Q.1 CODE :

Filled_rectangle.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Filled_rectangle.cpp > ...

```cpp
#include<iostream>

#include<opencv2/highgui.hpp>
#include<opencv2/imgcodecs.hpp>
#include<opencv2/imgproc.hpp>

using namespace std;
using namespace cv;

int main(){
    Mat img;
    img=imread("train.jpeg");
    imshow("Original Image", img);
    int wid=img.size().width;
    int height=img.size().height;
    int xstart, ystart, len, hgt, row, col, channel, xfinal, yfinal;
    cout<<"\n(1/4) Enter x and y coordiantes of top-left corner of rectangle : \n";
    cin>>xstart>>ystart;
    cout<<"\n(2/4) Enter length of rectangle : \n";
    cin>>len;
    cout<<"\n(3/4) Enter height of rectangle : \n";
    cin>>hgt;
    cout<<"\n(4/4) Enter code for color for rectangle : 0->blue; 1->green; 2->red :\n";
    cin>>channel;
    xfinal=xstart+len;
    yfinal=ystart+hgt;
    for(row=xstart; row<=xfinal; row++){
        for(col=ystart; col<=yfinal; col++){
            if(channel==0){
                img.at<Vec3b>(col, row)[0] = 255;
                img.at<Vec3b>(col, row)[1] = 0;
```

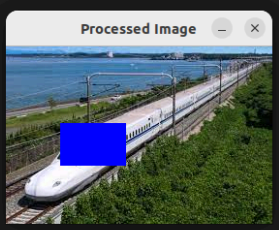
Filled_rectangle.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Filled_rectangle.cpp > ...

```cpp
int main(){
    cin>>channel;
    xfinal=xstart+len;
    yfinal=ystart+hgt;
    for(row=xstart; row<=xfinal; row++){
        for(col=ystart; col<=yfinal; col++){
            if(channel==0){
                img.at<Vec3b>(col, row)[0] = 255;
                img.at<Vec3b>(col, row)[1] = 0;
                img.at<Vec3b>(col, row)[2] = 0;
            }
            else if(channel==1){
                img.at<Vec3b>(col, row)[0] = 0;
                img.at<Vec3b>(col, row)[1] = 255;
                img.at<Vec3b>(col, row)[2] = 0;
            }
            else if(channel==2){
                img.at<Vec3b>(col, row)[0] = 0;
                img.at<Vec3b>(col, row)[1] = 0;
                img.at<Vec3b>(col, row)[2] = 255;
            }
            else{
                cout<<"\nInvalid digit entered. Please enter one from given options.\n";
            }
        }
    }
    imshow("Processed Image", img);
    waitKey(0);
    return 0;
}
```

Q.1 OUTPUT :

```
PROBLEMS 8  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ make SRC=./Filled_rectangle.cpp
Building...
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ ./assignment_2

(1/4) Enter x and y coordiantes of top-left corner of rectangle :
56      79

(2/4) Enter length of rectangle :
67

(3/4) Enter height of rectangle :
43

(4/4) Enter code for color for rectangle : 0->blue; 1->green; 2->red :
0
```



Q.2 CODE :

Filled_Circle.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Filled_Circle.cpp > ...

```cpp
#include <iostream>
#include<cmath>
#include <opencv2/highgui.hpp>
#include <opencv2/imgcodecs.hpp>
#include <opencv2/imgproc.hpp>

using namespace std;
using namespace cv;

int main(){
    Mat img;
    img=imread("./Images/train.jpeg");
    imshow("Original Image", img);
    //waitKey(0);
    int xcen, ycen, rad, row, col, X, Y, R, channel, xnew, ynew;
    cout<<"\n****Information : ****";
    cout<<"\nWidth of image : "<<img.size().width;
    cout<<"\nHeight of image : "<<img.size().height;
    cout<<"\n(1/3) Enter x and y coordiantes of centre of circle : \n";
    cin>>xcen>>ycen;
    cout<<"\n(2/3) Enter radius of circle : \n";
    cin>>rad;
    cout<<"\n(3/3) Enter code for color for circle : 0->blue; 1->green; 2->red :\n";
    cin>>channel;
    R=rad*rad;
    for(row=0; row<=img.size().width; row++){
        for(col=0; col<=img.size().height; col++){
            xnew=row;
            ynew=col;
            X=(xnew-xcen)*(xnew-xcen);
            Y=(ynew-ycen)*(ynew-ycen);
            if((X+Y)<=R){
```

