NATIONAL UNIVERSITY OF SINGAPORE

CS3241 — COMPUTER GRAPHICS

(AY2021/2022 SEMESTER 1)

FINAL ASSESSMENT

Time Allowed: **2 Hours**

SOLUTIONS

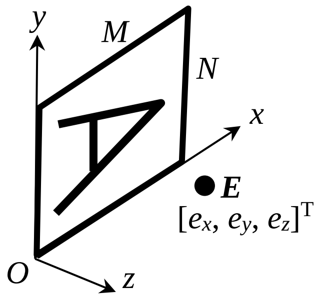
INSTRUCTIONS

1. This assessment contains **22 questions** in **7 sections**.
2. The full score of this assessment is **90 marks**.
3. Answer **all questions**.
4. This is an **open-book** assessment.
5. You are allowed to use an approved **calculator**.
6. **Follow the instructions of your invigilator or the module coordinator to submit your answers**.

Section A [7 marks]

(1) [7 marks]

A rectangle, of size M x N, is located in the **world space** as shown in the diagram below. A viewpoint $\boldsymbol{E}$ is located at world coordinates $[e_x, e_y, e_z]^T$, where $e_z \geq 1$. You are to complete the following program to set up the view transformation and the perspective view volume so that the rectangle appears to cover the entire viewport *exactly*. The rectangle's corner at the world origin O should appear at the bottom-right corner of the viewport (so that the letter "A" appears upright). You should set the near plane exactly midway between the rectangle and the viewpoint, and the far plane exactly one unit behind the rectangle. Use `gluLookAt()` to set the view transformation, and `glFrustum()` to set the view volume.




```
...
double M, N;          // rectangle size.
double ex, ey, ez;    // viewpoint world coordinates.
...
glViewport( 0, 0, vp_width, vp_height );
...
glMatrixMode( GL_PROJECTION );
glLoadIdentity();

// Call glFrustum() to set the view volume:
// glFrustum( ... );

glMatrixMode( GL_MODELVIEW );
glLoadIdentity();

// Call gluLookAt() to set the view transformation:
// gluLookAt( ... );
```

Write the calls to `glFrustum()` and `gluLookAt()` in the following space.

```
glFrustum( (ey - N)/2, ey/2, -ex/2, (M - ex)/2, ez/2, ez + 1 );
...
gluLookAt( ex, ey, ez,   ex, ey, ez - 1,   1, 0, 0 );
```

Section B [15 marks]

(2) [6 marks]

For each of the following statements, indicate whether it is **true** or **false**. Write your answer in the space following each statement. You must write "**T**" for **true** and "**F**" for **false**, without the quotation marks.

(a) Bump mapping works by modifying the z values (depth values) of the fragments according to the given bump map. T/F?

F

(b) Bump mapping can only produce bumps (protrusions) but not dents (depressions). T/F?

F

(c) Bumps produced by bump mapping on the same flat surface cannot occlude each other. T/F?

T

(d) Reflection mapping can be used for scenes that have moving objects. T/F?

T

(e) Reflection mapping is geometrically accurate if the viewpoint is very far away from the reflective object. T/F?

F

(f) Reflection mapping and bump mapping can be combined to produce shiny and bumpy surfaces. T/F?

T

(3) [5 marks]

For each of the following statements, indicate whether it is **true** or **false**. Write your answer in the space following each statement. You must write "**T**" for **true** and "**F**" for **false**, without the quotation marks.

(a) The fragment processing stage can **directly** read the content of the color buffer to use it as a texture image. T/F?

F

(b) After a fragment is processed by the per-fragment operations stage, it can be written to multiple pixel locations in the color buffer. T/F?

F

(c) Z values (depth values) on a polygon are not interpolated when flat shading is turned on. T/F?

F

(d) Z testing (depth testing) on a fragment is performed after texture mapping in the OpenGL rendering pipeline. T/F?

