THIS PAPER IS NOT TO BE REMOVED FROM THE EXAMINATION HALL

UNIVERSITY OF LONDON

CO1112 ZA

BSc, CertHE and Diploma EXAMINATION

CREATIVE COMPUTING and COMBINED DEGREE SCHEME

Creative Computing 1: Image, sound and motion

Friday 17 May 2019: 14.30 – 17.30

Time allowed: 3 hours

DO NOT TURN OVER UNTIL TOLD TO BEGIN

There are **SIX** questions in this paper. Candidates should answer **FOUR** questions. All questions carry equal marks, and full marks can be obtained for complete answers to a total of **FOUR** questions. The marks for each part of a question are indicated at the end of the part in [.] brackets.

Only your first **FOUR** answers, in the order that they appear in your answer book, will be marked.

There are 100 marks available on this paper.

A handheld calculator may be used when answering questions on this paper but it must not be pre-programmed or able to display graphics text or algebraic equations. The make and type of machine must be stated clearly on the front cover of the answer book.

Question 1 General

(a) Give 3 ways in which a computer, with appropriate software and peripherals, can aid creativity in visual arts. For each of the 3 ways that you list, describe what software and peripherals – if any – might be needed, and for each way, describe very clearly the way in which creativity is aided. Also discuss any limitations in any of these. [10]

(b) Discuss the potential use of computers to aid creativity in the field of fashion design. [4]

(c) i. What is the Turing Test? [2]

ii. How could the Turing Test be applied to creativity? Your answer should discuss the strengths and limitations of this. [4]

(d) Kandinsky had an strong interest in the relationship between sound and colour, and there has been debate over whether he may have had the condition of synaesthesia. What is synaesthesia, and what effect might this have had on the development of his art and the work of the Bauhaus? Use examples to support your answer if appropriate. [5]

Question 2 Data, sound and motion

(a) Name the class that *Processing* provides for the manipulation of images. Each instance of this class possesses an array and two `int` variables. Explain what role each of these plays in storing the image. [5]

(b) Considering what you have learned about compression of an audio signal, discuss the tradeoffs between lossy and lossless compression of images. [5]

(c) In the context of 3-D graphics, what is translation and what is rotation? Illustrate your answer with an example. [4]

(d) Is matrix multiplication commutative – that is, does A*B have the same result as B*A for two matrices A and B? Justify your response. [6]

(e) Briefly describe the Theremin device, including stating how pitch and volume are controlled. [5]

Question 3 Shape and structure

(a) What is a fractal? Give an example, either a drawing or a description, of a fractal. [3]

(b) *Processing* provides the ability to perform both recursion and repetition in code.

i. What is the difference between repetition and recursion? [2]

ii. Discuss the work of one artist who uses the concept of recursion in their work and one artist who uses repetition in their work. You may take the term 'artist' to include all forms of art, such as music, dance, theatre performance, *etc.* [8]

(c) What are each of the following Gestalt principles that relate to how images are perceived? For each, illustrate your description with a diagram, and explain what aspect of the diagram gives the effect. [12]

i. reification

ii. multistability

iii. closure

iv. proximity.

Question 4 Motion and 2D Graphics

Study the *Processing* sketch below, then answer the questions that follow.

```
1  float x = 200, y = 200;
2  float dx = 5, dy = 10;
3  float gravity = 5;
4  float ballSize = 50;
5
6  void setup() {
7   size(800,800);
8   fill(0,255,0);
9   frameRate(20);
10  noStroke();
11 }
12
13 void draw() {
14   background(0);
15   translate(x, y);
16   ellipse(0, 0, ballSize, ballSize);
17   if (x > width-ballSize/2 || x < ballSize/2)
18     dx = -dx;
19   if (y > height-ballSize/2)
20     dy = -dy;
21   else
22     dy += gravity;
23    x += dx;
24    y += dy;
25 }
```

(a) Draw a diagram to show the output of this sketch after the first call to `draw()`. State the size, position and colour of any objects drawn, the dimensions of the window, and the colour of the background. [3]

(b) Explain the purpose of lines 19 and 20 of the sketch, and the effect they have on the sketch's behaviour. [3]

(c) Line 17 includes the condition "`x < ballSize/2`". However, line 19 does not include the condition "`y < ballSize/2`". Explain why this is not required in the context of this sketch. (Quantitative calculations are *not* required.) [5]

(d) Give a one-sentence definition of velocity, and a one-sentence definition of acceleration. [2]

(e) Does the ball in this sketch experience acceleration in the `x` direction? Explain your answer, with reference to specific lines in the sketch where necessary. [3]

(f) Write code that will produce a visible trail of the ball's path of movement such that the old parts of the trail gradually fade away to the background colour.

