**Physics 2016**

**Question 1**

**(a) Mathematics and in particular vector mathematics would be used throughout**

**games. Given the following situation:**

**Vector3 TargetPosition = new Vector3(8,-3,-7);**

**Vector3 PlayerPosition= new Vector3(5,1,5);**

**Vector3 PlayerForward = Vector3.normalise( new Vector3 (2,-6,-9));**

**(i) Find the vector PlayetToTarget**

(8-5),(-3-1),(-7-5) = (3,-4,-12)

**(ii) How far is the player from the target?**

Size of (3,-4,-12)**=**  =  = = 11

**(iii) Calculate the PlayerForward vector.**

Need size of (2,-6,-9) = = = = 11

Answer = (2,-6,-9)

**(iv) Determine, by appropriate use of a scalar dot product, whether the Target is**

**"in front" of the Player**

(3,-4,-12) . (2,-6,-9) = ((3.2) + ((-4).(-6)) + ((-12).(-9)) = 138/11 = 12.54 > 0 So yes the Target is in front of the Player

**(b) Frame rates are a key consideration when implementing movement, and in particular keeping movement Frame Rate Independent**

**(i) What are the physical rules governing motions that are used to ensure Frame Rate Independence? (3 Marks)**

s = u \* t

v = u + a\*t

F = m \* a

**(ii) Illustrate (code or pseudo code) how Frame Rate Independent motion could be implemented (4 Marks)**

transform.position += velocity \* Time.deltaTime;

velocity += acceleration \* Time.deltaTime;

**(iii) Illustrate how forces could be applied to an object, giving justification with reference to the appropriate physics formula. (3 Marks)**

void FixedUpdate()

{

rb.AddForce(transform.forward \* thrust);

}

public void AddForce(float x, float y, float z, ForceMode mode = ForceMode.Force);

**(iv) Describe how the following could be implemented.**

**• Gravity**

Acceleration = Vector3.Down \* 9.f;