Filled_Circle.cpp 4, U

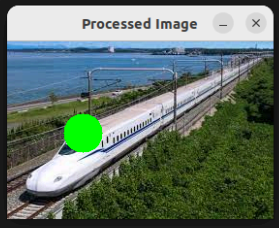
SRA_Pixels_Assignment_2 > Tasks > Filled_Circle.cpp > ...

```
10  int main(){
28              xnew=row;
29              ynew=col;
30              X=(xnew-xcen)*(xnew-xcen);
31              Y=(ynew-ycen)*(ynew-ycen);
32              if((X+Y)<=R){
33                  if(channel==0){
34                      img.at<Vec3b>(ynew,xnew)[0] = 255;
35                      img.at<Vec3b>(ynew,xnew)[1] = 0;
36                      img.at<Vec3b>(ynew,xnew)[2] = 0;
37                  }
38                  else if(channel==1){
39                      img.at<Vec3b>(ynew,xnew)[0] = 0;
40                      img.at<Vec3b>(ynew,xnew)[1] = 255;
41                      img.at<Vec3b>(ynew,xnew)[2] = 0;
42                  }
43                  else if(channel==2){
44                      img.at<Vec3b>(ynew,xnew)[0] = 0;
45                      img.at<Vec3b>(ynew,xnew)[1] = 0;
46                      img.at<Vec3b>(ynew,xnew)[2] = 255;
47                  }
48                  else{
49                      cout<<"\nInvalid digit entered. Please enter one from given options.\n";
50                  }
51              }
52          }
53      }
54      imshow("Processed Image", img);
55      waitKey(0);
56      return 0;
57  }
```

Q.2 OUTPUT :

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ make SRC=./Filled_Circle.cpp
Building...
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ ./assignment_2

(1/3) Enter x and y coordiantes of centre of circle :
78      94

(2/3) Enter radius of circle :
20

(3/3) Enter code for color for circle : 0->blue; 1->green; 2->red :
1
```



Q.3 CODE :

Hollow_Rectangle.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Hollow_Rectangle.cpp > ...

```cpp
#include <iostream>

#include <opencv2/highgui.hpp>
#include <opencv2/imgcodecs.hpp>
#include <opencv2/imgproc.hpp>

using namespace std;
using namespace cv;

int main(){
    Mat img, img1;
    img1=imread("train.jpeg");
    img=imread("train.jpeg");
    imshow("Original Image", img);
    // waitKey(0);
    int xstart, ystart, len, hgt, row, col, channel, xfinal, yfinal, wt;
    cout<<"\n****Information : ****";
    cout<<"\nWidth of image : "<<img1.size().width;
    cout<<"\nHeight of image : "<<img1.size().height;
    cout<<"\n(1/5) Enter x and y coordiantes of top-left corner of rectangle : \n";
    cin>>xstart>>ystart;
    cout<<"\n(2/5) Enter length of rectangle : \n";
    cin>>len;
    cout<<"\n(3/5) Enter height of rectangle : \n";
    cin>>hgt;
    cout<<"\n(4/5) Enter thickness of border of rectangle : \n";
    cin>>wt;
    cout<<"\n(5/5) Enter code for color for rectangle : 0->blue; 1->green; 2->red :\n";
    cin>>channel;
    xfinal=xstart+len;
    yfinal=ystart+hgt;
```

Hollow_Rectangle.cpp 4, U

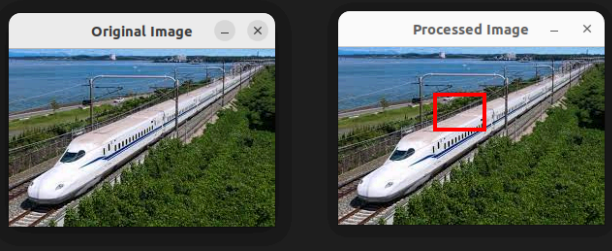
SRA_Pixels_Assignment_2 > Tasks > Hollow_Rectangle.cpp > ...

```cpp
int main(){
    xfinal=xstart+len;
    yfinal=ystart+hgt;
    for(row=xstart; row<=xfinal; row++){
        for(col=ystart; col<=yfinal; col++){
            if(channel==0){
                img.at<Vec3b>(col, row)[0] = 255;
                img.at<Vec3b>(col, row)[1] = 0;
                img.at<Vec3b>(col, row)[2] = 0;
            }
            else if(channel==1){
                img.at<Vec3b>(col, row)[0] = 0;
                img.at<Vec3b>(col, row)[1] = 255;
                img.at<Vec3b>(col, row)[2] = 0;
            }
            else if(channel==2){
                img.at<Vec3b>(col, row)[0] = 0;
                img.at<Vec3b>(col, row)[1] = 0;
                img.at<Vec3b>(col, row)[2] = 255;
            }
            else{
                cout<<"\nInvalid digit entered. Please enter one from given options.\n";
            }
        }
    }
    xstart=xstart+wt;
    ystart=ystart+wt;
    xfinal=xfinal-wt;
    yfinal=yfinal-wt;
    len=len-(2*wt);
    hgt=hgt-(2*wt);
```

