**Graphics 2 Matlab exercise classes**

**Exercise 6: Colour**

Outline Matlab code for this exercise is in `ex6_colour.m`

Read and display the image BU.jpg in grey using the following code

```
im=imread('BUtower.jpg');
imshow(im);
% Extract the individual RGB planes
imR=im(:,:,1);
imG=im(:,:,2);
imB=im(:,:,3);
```

Swap the following colour planes:

1. R with G
2. G with B

Start from the original RGB image and create a grey-level (monochrome) image by combining the RGB planes:

```
imGrey=0.4*imR+0.4*imG+0.2*imB;
imshow(imGrey);
```

Define colour maps specified in (1)-(3) below and re-colour the `imGrey` image. A sample code for a 128-long colour map specifying shades of grey and uploading the colour map is given below. Code in `ex6_colour.m` is different (it uses a parametric equation).

`C` is the colour map. Code `colormap(C)` loads the colour map and as a result whatever figure is currently displayed is re-coloured according to `C`.

```
% Define 64-long grey-scale colour map.
% A figure must be already displayed when calling this function
M=64;
C=zeros(M,3);
for i=1:M
    v=(i-1)/(M-1);
    C(i,:)=[v v v];
end
colormap(C);
```

Display a sphere (code provided in `ex6_colour.m`). Define the following colour maps to colour the sphere:

1. 64-long monochrome (grey-scale) map
2. 64-long colour map specifying rainbow colours.
3. 32-long colour map specifying red hue of decreasing saturation.
4. 256-long colour map specifying red hue of increasing brightness.
5. 128-long colour map giving the effect of the ‘film negative’ in the shades of grey (i.e. black becomes white, white becomes black, dark colours become bright and vice versa).

For each of the colour maps display the colour mapping plots using Matlab function `rgbplot(C)`.