T

(e) Assigning a RGBA color to a fragment, where the A value is 0.5, will automatically make the fragment look translucent. T/F?

F

(4) [4 marks]

Which of the following view volume settings is the most prone to **z-fighting**?

A. `gluPerspective(30, 1, 101, 110);`
B. `gluPerspective(120, 1, 101, 110);`
C. `gluPerspective(30, 1, 111, 120);`
D. `gluPerspective(120, 1, 111, 120);`
E. `gluPerspective(30, 1, 1, 10);`
F. `gluPerspective(120, 1, 1, 10);` **(answer)**

Section C [10 marks]

(5) [3 marks]

Suppose in a mipmap, each texel in a mipmap level is the average of the corresponding 3x3 texels in the previous level, and all these 3x3-texel regions are non-overlapping. Given a texture image of size 2187 x 2187 texels, we want to create a **mipmap** from it. What would be the **number of levels** (including the original texture image) in the mipmap? (Note that $2187 = 3^7$).

A. 1
B. 3
C. 7
D. 8 **(answer)**
E. 9
F. 11
G. 12
H. 13

(6) [4 marks]

Suppose in a mipmap, each texel in a mipmap level is the average of the corresponding 3x3 texels in the previous level, and all these 3x3-texel regions are non-overlapping. Given a **mipmap** constructed from an original texture image of size 2187 x 2187 texels, what would be the **exact mipmap level** to texture map a 3D square that appears in a 200 x 200 pixels region in the viewport? The selected mipmap level can be **non-integer**, and you must round your answer to **2 decimal places**. You may use the formula $\log_a(x) = \log_b(x) / \log_b(a)$.

$\log_3(2187) - \log_3(200) = \log_3(2187/200) \approx 2.18$

(7) [3 marks]

Consider a texture-mapped scene rendered using **mipmapping**. During rendering, suppose the mipmap level selection is always **biased by –2 levels**, how may the rendered result differ from the case when no bias is used?

A. There is no difference in the rendered result.
B. The resulting image will look darker in the textured regions.
C. The resulting image will look brighter in the textured regions.
D. The resulting image may have more aliasing artifacts. **(answer)**
E. The resulting image will look more blurred in the textured regions.
F. The resulting image will look more blocky.

Section D [12 marks]

Given a 400 x 300 (*width* x *height*) gray-scale image ***I*** [0...399, 0...299], its ***modified summed area table*** (MSAT) ***S*** [0...399, 0...299] is defined as follows:

$$\boldsymbol{S}[m,n] = \sum_{i=m}^{399} \sum_{j=n}^{299} \boldsymbol{I}[i,j]$$

for $0 \le m \le 399$ and $0 \le n \le 299$.

(The notation $\boldsymbol{A}[a...b, c...d]$ refers to all elements $\boldsymbol{A}[i, j]$ of the 2D array $\boldsymbol{A}$ where $a \le i \le b$ and $c \le j \le d$.)

(8) [3 marks]

Which element of the MSAT ***S*** has the sum of *all* the pixels' values of image ***I*** ?

A. ***S*** [0, 0] **(answer)**
B. ***S*** [1, 1]
C. ***S*** [0, 299]
D. ***S*** [399, 0]
E. ***S*** [199, 149]
F. ***S*** [398, 298]
G. ***S*** [399, 299]
H. ***S*** [400, 300]

(9) [4 marks]

Given that the pre-image of a fragment is the quadrilateral whose vertices are at (0.2, 0.2), (0.5, 0.3), (0.4, 0.6), and (0.3, 0.6) in the texture space. Which of the following would be the best rectangular region of image ***I*** that we should area sample (provided that the rectangular image region must enclose the pre-image)?

A. ***I*** [0...80, 0...60]
B. ***I*** [0...199, 0...179]
C. ***I*** [80...399, 60...299]
D. ***I*** [60...80, 179...199]
E. ***I*** [80...179, 60...199]
F. ***I*** [60...179, 60...179]
G. ***I*** [80...199, 80...199]
H. ***I*** [80...199, 60...179] **(answer)**

(10) [5 marks]

Which of the following expressions correctly uses the MSAT ***S*** to efficiently compute the **average** of all pixel values for the rectangular image region ***I*** [20...220, 50...150]?

