RAJARATA UNIVERSITY OF SRI LANKA
FACULTY OF APPLIED SCIENCES

B.Sc. (General) Degree in Applied Sciences
Second Year - Semester II Examination – October/November 2017

COM 2304 – COMPUTER GRAPHICS & IMAGE PROCESSING

Time: Three (3) hours

Examination Index No: _____

Important Instructions:

- This paper has 4 questions in 13 pages.
- Answer all questions (25 marks each).
- Write your answers in English using the space provided in this question paper.
- Do not tear off any part of this question paper.
- Note that questions appear on both sides of the paper.
- If a page is not printed, please inform the supervisor immediately.

To be completed by the examiners:

	Question numbers				Total Marks
Questions	1	2	3	4	
Marks					

Question 01

i. What is Digital Image Processing? (2 Marks)

ii. Write two examples for Computer Vision Systems? (2 Marks)

iii. Which frequency bands in Electro Magnetic Energy Spectrum are used as the main imaging source for Computer Axial Tomography (CAT), Fluorescence Microscope, Light Microscope and Magnetic Resonance Imaging (MRI)? (2 Marks)

iv. State why the higher frequency bands carry more energy per photon than the low frequency bands in Electro Magnetic Energy Spectrum. (2 Marks)

v. What is the main function of a typical optical sensor? (4 Marks)

vi. Describe the Sampling and Quantization in the context of the digital image formation process. (4 Marks)

vii. How many samples required when forming a gray (8 bits) image of 8.192 Kilobit size? Assume this image has equal samples for width (x) and height(y). (2 Marks)

viii. What is meant by Dynamic Range in a digital image? (2 Marks)

ix. What type of transformation occurs if the image of intensity resolution 8 bits/pixel will change into 1 bit/pixel? (2 Marks)

x. How interpolation could occur when resizing a digital image? (2 Marks)

xi. What is the value of Cyan color in CMY color model if RGB color values are given? (1 Mark)

Question 02

i. Intensity transformation can express as $s= T(r)$ where, transformation T maps a pixel value r in input source image into a pixel value s in output image. Following statements justify a certain intensity transformation method applied on an input source image to enhance it.

- It has been applied identity transformation for the gray values ranging from 0 to 100 in the input source image.
- For all intensity values between 101 and 150 in the input source image are unchanged.
- Remaining intensity values in the gray scale of the input source image have been negatively transformed.

Model the gray level transformation graph between intensities of input source image (r) and output image (s). Clearly state the intensity values of input source image where the changes have been applied due to the aforementioned enhancement method. Assume that the input source image is a gray scale image. (5 Marks)

ii. Find the dynamic range of the intensities of the resulting image obtained after applying the enhancement method mentioned in Question 2 Part (i) and discuss the consequence of the enhancement method based on the visual quality of the resulting image. (3 Marks)

iii. State the suitable image processing technique(s) to operate the following given cases.

(5 Marks)

(a) To remove salt and pepper noise from a gray scale image:	
(b) To detect the vertical and horizontal edges in a gray scale image:	
(c) To remove illumination imbalance of a gray scale image:	
(d) To remove tiny extrusions recorded on the edge of circular shape object within a binary image:	
(e) To fill selected regions in a color image interactively using a specific color:	

iv. Write a short note on segmentation which used by a typical image processing system.

(6 Marks)

v. Apply Morphological Gradient operation for the image depicted in Figure 1 using the structuring element given in Figure 2. Assume black pixels represent off pixels (0) and white pixels represent on pixels (255). Show all intermediate steps of the operation.

(6 Marks)

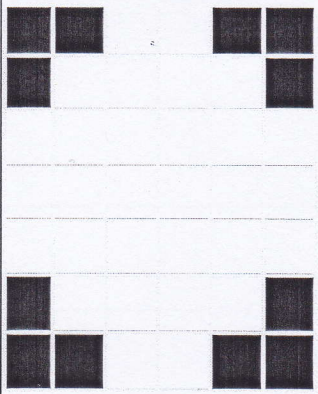


Figure 1 – Selected Image

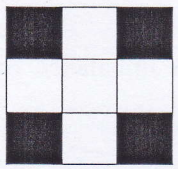


Figure 2- Cross shape structuring element

Question 03

i. State the main purpose of following components in a Computer Graphic System.

(4 Marks)

(a) Display Processor:	
(b) Frame Buffer:	
(c) Video Controller:	
(d) Monitor:	

ii. Briefly explain how "Appeal" important in an animation. (2 Marks)

iii. State why Homogeneous Coordinates have been devised for implementing two dimensional transformations? (2 Marks)

iv. Justify how to perform following 2D transformation using Homogeneous Coordinates. "Scale a polygon modeled at x coordinates 2 and y coordinates 3 by 2 times and shear (deform) it along x axis 2 times." Note that the individual points of this shape is defined using a column vector. (6 Marks)

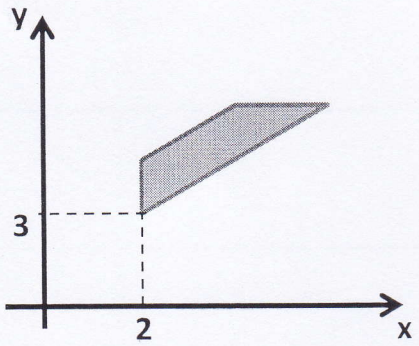


Figure 3 – Polygon Model

v. Briefly explain the difference between following text clipping methods by using appropriate diagrams. (3 Marks)

(a) All-or-none string clipping:

(b) All-or-none character clipping:

