CSE464
DIGITAL IMAGE PROCESSING
HW3
REPORT

HAKKI ERDEM DUMAN
151044005

JpegCompression Class

This class has all of functions to compress the given image. Functions and their functionalities are like this:

```
def __init__(self, img):
```

Constructor of class. This function gets input image as parameter and makes is a data field.

```
def colorSpaceConversion(self):
```

This function changes input image's color space from RGB to YcrCb.

```
def preprocess(self):
```

This function uses input image and subtract 128 from each of its pixels.

```
def __padding(self):
```

Since our block is 8x8, we have to complete the image resolution to be divisible by eight. This private function does it. (It is called from preprocess function)

```
def dctAndQuantization(self):
```

This function does DCT and Quantization at the same time.

```
def __dAndQHelper(self, cur_h, cur_w):
```

This is a helper function of dctAndQuantization. "cur_h" and "cur_w" parameters are the current coordinates that are divisible by eight.

```
def inverseDctAndQ(self):
```

This function does the exact opposite of dctAndQuantization function.

```
def __inverseDctAndQHelper(self,cur_h, cur_w):
```
This is a helper function of inverseDctAndQ. “cur_h” and “cur_w” parameters are the current coordinates that are divisible by eight.

```
def postprocess(self):
```
In this function, image is cropped and restored to the first state, which is before padding. And then 128 value is added to all of the pixels.

```
def meanSquareError(self):
```
This function calculates the mean square error.

```
def printImg(self):
```
This function saves the output image as “out.jpg”.

USAGE

You can change “img_path” variable in the “main.py” file. This variable selects the image that will be compressed.

Run “main.py” with Python 3 to see the result.