Examiners' commentaries 2016–17

CO3355 Advanced graphics and animation – Zone A

General remarks

This year, the general standard of performance on this examination paper fell compared to the previous year. Although there were a number of very strong papers, many candidates failed to achieve a passing mark.

While most candidates could demonstrate some appreciation of the relevant issues in the questions, some struggled to provide the clear and concise explanations and so lost marks. Providing succinct and accurate explanations is worth focusing on in revision and examination practice, and should build upon skills developed in earlier assignments and examinations.

In terms of individual popularity, Question 3 was the most popular with 91 per cent of candidates attempting it, closely followed by Questions 1 and 5 (about 82 per cent). Forty-four per cent of candidates answered Question 4, while Question 2 was the least popular with hardly any candidates answering it.

The average performance was almost uniform across questions, though Question 5 attracted the highest marks and was followed by Questions 1, 4, 3 and 2 respectively.

Comments on specific questions

Question 1

a. This required very basic knowledge of related mathematics. About 50 per cent of candidates provided a good answer. Candidates are advised to make sure they master the underlying mathematics (geometry and linear algebra) to a good degree. Working through the exercises of Chapter 2 in the subject guide is highly beneficial to that end. Any issues should be raised on the VLE discussion board.

b. This matrix represented a scaling by [2, 2, 2] followed by a translation by [1, 1, 1]. The result of the calculation when applied to the given vector was: $[2, 5, 3, 1]^T$.

   The majority of candidates managed to calculate the result correctly. However, most had issues explaining the type of transformations performed. This is dependent on very basic understanding of the transformation matrix elements that indicate S, R, and T respectively. Knowledge of the transformation matrix structure leads easily to the correct interpretation.

c. This section examined the effect of order when performing geometric transformations. Candidates had to assess whether certain propositions were true or false and to justify their answers. Regarding the specific propositions:

   i. was false, as in both cases the object moves.

   ii. was true, as the scaling matrix was the same.

   iii. was false, as scaling when performed after translation will modify the latter and translate the object more.

iv. depended on the scaling factor. If greater than 1 (the object getting larger) then it would be translated more in case b and the proposition would be false. Otherwise the proposition would be true.

Marking was fairly flexible, accommodating some answers that made other assumptions, as long as they were reasonable and well-explained. However, many candidates had problems in providing satisfactory explanations, even where they identified the validity of each proposition correctly.

d. This part asked candidates to compare perspective and parallel projections by identifying an advantage of each and stating how a set of parallel lines would appear in each case. The vast majority of candidates answered this correctly, though a few that confused the two transformations.

e. This part asked candidates to calculate the result of an orthographic projection on the plane z=-1. Only a few candidates managed to provide a correct answer, though it was relatively simple; the resulting vector would be similar to that given, but with its z dimension changed to -1.

Question 2

a. This part examined *layering* in traditional animation, asking candidates to identify the problem it solves and the way it does it. This was essentially bookwork, covered in Chapter 8 of the subject guide.

b. This part tested candidates' understanding of linear interpolation. As a key frame was used for every 10 frames, using a formula such as $P(t) = (1/10)*(t*P(t_A) + (10-t)*P(t_B))$, with t=1 yields the result: [6.7, 7.7, 2.7].

c. This part was about Hermite and Bezier forms for curve representation. Good answers specified that although the curves are essentially equivalent, in Hermite form, instead of specifying four control points, two end points and their tangents are specified.

d. This part was about skeletal animation, asking candidates to describe how bone position is calculated using forward kinematics and to derive the corresponding formula for the given figure. The former was essentially bookwork, while for the latter a formula of the form $R_1(R_2(L_2)+L_1)$ would suffice.

e. This part of the question required application of the law of cosines to solve an instance of the inverse kinematics problem presented in (b). Good answers included verification that the desired position was within reach, and then solved the expression to calculate the cosine of each of the three angles, and then the angles themselves.

