NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING

EXAMINATION FOR
Semester 1, 2012/2013

CS3241 - COMPUTER GRAPHICS

Nov 2012 Time Allowed: 2 Hours

INSTRUCTION TO CANDIDATES

1. This is an OPEN book examination.
2. This examination paper contains **EIGHT (8)** questions and comprises **EIGHT(8)** printed pages.
3. Answer *ALL* questions within the spaces provided in this booklet.
4. You are allowed to use the back of the paper but please remember to state "P.T.O."
5. *Cross out any draft* or otherwise we will mark the poorer answers.
6. No calculator should be needed in this exam. You can leave your answer in surd form, namely, you can write $\sqrt{2}$ instead of 1.4142...
7. **In the programming questions, you can assume some basic normal vector arithmetic functions are provided for the 2D or 3D vector class `Vector2D (Vector3D)`, even a function `normalize(Vector2D)` to compute a normalized vector.**
8. Please write your matriculation number below, but NOT your name.

TIDINESS COUNTS!

We will deduct marks if your writing is too messy.

MATRICULATION NUMBER:_____

(this portion is for examiner's use only)

Question	Max. Marks	Score	Check
Q1	10		
Q2	6		
Q3	6		
Q4	8		
Q5	6		
Q6	8		
Q7	3		
Q8	8		
Total	55		

Question 1 [10 marks]

You are given 10 statements below. State if each statement is True (T) or False (F) by writing your answer in the box provided. A correct answer will give you 1 mark. An empty answer will give you zero, but ***a wrong answer will result in -1 mark***. (The lowest mark for this whole question is ZERO)

☐ The new generation TV or retina displays are too fine, so that they will not have any aliasing effect even without anti-aliasing techniques.

☐ If two parametric curves are G1 continuous, they are always C1 continous.

☐ The current highend game consoles, such as Playstation 3, Xbox, are capable of performing real time ray tracing for rendering.

☐ To model scales on the surface of a snake, the best and most efficient method is to model each scale one by one with transformation such as translation, rotation and scaling.

☐ Bump mapping works better with Phong shading than Gouraud shading in general.

☐ Implicit surfaces work better in ray tracing when compared to parametric surfaces.

☐ Environmental mapping is fast enough for real time rendering.

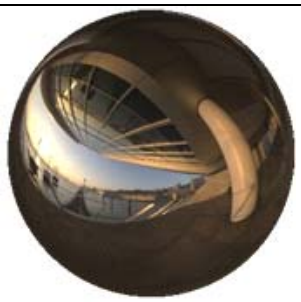
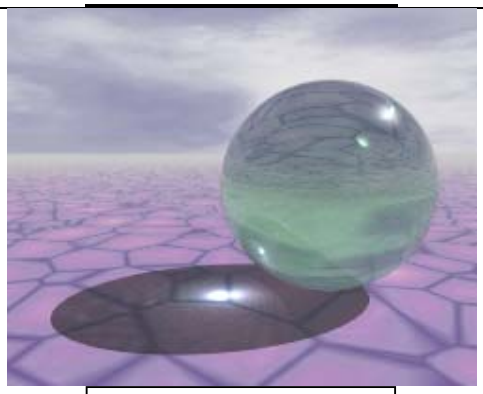
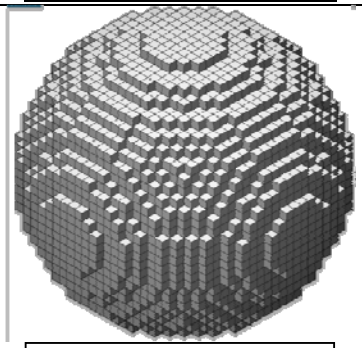
☐ If a sequence of transformation consists of only translations, all these transformation are equivalent to one single translation. It is also the same case for scaling and rotation.

☐ Halftoning and dithering techniques can be only used for converting grey scale images into black and white. Namely, these techniques are not applicable to color images.

☐ We can combine a sequence of translation, rotation and scaling to produce a perspective projection transformation.

Question 2 [6 marks]

Describe the best techniques taught in our course to draw the following scenes. For each picture, give ONE most important and unique technique.

<table>
<tr><td>  </td><td>  </td></tr>
<tr><td>  </td><td>  </td></tr>
<tr><td>  </td><td>  </td></tr>
</table>

Question 3 (6 marks)

Fill in the code for setting up the normal vectors for the following C++ codes to draw a cylinder for both smooth and flat shadings:

