Code

```
im = imread('colorfulrocks2.jpg');  % Read an image
figure,imshow(im);                  % Plot an image

%% Changing RGB to Gray
imgray = rgb2gray(im);              % Convert image to gray of scale 0-1
imgray = im2double(imgray);         % Changes the unit8 to double
figure, imshow(imgray);

%% Thresholding
level = 0.72;
bw = im2bw(imgray, level);
  figure,
imshow(bw);
saveas(gcf,'rocksimage.png');
%% Count Gray Objects
imRed = im(:,:,1)>=100 & im(:,:,1)<=170 ;
imGreen = im(:,:,2)>=90 & im(:,:,2)<=170;
imBlue = im(:,:,3)>= 63 & im(:,:,3)<=160;
figure, imshow(imBlue);
bwcomp = imcomplement(imBlue);
figure, imshow(bwcomp);
se=strel('disk',4);
ao=imopen(bwcomp,se);
ac=imclose(ao,se);              %Removing noise.
figure,imshow(ac);              %Got Gray rocks.
[L,num]=(bwlabel(~ac,4))        %Number of objects.
%% Area of gray rocks
bwcomp1 = imcomplement(ac);
figure,imshow(bwcomp1);
foregroundArea = sum(ac(:));     %Area of all gray rocks.
cc = bwconncomp(bwcomp1,4);      %Area of individual gray rocks
labeled =(labelmatrix(cc))
%%
if exist('final.mat','file') ~=2
    save('final.mat','bwcomp1','cc')
else
    load('final.mat')
end
%% Red star in gray rocks
pos   = [104 116;52 259;190 219;244 159];
color = {'red','red','red','red'};
RGB = insertMarker(im,pos,'*','color',color,'size',3);
figure,imshow(RGB)
```

Explanation

Q4) Find Area.

For Area:

- Converted image to RGB scale and took Blue scale image.
- Complemented Blue scale image.
- Removed the noise.
- Calculated number of gray rocks.
- Complemented that image(ac).
- Used *bwconncomp* function to find area of individual rocks.

Q5) Estimate the center.

- Using *Datatips* calculated the dimensions of gray rocks and estimated their centroid.

Images

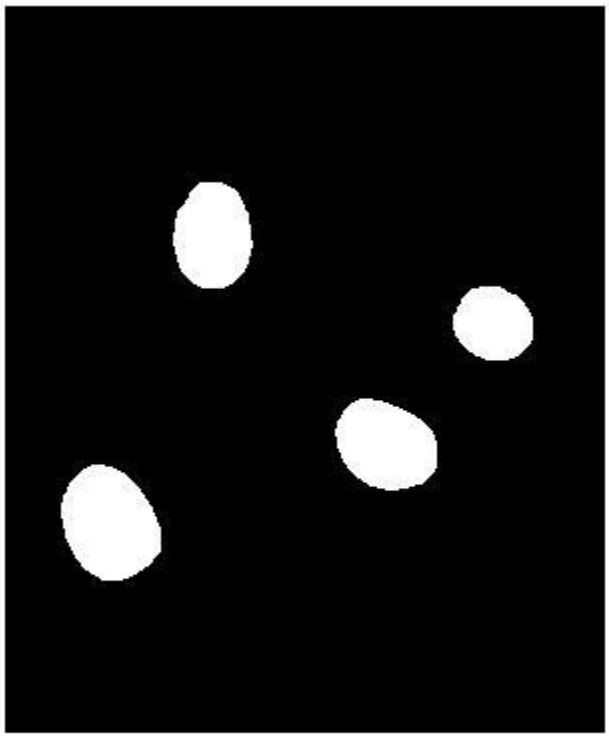

Figure 1: Gray Rocks (Complemented)

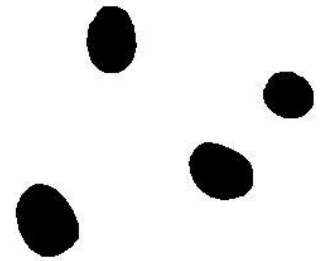

Figure 2: Gray rocks

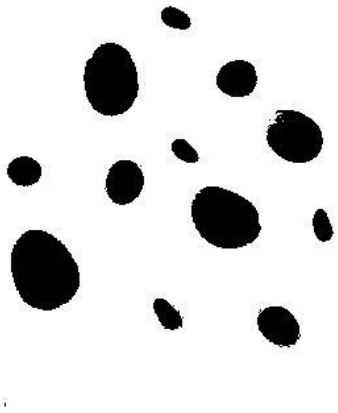

Figure 3: Segmentation

Figure 4: Star in Gray rocks