**Introduction to Computer Graphics**

**Name:**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Honor Code:**\_\_\_

**Instructions:** Select one of two options:

1. Select at least one exercise from questions 7-10. Answer a total of 150 points excluding the program.
2. Answer any 75 points of questions.
3. Give complete definitions of vector and affine spaces. **(10 points)**
4. Write the transformation matrices for the following operations: (assume homogenous coordinates) **(10 points)**
   1. **Identity**
   2. **Translation of a point by dx, dy, dz**
   3. **Scaling by the values sx, sy, sz**
   4. **Rotation about the z-axis by θ**
5. We derived the matrix for rotation about the z-axis in class. Please derive the matrix for rotation of a point about the x-axis. **(20 points)**
6. Perform the following vector operations **(15 points, 3 points each)**
   1. Vector Addition: (a, b, c) + (d, e, f)
   2. Scalar-Vector multiplication: x (x, y, z)
   3. Dot Product: (a, b, c) • (d, e, f)
   4. |(a, b, c)|
   5. Assume that the vector (a, b, c) is not the zero Vector. Normalize( (a, b, c))
7. Two vectors, *v1* and *v2*, are considered orthogonal, the angle between them forms a 90 degree angle.
   1. What is the value of the dot product  (10 points)
   2. If *v1* = (a, b, c), give a possible vector value for *v2*. (15 points)
8. Given vectors, **p** and **q** prove or disprove the following:
   1. **p** • **p** = |**p**|2 **(10 points)**
   2. **p** × **q = q**  × **p (15 points)**
9. Consider two lines. Line1 is defined by a point (0, 0, 0) and a direction (1,1, 0). Line 2 is defined by the point (0, 1, 0) and a direction (1, -1, 0).
   1. Write the parametric equations for both lines. **(5 points)**
   2. Using the equations from part a, find the intersection**. (15 points)**
10. A circle with radius 100 centered at (0, 5, 0). It is to be rotated 45 degrees about the vector (1, 1, 1). A point P(x,y,z) is on this circle. Give the coordinates of point P \*\*\*after\*\*\* the rotation. ***Rotate using Matrices***. (Do not write a program.) Show your work. If you only submit the rotated point, you will receive 10 points. **(30 points).**
11. The equation for a plane is Ax + By + Cz + D = 0…..Our in-class discussion offered another representation of this equation: n • (P – P0) = 0 where n is the normal(A,B,C) to the plane, P=(x, y, z), and P0 is a point on the plane. Using this information and what you know about vector operations, find the equation of the plane that contains the following three points: (1, 0, -1), (3, 1, 4), and (2, -2, 0). **(30 points)**
12. Given three arbitrary points p0, p1, and p2 how can you test for collinearity? In other words, if we wanted to write a program that draws a triangle from user input, how can we tell if the three given points are not on the same line? **(20 points)**
13. Program option**(75 points):** Write a program that displays four viewpoints of an octahedron with points (-1, 0, 0), (1, 0, 0), (0, -1, 0), (0, 1, 0), (0, 0, 1), (0, 0, -1). Note this program is similar to the cubes program. Examples of this exercise can be found in ~mcameron/CS440/Exam2. To receive full credit your program should perform as the program octahedron. The bonus version should be similar to the program goodOctahedron. Specifically, your program should do the following:
    1. Successfully compile. Grade will be zero for programs that do not compile.
    2. Must attempt this assignment. You may used the cube.c program as a template. However, your modified version must show octahedral shapes not cubes.
    3. Ask (and receive) the user for an axis of rotation. (5 points)
    4. In all four views, display the standard axis in black and the given axis of rotation in white. The axes will not rotate if enter the instructions for drawing the axes before any rotation calls. (10 points)
    5. Top left view presents point version of octahedron. ( 5 points)
    6. Top right view presents a spinning wire frame of octahedron about the y-axis (5 points)
    7. Bottom left view presents spinning flat shaded version of the octahedron about the y-axis. The bottom of the octahedron is to be placed at the origin. (15 points)
    8. Bottom right view presents a spinning smooth version of the octahedron that spins about the axis given by the user. The bottom point of the octahedron is to be placed at the origin (30 points)
    9. Bonus: Align the octahedron’s center to the axis of rotation. (5 points)