(c) Component clipping:

vi. Apply Sutherland-Hodgeman Polygon Clipping Algorithm to clip the polygon (P, Q, R, S) shown in the clipping window (A, B, C, D) of the following Figure 4. Cleary draw the clipping result according to the clipping edge using the space given below. (Clearly indicate the vertex labels.). (8 Marks)

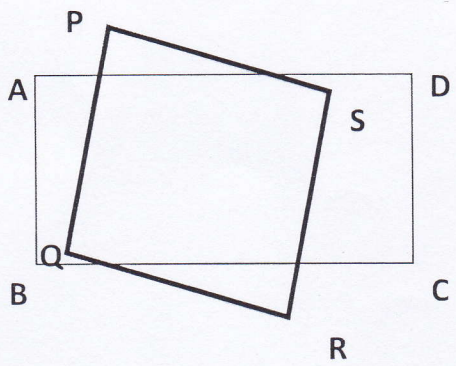
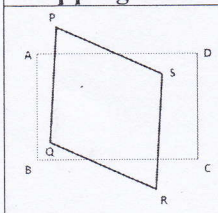
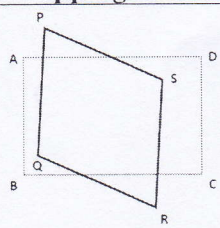
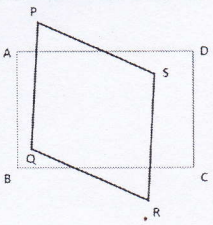
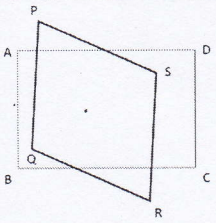


Figure 4 – Polygon and clipping window

Left clipper (AB)		Right Clipper (CD)		Bottom Clipper (BC)		Top Clipper (AD)	
Input	**Output**	**Input**	**Output**	**Input**	**Output**	**Input**	**Output**
[P, Q]							
[Q, R]							
[R,S]							
[S,P]							
Clipping Result 1		**Clipping Result 2**		**Clipping Result 3**		**Clipping Result 4**	
							

Question 04

i. State the suitable OpenCV function names used to implement the following Image processing operations. (3 Marks)

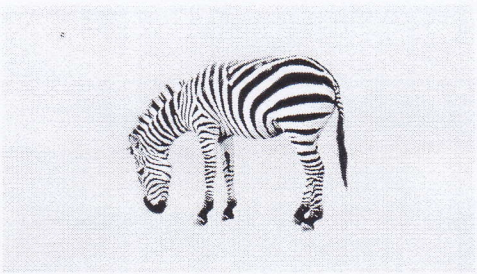
Description	Operation
(a) To create filled pentagon:	
(b) To create a slider:	
(c) To save the image:	
(d) To create a binary image from a gray scale image:	
(e) To keep the size details (width and height):	
(f) To apply median filter:	

ii. Comment on following OpenCV code segments. (5 Marks)

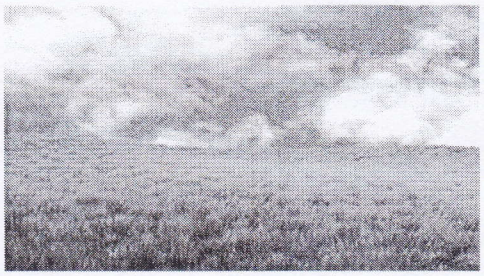
```
equalizeHist( img,result_img);



GaussianBlur( src, gaussian_dst, Size( 3, 3 ), 0, 0 );



Mat element = getStructuringElement(cv::MORPH_CROSS,
                                    cv::Size(3,3)cv::Point(-1,-1) );
```

iii. Write an algorithm to change the background of the source image called source.jpg given in Figure 5(a) using the given background image called land.jpg given in Figure 5(b). Assume that both images are color images and 512×512 of size. Moreover, the background of source image represents a fixed BGR value which is (0,255,145).

(5 Marks)



(a) Source image (source.jpg)



(b) Background image (land.jpg)

Figure 5 – Images for updating background

iv. Fill the blank code segments in the following C++ program which is written to implement binary threshold operation using an 8 bit three channel image called myImage.jpg. In this application threshold value is determined using a track bar called "Value". Moreover, the output is visualized within a window called "Threshold Demo" and is an adjustable according to the size of the image to be loaded into it. (12 Marks)

```
1. #include "opencv2/imgproc/imgproc.hpp"
2. #include "opencv2/highgui/highgui.hpp"
3. #include <stdlib.h>
4. #include <iostream>
5. using namespace cv;
6. using namespace std;
7. int threshold_value = 0;
8. int threshold_type = 1;;
9. int const max_value = 255;
10.int const max_BINARY_value = 255;

11.Mat src, src_gray, dst;

12.char* window_name =

13.char* trackbar_value = "Value";

14.void Threshold_Demo( int, void* );

15.int main()
16.{
17.     src = imread(                    );

18.     if (src.channels()>              )
19.     {
20.         cvtColor(          ,          , CV_BGR2GRAY );
21.     }
22.
23.     namedWindow( window_name,                    );
24.     createTrackbar( trackbar_value,window_name,
                        &threshold_value,max_value,                    );
25.
26.     while(          )
27.     {
28.         int c;
29.         c = waitKey( 20 );
30.         if( (char)c == 27 )
31.             break;
32.     }
33.     return 0;
34.}
35.void Threshold_Demo( int, void* )
36.{
37.     threshold(          , dst, threshold_value,
                                    max_BINARY_value,threshold_type );

38.     imshow(                    ,          );
39.}
```

**** END ****