname_in.png  fname_out.png
```

- 出力画像は，白255, 黒0，の2値画像とする
- ブロックサイズは４とし，右図のティザパターンを利用すること
- 画像の幅・高さが4の倍数でない場合，画像の右端・下端に余る領域が発生する．この領域は計算せず0を代入するか，そのままティザパターンを重ね合わせて計算すること

0	8	2	10
12	4	14	6
3	11	1	9
15	7	13	5

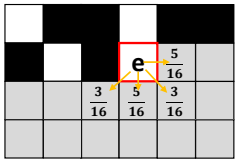
ティザパターン

課題7. ハーフトーン – 誤差拡散法（exer7.py）

画像を読み込み，グレースケール画像に変換後，誤差拡散法により濃淡を維持した二値画像を作成・保存せよ

- ファイル名は excer7.py とし，実行コマンドは以下の通り

```
$python  exer7.py  fname_in.png  fname_out.png
```



誤差拡散する隣接画素

- 出力画像は，白255, 黒0，の2値画像とする
- 拡散する誤差の割合は右図の通りとする
- 右端の画素を計算する際，その右隣の画素が存在しないため右へ誤差を拡散できない．この場合は，単純に右隣への誤差拡散を省略せよ．拡散係数を変化させるなどは行わなくてよい．
- 下端の画素の誤差拡散についても同様．

課題8. メディアンフィルタ（exer8.py）

画像を読み込み，グレースケール画像に変換後，メディアンフィルタを掛けた画像を保存するプログラムを作成せよ

- ファイル名は excer8.py とし，実行コマンドは以下の通り

```
$python  exer8.py  fname_in.png  fname_out.png
```

-1	0	1
-2	0	2
-1	0	1

- np.arrayには，平均・分散・中央値を求める関数があるので活用すること
- フィルタサイズは5x5とする
- 画像周囲の2画素分は，計算せず0を入れておくこと
- 今回はプログラミング練習が目的なので，次のフィルタ関数『cv2.medianBlur』は利用しないこと