A. (***S*** [220, 150] – ***S*** [19, 150] – ***S*** [220, 49] + ***S*** [19, 49]) / (201 * 101)
B. (***S*** [220, 150] – ***S*** [20, 150] – ***S*** [220, 50] + ***S*** [20, 50]) / (201 * 101)
C. (***S*** [220, 150] + ***S*** [19, 150] + ***S*** [220, 49] – ***S*** [19, 49]) / (201 * 101)
D. (***S*** [220, 150] + ***S*** [20, 150] + ***S*** [220, 50] – ***S*** [20, 50]) / (201 * 101)
E. (***S*** [20, 50] – ***S*** [221, 50] – ***S*** [20, 151] + ***S*** [221, 151]) / (201 * 101) **(answer)**
F. (***S*** [20, 50] + ***S*** [221, 50] + ***S*** [20, 151] – ***S*** [221, 151]) / (201 * 101)
G. (***S*** [20, 50] + ***S*** [220, 50] + ***S*** [20, 150] – ***S*** [220, 150]) / (201 * 101)
H. (***S*** [20, 50] – ***S*** [220, 50] – ***S*** [20, 150] – ***S*** [220, 150]) / (201 * 101)

Section E [8 marks]

(11) [4 marks]

Consider the **ray** (in parametric representation)

$$\boldsymbol{P}(t) = \begin{bmatrix} 100 \\ 80 \\ 60 \end{bmatrix} + t \begin{bmatrix} -20 \\ -16 \\ -12 \end{bmatrix}$$

and the **sphere** ***S*** (in implicit representation)

$$x^2 + y^2 + z^2 - 2^2 = 0.$$

Suppose the first intersection between the ray $\boldsymbol{P}(t)$ and the sphere $\boldsymbol{S}$ is when $t = t_0$. We want to construct a ray $\boldsymbol{R}(s)$ such that the first intersection between the ray $\boldsymbol{R}(s)$ and the **unit sphere** ***U***

$$x^2 + y^2 + z^2 - 1^2 = 0$$

is also when $s = t_0$. Which of the following is the correct $\boldsymbol{R}(s)$?

A. $\boldsymbol{R}(s) = [100\ \ 80\ \ 60]^T + s\,[-10\ \ -8\ \ -6]^T$

B. $\boldsymbol{R}(s) = [100\ \ 80\ \ 60]^T + s\,[-40\ \ -32\ \ -24]^T$

C. $\boldsymbol{R}(s) = [50\ \ 40\ \ 30]^T + s\,[-10\ \ -8\ \ -6]^T$ **(answer)**

D. $\boldsymbol{R}(s) = [200\ \ 160\ \ 120]^T + s\,[-10\ \ -8\ \ -6]^T$

E. $\boldsymbol{R}(s) = [50\ \ 40\ \ 30]^T + s\,[-40\ \ -32\ \ -24]^T$

F. $\boldsymbol{R}(s) = [200\ \ 160\ \ 120]^T + s\,[-40\ \ -32\ \ -24]^T$

G. $\boldsymbol{R}(s) = [50\ \ 40\ \ 30]^T + s\,[-20\ \ -16\ \ -12]^T$

H. $\boldsymbol{R}(s) = [200\ \ 160\ \ 120]^T + s\,[-20\ \ -16\ \ -12]^T$

(12) [4 marks]

The first intersection between a **ray** and the **sphere** ***S*** (in implicit representation)

$$(x-8)^2 + (y-5)^2 + (z-10)^2 - R^2 = 0$$

is at the location (x_0, y_0, z_0). Which of the following is the correct **normalized surface normal vector** at the intersection point?