Answers:

```
void drawCylinder(double r,double k)
{
      int i,j;
      int n = 20;
      double x1,y1,x2,y2;

      for(i=0;i<n;i++)
            for(j=0;j<n;j++)
            {
                  x1 = sin(i*2*M_PI/n);
                  y1 = cos(i*2*M_PI/n);
                  x2 = sin((i+1)*2*M_PI/n);
                  y2 = cos((i+1)*2*M_PI/n);
                  if(m_Smooth)
                  {
                        glBegin(GL_POLYGON);

                              glVertex3d(r*x1,r*y1,j*k/n-(k/2));

                              glVertex3d(r*x1,r*y1,(j+1)*k/n-(k/2));

                              glVertex3d(r*x2,r*y2,(j+1)*k/n-(k/2));

                              glVertex3d(r*x2,r*y2,j*k/n-(k/2));

                        glEnd();
                  } else      {
                        glBegin(GL_POLYGON);

                              glVertex3d(r*x1,r*y1,j*k/n-(k/2));

                              glVertex3d(r*x1,r*y1,(j+1)*k/n-(k/2));

                              glVertex3d(r*x2,r*y2,(j+1)*k/n-(k/2));

                              glVertex3d(r*x2,r*y2,j*k/n-(k/2));

                       glEnd();
                  }
            }
}
```

Question 4 (1+6+1 marks)

There are three spheres in the space with centers (0,-2,11), (0,3,9) and (0,1,8). They all have the same radius of 2 units. A ray starts tracing from the eye which is at the position, (0,1,0) with direction (0,0,1). Which is the first sphere that is hit by the ray?

a) Set up the ray equation $l(t)$

b) Compute the position of the intersections of the ray with the balls, if any.

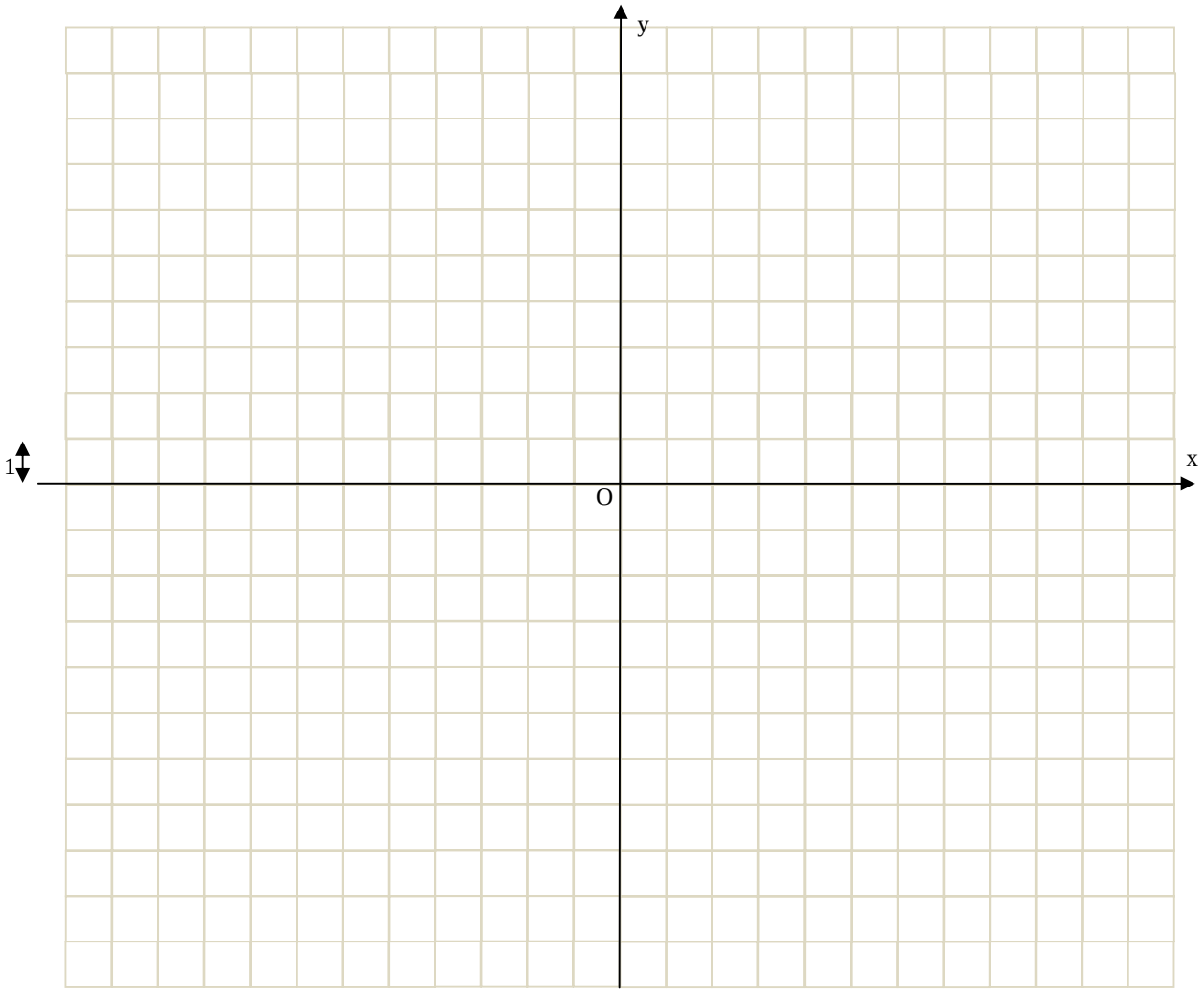
c) Which ball will be hit by the ray first? And why?

