Problem Solving Task Part B

Due: Wednesday, 11:50pm, 25 October 2017.

Group of 2 people (1 or 3 people teams need the unit coordinator’s approval).

**QUT Bot Jr.**

For full marks in this submission you need to complete all five of the tasks below. In addition to the functionality for each task, you need to submit a Statement of Completion that documents: the tasks you complete, how to use your software and screenshots that provide examples of your executing code.

The theme of the following tasks is articulated movement. The character is QUT Bot Jr. , who is a fan of Luxo Jr. What we want is some robot-like thingy that tries to mimic some actions of Luxo Jr.

*Note: Feel free to use unity’s built-in transformation functions or your own version of the transformation matrix for this assignment.*

**Task 1 for 50%**

1. Complete all theoretical questions listed at the end of this document. Organise the questions and answers in one folder called IGB283QA, then one sub-folder per major question and its answers. The files should be in the format of Word, PDF, or scanned hand-writing on paper.
2. Review the Luxo Jr. videos here:

a) Luxo Jr.: <https://www.youtube.com/watch?v=D4NPQ8mfKU0>

b) Luxo Jr. Spoof: <https://www.youtube.com/watch?v=QR1h2cfpPHM>

1. Copy the Robot Arm model you have created in Workshop 7 – articulated motion, including vertices, vertex indices, and relative angles between different components. Name the model QUT Jr. You can apply translation, rotation, and scaling to these components. You can also change the shapes and colours of these components to make it look more attractive.
2. Write C# Script code that continually rotates the head of QUT Jr. to mimic nodding of a person.
3. Write C# Script code that continually moves QUT Jr. left to right to mimic walking of a person. To do this, you need to think of the bot as a tree. The root of the tree corresponds to the entire object and its children to subcomponents of the bot.
4. Write code that continually moves QUT Jr. left to right while its head is nodding. This can be achieved by traversing the tree, updating information and then displaying by again traversing the tree to draw the component objects.

**Task 2 for 65%**

1. Complete Task 1
2. Write C# Script code that continually moves QUT Jr. up and down to mimic jumping up of a person (You only need to implement position change. You do not have to follow the exact gravity law).
3. Write C# Script code that makes QUT Jr. jump forward (i.e., up, forward and down) continuously.

**Task 3 for 75%**

1. Complete Tasks 1 and 2
2. Add C# Script code to steer your bot around the screen using the keyboard.

* key ’a’ for left, ’d’ for right
* key ’w’ for jumping up, key ’s’ for jumping forward

1. When no key is pressed, your bot should move at a constant speed in the direction of its motion (no moving backward) and never escape the world view.

**Task 4 for 85%**

1. Complete Tasks 1, 2 and 3
2. Add a ground to your scene (a straight-line is fine).
3. Add functionality so that when the ’**z**’ key is hit, keyboard control stops working and the bot falls straight down to the ground
4. The bot should accelerate all the way down to the ground.
5. When crash-landed, QUT Jr. faces down. The keyboard control should be resumed.
6. QUT Jr. lays on the floor for a few seconds, then bends up slowly, and finally stands straight and strong again. Hooray!

**Task 5 for 100%**

1. Describe an additional feature and argue for it’s inclusion in your Statement of Completion. For example, you can make QUT Jr. pick or crash a ball or box.
2. Implement your feature.

**What and How to Submit**

Your submission is to be uploaded onto the BlackBoard site of this unit.

The files to include in the submission are:

1. A statement of completion in Word or PDF format including program design, key screenshots, and user instructions of how to run your program etc. Provide one or two typical screen-shots for each task demonstrating your execution of the code.
2. A brief Statement of Contribution from the team members mentioning each team member’s role in this assignment (you are expected to make roughly equal contributions). Remember to include the student name and ID of all team members.
3. A copy of your unity project files (zip up the entire folder if possible)
4. A copy of any input files that are required by your program (if implemented. You do not need to have file I/O in this assignment).
5. A short video demonstrating the major functions of your application. Upload your video to YouTube and only include the link to your video in the report.

Turn over page for the theoretical questions.

**Theoretical questions**

1. **Vector algebra**

Given vectors **V**1 = <1, 2, 3>, **V**2 = <4, 5, 6> , and **V**3 = <7, 8, 9>.

1. The magnitude of **V**1 is:
2. The unit vector along the direction of **V**1 is:
3. The homogeneous representation of **V**1 is:
4. **-V**1 =
5. **V**1 + 2 **V**2 – **V**3  =
6. Draw a diagram to illustrate e).
7. **V**1 **x** **V**2  =
8. **V**1 ● **V**2  =
9. Calculate the angle between **V**1 and **V**2  .
10. Determine the vector which is the projection of **V**1 onto **V**2  .
11. Briefly explain how to construct a Cartesian coordinate system using **V**1 and **V**2
12. **Matrix algebra**

Let , , , and .

1. Calculate the inverse of C using elementary row operations and augmented matrix = . Show your key steps.
2. **Points, lines and planes**

Given three points, *P* (2, 0, –1), *Q* (–1, 3, 4) and *R* (3, –1, 2).

1. Find the vectors and .
2. Calculate the area of the triangle PQR.
3. Write the parametric representation of the line PR: =
4. Derive the equation of the plane that passes through PQR.
5. Calculate the normal of the plane that passes through PQR.
6. Given the parametric representation of a line:

Calculate the intersection of the line and the plane derived in d).

1. Translate the triangle PQR for a distance of <1, 2, 3>. Construct the translation matrix first, then apply the translation matrix to P, Q, and R to get their new location P’, Q’, R’. Draw a diagram to show that you result is right.
2. Scale the triangle PQR relative to the origin and along x axis for a factor of 2. Construct the scaling matrix and apply it to P, Q, and R to get their new locations P’, Q’, R’. Draw a diagram to show that you result is right.
3. Project the triangle PQR to XY plane using orthogonal projection. Construct the projection matrix and apply it to P, Q, and R to get their new locations P’, Q’, R’. Draw a diagram to show that you result is right.
4. Assume the eye is located in the origin and the projection plane is located at z = -5. Construct the projective projection matrix and apply it to P, Q, and R to get their new locations P’, Q’, R’. Draw a diagram to show that you result is right.