A. $[\,(x_0-8)/R,\ \ (y_0-5)/R,\ \ (z_0-10)/R\,]^T$ **(answer)**

B. $[\,x_0/R,\ \ y_0/R,\ \ z_0/R\,]^T$

C. $[\,8/R,\ \ 5/R,\ \ 10/R\,]^T$

D. $[\,(x_0-8)/R^2,\ \ (y_0-5)/R^2,\ \ (z_0-10)/R^2\,]^T$

E. $[\,x_0/R^2,\ \ y_0/R^2,\ \ z_0/R^2\,]^T$

F. $[\,8/R^2,\ \ 5/R^2,\ \ 10/R^2\,]^T$

G. $[\,(x_0-8),\ \ (y_0-5),\ \ (z_0-10)\,]^T$

H. $[\,x_0,\ \ y_0,\ \ z_0\,]^T$

I. $[\,8,\ \ 5,\ \ 10\,]^T$

Section F [12 marks]

Suppose there is an **opaque sphere** in an **enclosed** environment. All the other surfaces of the environment are **opaque** and have materials that have both **diffuse and specular** components (i.e. k_d, k_r, and k_{rg} are all greater than 0). However, the sphere's material has only **diffuse** component (i.e. $k_d > 0$, $k_r = 0$ and $k_{rg} = 0$). There are **three point light sources** in the scene.

We want to render a **400x300 pixels image** of the scene using **Whitted Ray Tracing**, with **one level of recursion**.

Assume the camera is within the environment but outside the sphere, and the **sphere occupies 20,000 pixels** (i.e. number of primary rays that hit the sphere directly) in the rendered image.

In addition, at any surface point where lighting computation is to be performed, a **shadow ray** is always shot towards each light source even when $N \bullet L \leq 0$ (where N is the normal vector at the surface point and L is the vector towards the light source).

(13) [2 marks]

What is the total number of **primary rays** shot?

- **A.** 20,000
- **B.** 60,000
- **C.** 100,000
- **D.** 120,000 **(answer)**
- **E.** 140,000
- **F.** 300,000
- **G.** 360,000
- **H.** 420,000
- **I.** None of the other options is the correct answer.

(14) [3 marks]

What is the total number of **shadow rays** shot from those points intersected by the primary rays?

- **A.** 0
- **B.** 60,000
- **C.** 300,000
- **D.** 360,000 **(answer)**
- **E.** 420,000
- **F.** More than 420,000 and less than or equal to 600,000
- **G.** More than 600,000 and less than or equal to 800,000
- **H.** More than 800,000 and less than or equal to 1,200,000
- **I.** None of the other options is the correct answer.

(15) [3 marks]

How many **first-level** secondary **reflection rays** are spawned? First-level secondary reflection rays are spawned from surface points hit by the primary rays.

A. 0
B. 20,000
C. 60,000
D. 80,000
E. 100,000 **(answer)**
F. 120,000
G. 300,000
H. 360,000
I. None of the other options is the correct answer.

(16) [4 marks]

How many **rays** are shot **altogether**? This includes all the primary rays, reflection rays and shadow rays.

A. 100,000
B. 120,000
C. 460,000
D. 480,000
E. 880,000 **(answer)**
F. 900,000
G. 940,000
H. 980,000
I. None of the other options is the correct answer.

Section G [26 marks]

(17) [3 marks]

Let **s**(u), for $0 \leq u \leq 1$, be the parametric representation of an unit-radius semicircle arc, which starts from the point (1, 0), continues counter-clockwise, and ends at the point (–1, 0). Which of the following is the correct parametric representation of the semicircle arc?