Question 5 (6 marks)

Draw in the space below what will the function `drawWhat(4)` produce on the screen, assuming that no other transformation is applied. Follow the scale given, namely, each square is 1 x 1 unit.

```
void drawUnitLine()
{
      glBegin(GL_LINES);
      glVertex2f(0,0);
      glVertex2f(0,8);
      glEnd();
}
```

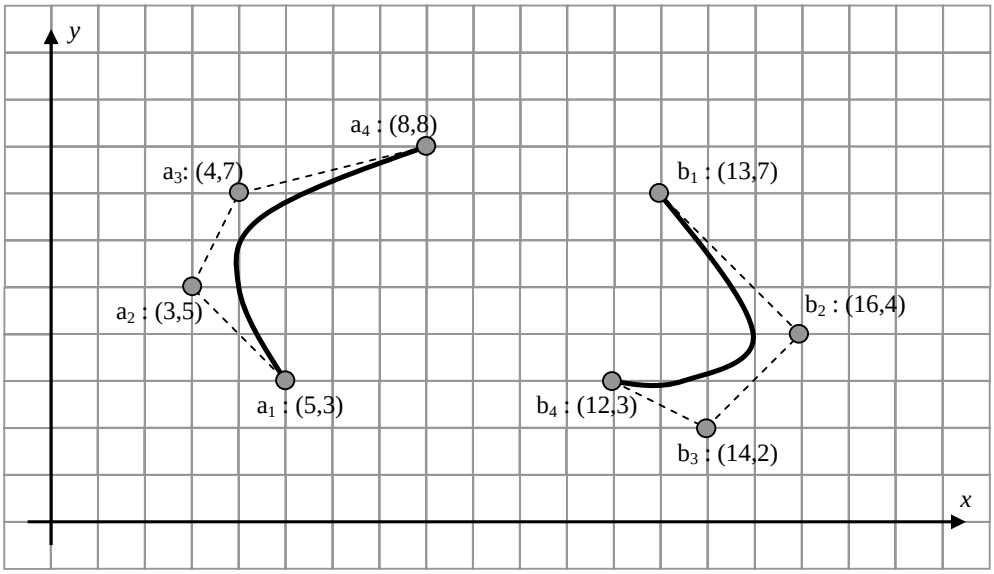
```
void drawWhat(int i)
{
      if(i==0) return;
      glPushMatrix();
            glScalef(0.5,0.5,0.5);
            drawUnitLine();
            glTranslatef(0,4,0);
            glPushMatrix();
                  glRotatef(-90,0,0,1);
                  for(int j=0;j<3;j++)
                  {
                        drawWhat(i-1);
                        glRotatef(90,0,0,1);
                  }
      glPopMatrix();
      glPopMatrix();
}
```



Question 6 [8 marks]

The diagram below shows two cubic Bezier curves with control points $a_1a_2a_3a_4$ and $b_1b_2b_3b_4$. Create two more cubic Bezier curves to join the given curves in order to maintain $\boldsymbol{C^1}$ continuity. Namely, create two curves with control points $c_1c_2c_3c_4$ and $d_1d_2d_3d_4$ such that the first curve connects a_4 to b_1 and the second one connects b_4 to a_1. Draw your new points accurately in the diagram in the correct locations, together with the broken lines indicating the control polygons. State the coordinates of the new points below the diagram. Also, draw roughly how the two new Bezier curves will look like.

Answers:



c_1 =

c_2 =

c_3 =

c_4 =

d_1 =

d_2 =

d_3 =

d_4 =

Question 7 [3 marks]

The following pseudo-codes suppose to copy a block of pixels from one location in the frame buffer to another.

```
Copy_Pixel (int from_Xmin, int from_Ymin, int from_Xmax, int from_Ymax,
                  Int to_Xmin, int to_Ymin )
{
      for ( int x = from_Xmin;    x <= from_Xmax;      x ++ )
            for (int y = from_Ymin;   y = from_Ymax;    y++ )
{
                  color = Read_Pixel ( x, y );
                  Write_Pixel ( to_Xmin + ( x – from_Xmin),
                              to_Ymin + ( y – from_Ymin), color );
            }
}
```

Draw a case to show that the above algorithm fails to do the proper copying.

Question 8 [2+3+3 marks]

Given a polygon with four vertices A:(10,10), B:(20,20), C:(50, 10), and D: (30, 0). A horizontal scanline is moving from the top to bottom of the screen at every 0.5 position as in the lecture notes.

a) Which one is the first scanline? And what are the two edges that intersect this first scanline?

b) What are the left and right intersections of this scan line with the two edges above?

c) What are the two Δx values (according to the lecture notes of SCA) for the first two edges?

- End of Paper -