```
Hollow_Rectangle.cpp 4, U
SRA_Pixels_Assignment_2 > Tasks > Hollow_Rectangle.cpp > ...
int main(){
            else if(channel==2){
                img.at<Vec3b>(col, row)[0] = 0;
                img.at<Vec3b>(col, row)[1] = 0;
                img.at<Vec3b>(col, row)[2] = 255;
            }
            else{
                cout<<"\nInvalid digit entered. Please enter one from given options.\n";
            }
        }
    }
    xstart=xstart+wt;
    ystart=ystart+wt;
    xfinal=xfinal-wt;
    yfinal=yfinal-wt;
    len=len-(2*wt);
    hgt=hgt-(2*wt);
    for(row=xstart; row<=xfinal; row++){
        for(col=ystart; col<=yfinal; col++){
            img.at<Vec3b>(col, row)[0]=img1.at<Vec3b>(col, row)[0];
            img.at<Vec3b>(col, row)[1]=img1.at<Vec3b>(col, row)[1];
            img.at<Vec3b>(col, row)[2]=img1.at<Vec3b>(col, row)[2];
        }
    }
    imshow("Processed Image", img);
    waitKey(0);
    return 0;
}
```

Q.3 OUTPUT :

```
PROBLEMS 4  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ make SRC=./Hollow_Rectangle.cpp
Building...
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ ./assignment_2

(1/5) Enter x and y coordiantes of top-left corner of rectangle :
98      47

(2/5) Enter length of rectangle :
54

(3/5) Enter height of rectangle :
39

(4/5) Enter thickness of border of rectangle :
4

(5/5) Enter code for color for rectangle : 0->blue; 1->green; 2->red :
2
```



Q.4 CODE :

Hollow_Circle.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Hollow_Circle.cpp > ...

```cpp
#include <iostream>
#include<cmath>
#include <opencv2/highgui.hpp>
#include <opencv2/imgcodecs.hpp>
#include <opencv2/imgproc.hpp>

using namespace std;
using namespace cv;

int main(){
    Mat img1, img;
    img1=imread("train.jpeg");
    img=imread("train.jpeg");
    imshow("Original Image", img1);
    int xcen, ycen, rad, row, col, X, Y, R, channel, xnew, ynew, wt;
    cout<<"\n(1/4) Enter x and y coordiantes of centre of circle : \n";
    cin>>xcen>>ycen;
    cout<<"\n(2/4) Enter radius of circle : \n";
    cin>>rad;
    cout<<"\n(3/4) Enter thickness of border of circle : \n";
    cin>>wt;
    cout<<"\n(4/4) Enter code for color for circle : 0->blue; 1->green; 2->red :\n";
    cin>>channel;
    R=rad*rad;
    for(row=0; row<=img.size().width; row++){
        for(col=0; col<=img.size().height; col++){
            xnew=row;
            ynew=col;
            X=(xnew-xcen)*(xnew-xcen);
            Y=(ynew-ycen)*(ynew-ycen);
            if((X+Y)<=R){
```

Hollow_Circle.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Hollow_Circle.cpp > ...

```cpp
int main(){
                if(channel==0){
                    img.at<Vec3b>(ynew,xnew)[0] = 255;
                    img.at<Vec3b>(ynew,xnew)[1] = 0;
                    img.at<Vec3b>(ynew,xnew)[2] = 0;
                }
                else if(channel==1){
                    img.at<Vec3b>(ynew,xnew)[0] = 0;
                    img.at<Vec3b>(ynew,xnew)[1] = 255;
                    img.at<Vec3b>(ynew,xnew)[2] = 0;
                }
                else if(channel==2){
                    img.at<Vec3b>(ynew,xnew)[0] = 0;
                    img.at<Vec3b>(ynew,xnew)[1] = 0;
                    img.at<Vec3b>(ynew,xnew)[2] = 255;
                }
                else{
                    cout<<"\nInvalid digit entered. Please enter one from given options.\n";
                }
            }
        }
    }
    R=(rad-wt)*(rad-wt);
    for(row=0; row<=274; row++){
        for(col=0; col<=182; col++){
            xnew=row;
            ynew=col;
            X=(xnew-xcen)*(xnew-xcen);
            Y=(ynew-ycen)*(ynew-ycen);
            if((X+Y)<=R){
                img.at<Vec3b>(ynew,xnew)[0] = img1.at<Vec3b>(ynew,xnew)[0];
```

Hollow_Circle.cpp 4, U

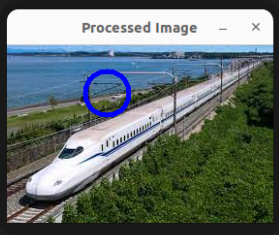
SRA_Pixels_Assignment_2 > Tasks > Hollow_Circle.cpp > ...