A. $\mathbf{s}(u) = [\ \cos \pi u\ ,\ \sin \pi u\ ]^T$ **(answer)**
B. $\mathbf{s}(u) = [\ \sin \pi u\ ,\ \cos \pi u\ ]^T$
C. $\mathbf{s}(u) = [\ \cos 2\pi u\ ,\ \sin 2\pi u\ ]^T$
D. $\mathbf{s}(u) = [\ \sin 2\pi u\ ,\ \cos 2\pi u\ ]^T$
E. $\mathbf{s}(u) = [\ \cos (½)\pi u\ ,\ \sin (½)\pi u\ ]^T$
F. $\mathbf{s}(u) = [\ \sin (½)\pi u\ ,\ \cos (½)\pi u\ ]^T$

(18) [2 marks]

Consider the **cubic Bézier 2D curve segment p**(u), for $0 \leq u \leq 1$, whose first and fourth control points are $\mathbf{p}_0$ = (1, 1) and $\mathbf{p}_3$ = (10, 10), respectively.

Write the coordinates of the **second and third control points**, $\mathbf{p}_1$ and $\mathbf{p}_2$, so that **p**(u) passes through **all** its 4 control points.

You must write your answer in the following form:

$\mathbf{p}_1$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_2$ = (*x-coordinate*, *y-coordinate*)

$\mathbf{p}_1 = (\ a,\ \ a\ )$ —— [1 marks]
$\mathbf{p}_2 = (\ b,\ \ b\ )$ —— [1 marks]
where $1 < a < b < 10$.

(19) [4 marks]

Consider the following **parametric cubic polynomial 2D curve segment** for $0 \le u \le 1$:

$$\mathbf{p}(u) = \begin{bmatrix}5\\2\end{bmatrix} + \begin{bmatrix}3\\6\end{bmatrix}u + \begin{bmatrix}36\\18\end{bmatrix}u^2 + \begin{bmatrix}81\\54\end{bmatrix}u^3$$

Write the coordinates of the **four control points**, $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$, for the **cubic interpolating curve segment** that corresponds to $\mathbf{p}(u)$. The control points must be given in the right order such that $\mathbf{p}_0$ corresponds to $\mathbf{p}(0)$, and so on.

You must write your answer in the following form:

$\mathbf{p}_0$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_1$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_2$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_3$ = (*x-coordinate*, *y-coordinate*)

$\mathbf{p}_0 = \mathbf{p}(0) = (5,\ 2)$ —— [1 marks]
$\mathbf{p}_1 = \mathbf{p}(1/3) = (13,\ 8)$ —— [1 marks]
$\mathbf{p}_2 = \mathbf{p}(2/3) = (47,\ 30)$ —— [1 marks]
$\mathbf{p}_3 = \mathbf{p}(1) = (125,\ 80)$ —— [1 marks]

(20) [5 marks]

Consider the following **parametric cubic polynomial 2D curve segment** for $0 \le u \le 1$:

$$\mathbf{p}(u) = \begin{bmatrix}4\\2\end{bmatrix} + \begin{bmatrix}9\\6\end{bmatrix}u + \begin{bmatrix}3\\6\end{bmatrix}u^2 + \begin{bmatrix}6\\3\end{bmatrix}u^3$$

Write the coordinates of the **four control points**, $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$, for the **cubic Bézier curve segment** that corresponds to $\mathbf{p}(u)$. The control points must be given in the right order such that $\mathbf{p}_0$ corresponds to $\mathbf{p}(0)$, and so on.

You must write your answer in the following form:

$\mathbf{p}_0$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_1$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_2$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{p}_3$ = (*x-coordinate*, *y-coordinate*)

$\mathbf{p}_0 = \mathbf{p}(0) = (4,\ 2)$ —— [0.5 marks]
$\mathbf{p}_1 = (\mathbf{p}'(0) + 3\mathbf{p}_0) / 3 = (9, 6) / 3 + (4, 2) = (7,\ 4)$ —— [2 marks]
$\mathbf{p}_2 = (3\mathbf{p}_3 - \mathbf{p}'(1)) / 3 = (22, 17) - (33, 27) / 3 = (11,\ 8)$ —— [2 marks]
$\mathbf{p}_3 = \mathbf{p}(1) = (22,\ 17)$ —— [0.5 marks]