Question 3

a. This part required candidates to describe the term *depth cueing*. Depth cueing models the light attenuation as it travels away from its source, according to its distance. An answer explaining this would be sufficient for full marks. However, few candidates managed to provide it.

b. This part asked why Phong shading is more computationally intensive than Gouraud. Gouraud shading computes lighting values per vertex and interpolates them over a polygon. Phong shading on the other hand computes lighting values per pixel, by first interpolating the surface normals. Phong is more computationally expensive since the reflection model must be computed at each pixel instead of at each vertex. This was generally well-answered.

c. This part required explanation of the concept of *ray-tracing*. This was essentially bookwork, covered in section 7.7 of the subject guide. Most candidates provided a satisfactory explanation.

d. This was a simple exercise to test the candidates' understanding of the Phong lighting model. It provided a diagram consisting of a light source and three cameras pointing on a surface, asking which one sees the brightest (i) diffuse and (ii) specular illumination. As diffuse reflection does not depend on the angle of view, it does not change among the cameras. Camera C would see the brightest specular reflection, as the angle from the point to the camera was closest to the angle from the light to the point.

Many candidates were confused by (i) choosing either A or B, while almost all provided the correct answer for (ii). Again, some failed to provide the rationale behind their choice.

e. This part of the question asked for the functionality of the provided shader code together with a description of its main steps. The code is a fragment shader that showed a fragment colour varying with time. It used a "time" variable and a sine function to choose the R colour component and then exchanged the G and B components of the vertex colour. The result was the composite fragment colour. This was generally well-answered. However, some candidates failed to identify that this was a fragment shader, while others did not describe the colour component exchange.

Question 4

a. This part asked candidates about *map entities* and their use in texture mapping. Roughly about 50 per cent of candidates answered this question well. The most common problem was to confuse the notions of map entity and map shape. There were some muddled explanations, and some candidates also failed to give proper examples of map entities.

b. This part was about the *Object Coordinate System* where the coordinates are fixed relative to the object. Most mapping techniques use it to keep the texture in place when the object moves. About 50 per cent of candidates failed to identify this.

c. This part of the question compared the techniques of *texture mapping* and *bump mapping* in terms of the achieved realism in three different situations when lighting is modelled in the scene. In all three conditions, bump mapping helps produce more realistic results. Very few candidates provided clear answers to this one.

d. i. In this part, the code of a vertex shader that performs simple texture mapping was given, and candidates were asked to explain what it does. The code transforms the texture coordinates of each vertex, first rescaling them and taking into account texture inversion along the Y-axis and the non-power-of-two textures, passing the transformed coordinates to the fragment shader. This was mostly well answered, although some answers lacked clarity.

ii. This part asked candidates to consider the corresponding fragment shader, and fill in the missing commands. The answer could be given in one command such as:

```
gl_FragColor = texture2D(texture, vertTexCoord.
st) * vertColor;
```

Only a few candidates provided this correctly.

Question 5

a. This part of the question asked about the notion of varying variables in GPU programs, requiring three examples of their use. Most candidates provided good answers, accompanied with appropriate examples.

b. This part required a drawing to illustrate a situation where Painter's algorithm would fail to be represented correctly. It was important that the drawings clearly demonstrated the issue and most candidates answered this part well.

c. This part asked candidates to identify what a depth buffer is, how it differs from a framebuffer, and to explain how it is used in real-time graphics.

Those candidates who appreciated the notion of a depth buffer had no difficulty in achieving full marks. Some also explained the Z-buffer algorithm in detail. Although this was appreciated, it was not necessary.

d. This part asked candidates to state whether the use of a fragment or a vertex shader would be more appropriate for four different tasks, justifying their answers. A fragment shader is appropriate for edge detection and Phong shading, as both of these operations happen in the pixel/fragment level. A vertex shader is suitable for skinning and displacement mapping that involve transformation of vertices. Only a few candidates managed to provide correct answers with clear justification.