```cpp
int main(){
                    else{
                        cout<<"\nInvalid digit entered. Please enter one from given options.\n";
                    }
                }
            }
        }
        R=(rad-wt)*(rad-wt);
        for(row=0; row<=274; row++){
            for(col=0; col<=182; col++){
                xnew=row;
                ynew=col;
                X=(xnew-xcen)*(xnew-xcen);
                Y=(ynew-ycen)*(ynew-ycen);
                if((X+Y)<=R){
                    img.at<Vec3b>(ynew,xnew)[0] = img1.at<Vec3b>(ynew,xnew)[0];
                    img.at<Vec3b>(ynew,xnew)[1] = img1.at<Vec3b>(ynew,xnew)[1];
                    img.at<Vec3b>(ynew,xnew)[2] = img1.at<Vec3b>(ynew,xnew)[2];
                }
            }
        }
        imshow("Processed Image", img);
        waitKey(0);
        return 0;
}
```

Q.4 OUTPUT :

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ make SRC=./Hollow_Circle.cpp
Building...
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ ./assignment_2

(1/4) Enter x and y coordiantes of centre of circle :
103     49

(2/4) Enter radius of circle :
25

(3/4) Enter thickness of border of circle :
5

(4/4) Enter code for color for circle : 0->blue; 1->green; 2->red :
0
```



Q.5 CODE :

Rotate_180_deg.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Rotate_180_deg.cpp > ...

```cpp
#include <iostream>
#include <cstdlib>
#include <opencv2/highgui.hpp>
#include <opencv2/imgcodecs.hpp>
#include <opencv2/imgproc.hpp>

using namespace std;
using namespace cv;

int main(){
    Mat orig=imread("train.jpeg");
    imshow("Original Image", orig);
    int maxdim;
    int orig_wid=orig.size().width;
    int orig_hgt=orig.size().height;
    if(orig_wid>orig_hgt){
        maxdim=orig_wid;
    }
    else{
        maxdim=orig_hgt;
    }

    Mat space(maxdim, maxdim, CV_8UC3, Scalar(0, 0, 0));
    Mat space1(maxdim, maxdim, CV_8UC3, Scalar(0, 0, 0));
    Mat space2(maxdim, maxdim, CV_8UC3, Scalar(0, 0, 0));
    Mat space_done(orig_hgt, orig_wid, CV_8UC3, Scalar(0, 0, 0));

    for(int y=0; y<orig_wid; y++){
        for(int x=0; x<orig_hgt; x++){
            space.at<Vec3b>(x, y)=orig.at<Vec3b>(x, y);
        }
    }
```

Rotate_180_deg.cpp 4, U

SRA_Pixels_Assignment_2 > Tasks > Rotate_180_deg.cpp > ...

```cpp
int main(){
    }

    for(int y=0; y<maxdim; y++){
        for(int x=0; x<maxdim; x++){
            space1.at<Vec3b>(y, x)=space.at<Vec3b>(x, y);

        }
    }

    for(int y=0, y1=(maxdim-1); y<maxdim, y1>=0; y++, y1--){
        for(int x=0, x1=(maxdim-1); x<maxdim, x1>=0; x++, x1--){
            space2.at<Vec3b>(y, x)=space1.at<Vec3b>(x1, y1);
        }
    }

    int differ=abs(orig_hgt-orig_wid)-1;

    for(int y=0; y<maxdim; y++){
        for(int t=differ; t<maxdim; t++){
            int t1=t-differ;
            space_done.at<Vec3b>(t1, y)=space2.at<Vec3b>(t, y);
        }
    }

    imshow("Processed Image", space_done);
    waitKey(0);

    return 0;
}
```

Q.5 OUTPUT :

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ make SRC=./Rotate_180_deg.cpp
Building...
prithvi@prithvi-Inspiron-15-3530:~/SRA_Pixels/SRA_Pixels_Assignment_2/Tasks$ ./assignment_2
```