(21) [6 marks]

Consider the following **two parametric cubic polynomial 2D curve segments** for $0 \le u \le 1$:

$$\mathbf{p}(u) = (1-u)^3 \begin{bmatrix} 3 \\ 12 \end{bmatrix} + 3u(1-u)^2 \begin{bmatrix} 5 \\ 18 \end{bmatrix} + 3u^2(1-u) \begin{bmatrix} 11 \\ 18 \end{bmatrix} + u^3 \begin{bmatrix} 13 \\ 15 \end{bmatrix}$$

$$\mathbf{q}(u) = (1-u)^3 \begin{bmatrix} 15 \\ 15 \end{bmatrix} + 3u(1-u)^2 \begin{bmatrix} 21 \\ 16 \end{bmatrix} + 3u^2(1-u) \begin{bmatrix} 17 \\ 10 \end{bmatrix} + u^3 \begin{bmatrix} 23 \\ 12 \end{bmatrix}$$

Suppose we want to use a **cubic Bézier curve segment** $\mathbf{s}(u)$ to join to curve $\mathbf{p}(u)$ and $\mathbf{q}(u)$, such that $\mathbf{p}(1) = \mathbf{s}(0)$ and $\mathbf{s}(1) = \mathbf{q}(0)$. We also require C^1 continuity at all joint points.

Write the coordinates of the **four control points**, $\mathbf{s}_0$, $\mathbf{s}_1$, $\mathbf{s}_2$ and $\mathbf{s}_3$, for the **cubic Bézier curve segment** that corresponds to $\mathbf{s}(u)$. The control points must be given in the right order such that $\mathbf{s}_0$ corresponds to $\mathbf{s}(0)$, and so on.

You must write your answer in the following form:

$\mathbf{s}_0$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{s}_1$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{s}_2$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{s}_3$ = (*x-coordinate*, *y-coordinate*)

$\mathbf{s}_0 = \mathbf{p}(1)$ = (13, 15) —— [1 marks]
$\mathbf{s}_1$ = (15, 12) —— [2 marks]
$\mathbf{s}_2$ = (9, 14) —— [2 marks]
$\mathbf{s}_3 = \mathbf{q}(0)$ = (15, 15) —— [1 marks]

(22) [6 marks]

Consider the **cubic Bézier 2D curve segment** $\mathbf{p}(u)$, for $0 \leq u \leq 1$, whose control points are

$\mathbf{p}_0 = (0, 0)$
$\mathbf{p}_1 = (0, 32)$
$\mathbf{p}_2 = (32, 32)$
$\mathbf{p}_3 = (32, 0)$

Suppose $\mathbf{p}(u)$ is split into two segments at $u = 0.25$, where the sub-segment $\mathbf{s}(t)$, for $0 \leq t \leq 1$, is the same as $\mathbf{p}(u)$ for $0 \leq u \leq 0.25$.

Write the coordinates of the **four control points**, $\mathbf{s}_0$, $\mathbf{s}_1$, $\mathbf{s}_2$ and $\mathbf{s}_3$, for the **cubic Bézier curve segment** that corresponds to $\mathbf{s}(t)$. The control points must be given in the right order such that $\mathbf{s}_0$ corresponds to $\mathbf{s}(0)$, and so on.

You must write your answer in the following form:

$\mathbf{s}_0$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{s}_1$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{s}_2$ = (*x-coordinate*, *y-coordinate*)
$\mathbf{s}_3$ = (*x-coordinate*, *y-coordinate*)

$\mathbf{s}_0 = \mathbf{p}(0) = (0, 0)$ —— [0 marks]
$\mathbf{s}_1 = (0, 8)$ —— [2 marks]
$\mathbf{s}_2 = (2, 14)$ —— [2 marks]
$\mathbf{s}_3 = \mathbf{p}(0.25) = (5, 18)$ —— [2 marks]

—— **END OF PAPER** ——