You do not need to rewrite the whole sketch in your answer. Just provide any new lines of code you would add, give a line number to indicate where you would insert the code, and provide an explanation of what each new line does. Also state which existing lines you would delete or move (if any). [5]

(g) Write code that causes the ball to lose five percent of its speed each time it hits the bottom edge of the window.

You do not need to rewrite the whole sketch in your answer. Just provide any new lines of code you would add, or state which existing lines you would delete or modify (if any). Provide an explanation of what each new or modified line does. [4]

Question 5 Generative systems

The *Processing* code shown below implements a simple L-System. Study the code and then answer the questions that follow.

```
1  float  d = 100;
2  float  ang = -PI/4;
3  String state = "F";
4  String F_rule = "F[-F]+F";
5  int    n = 2;
6
7  void setup() {
8    size(800,800);
9    background(255);
10   stroke(0);
11   noLoop();
12   for (int i=0; i < n; i++)
13     state = substitute(state, F_rule);
14 }
15
16 void draw() {
17   translate(width/2, (height*7)/8);
18   rotate(-HALF_PI);
19   for (int i=0; i < state.length(); i++) {
20     turtle(state.charAt(i));
21   }
22   println(state);
23 }
24
25 void turtle(char c) {
26   switch(c) {
27   case 'F': line(0, 0, d, 0); translate(d, 0); break;
28   case '-': rotate(ang);  break;
29   case '+': rotate(-ang); break;
30   case '[': pushMatrix(); break;
31   case ']': popMatrix();  break;
32   }
33 }
34
35 String substitute(String s, String F) {
36   String s2 = new String();
37   for (int i=0; i < s.length(); i++) {
38     if(s.charAt(i)=='F')
39       s2 = s2 + F;
40     else
41       s2 = s2 + s.charAt(i);
42   }
43   return s2;
44 }
```

(a) When describing substitution systems such as this L-System, the terms *initiator* and *generator* are often used. What initiator(s) and generator(s) are used in this code? Give both the variable names and values used. [2]

(b) The `noLoop()` function is called on line 11 of the sketch. What effect does this function have, and why is it appropriate to use it in this sketch? [3]

(c) The functions `pushMatrix()` and `popMatrix()` are often used in *Processing* sketches. Explain what these two functions do, and why they are useful. [4]

(d) Does the code make use of recursion? Justify your answer in detail. [4]

(e) Line 22 calls the `println()` function. Write down the text that this line outputs. [3]

(f) The `turtle()` function takes an individual character from the `state` string and performs certain graphical operations according to what character it is given. What operations are performed when the function is passed an `F` character? [3]

(g) Draw the output of this sketch. Label your drawing with the co-ordinates of the beginning of the first line that is drawn. Also label the drawing with the co-ordinates of the end of the last line that is drawn. [6]

Question 6 3D motion and Sound

Study the *Processing* sketch below, then answer the questions that follow.

```
1  float angle = 0.0;
2
3  void setup() {
4    size(500,500,P3D);
5    frameRate(30);
6    sphereDetail(10);
7    noFill();
8  }
9
10 void draw() {
11   background(255);
12   translate(width/2,height/2,0);
13   rotateY(angle);
14   stroke(255,0,0);
15   sphere(100);          // planet
16   translate(200,0);
17   stroke(0,0,255);
18   sphere(10);           // satellite
19   angle += TWO_PI/(10*frameRate);
20 }
```

(a) Draw a diagram to show the output of this sketch after the first call to `draw()`. State the size, position and colour of any objects drawn, the dimensions of the window, and the colour of the background. [5]

(b) Does the sphere labelled "satellite" in the sketch (line 18) appear to move as the sketch runs? If so, how many seconds does it take for it to return to its initial position? [3]

(c) Explain the effect of the command "`sphereDetail(10)`" on line 8 of the sketch. Name one advantage, and one disadvantage, of using a higher number as the argument to this function. [3]

(d) Would the behaviour of the sketch change if we swapped lines 12 and 13 around, *i.e.* if `rotateY()` came before `translate()`? Explain your answer. [4]

(e) The movement in this sketch could be implemented in an alternative way by controlling the camera position instead of using the `rotateY()` function. State what *Processing* function you would call to set the camera position, and describe the meaning of each of its parameters. [4]

(f) We would like to extend the sketch so that it plays background music from an audio file while the sketch runs.

i. State the command that is required at the top of the sketch to allow it to make use of the *Processing* sound library. [2]

ii. Assume that the desired audio file is named "`planets.wav`" and is located in the sketch's data folder. What code should be added to the sketch to allow it to load and play this file?

You do not need to rewrite the whole sketch in your answer. Just provide any new lines of code you would add, and give a line number to indicate where you would insert each line. [4]

END OF PAPER