Fantazm - Unity Assessment Test
Jeffrey Saydam

1. **Kunnen threads gebruikt worden om een texture in runtime aan te passen?**

Nee, dat kan niet, omdat Unity het niet toestaat dat een thread toegang heeft tot de interne functies van Unity (zoals van MonoBehaviour).

2. **Kunnen threads gebruikt worden om een GameObject in de scene te verplaatsen?**

Nee, je kan niet vanuit een thread direct toegang hebben tot de interne functies van Unity. Wat wel kan is dat je een thread gebruikt met algoritmes om zo de verplaatsing van een GameObject te voorspellen/berekenen.

3. **Verbeter deze code door gebruik te maken van threads, zdoat de 1.000.000 random nummer generatie draait zonder in te boeten op performance.**

```
using System.Threading;

class RandomGenerator : MonoBehaviour
{
    public float[] randomList;

    private Thread t1;

    void Start()
    {
            randomList = new float[1000000];
            t1 = new Thread(Generate());
            t1.Start();
    }

    void Generate()
    {
            System.Random rnd = new System.Random();
            for(int i = 0; i = <randomList.Length; i++)
            {
                    randomList[i] = float() rnd.NextDouble();
            }
    }
}
```

4. **Leg uit wat een vertex en een pixel shader is.**

Vertex: Een vertex is een punt of hoek waar edges elkaar snijden. Bijvoorbeeld een vijfhoek heeft 5 vertices (meervoud van vertex), omdat hij 5 hoeken heeft.

Pixel shader: Een ander woord voor pixel shader is fragment shader. Een pixel shader is een programma dat de kleur, brightness, contrast en andere karakteristieken berekent voor één pixel.

5. **Waarom moet je gebruik maken van Time.deltaTime om dingen die afhankelijk zijn van tijd goed te laten draaien?**

Time.deltaTime is de tijd tussen 2 frames, deze wordt berekend door de engine zelf. Door met Time.deltaTime te vermenigvuldigen zet je de tijd in frames om naar de tijd in seconden.

6. **Maak deze code af, zodat het GameObject dat dit script bevat met een constante snelheid (speed) naar het doel (target) beweegt en dat het stopt als het 1.0 of minder units afstand heeft.**

```
class Mover : MonoBehaviour
{
    Vector3 target;
    float speed = 3.0f;

    void Update()
    {
            //if distance is bigger than 1 unit
            if(Vector3.Distance(transform.position, target) >= 1.0f )
            {
                    //Calculate the step distance
                    float step = speed * Time.deltaTime;
                    //Move to target
                    transform.position = Vector3.MoveTowards(transform.position, target,
                    step);
            }
    }
}
```

7. **Welke van de twee voorbelden is sneller in uitvoering?**
   A: 1000 GameObjects ieder met een MonoBehaviour die een Update() callback uitvoert.
   B: Een GameObject met een MonoBehaviour met een Array van 1000 classes, met ieder een eigen Update() callback.

   A is sneller, deels omdat Unity dan GameObjecten kan uitschakkelen die niet van belang zijn op dat moment, zonder dat de hele code dan niet meer werkt.

8. **Plaats de volgende events in de volgorde, zoals ze aangeroepen worden als een applicatie afsluit.**

Awake() > OnEnable() > Start() > Update() > LateUpdate() > OnGUI() > OnDestroy() > OnApplicationQuit() > OnDisable()

9. **Leg uit wat er met de volgende code mis is.**

   De fout zit hem in de zin:

   ```
   transform.position.x = 10;
   ```

   transform.position.x is een float. 10 is geen float maar een int.
   De oplossing:

   ```
   transform.position.x = 10.0f;
   ```

Demo Code (Against The Aliens Of Algadar)

Controls:
- Schip besturen: WASD
- Schieten: Spatie

Alle code is door mij gemaakt.
De meeste plaatjes zijn van internet, waarna ik ze geëdit heb ter efficiëntie en samenhoring.
Audio is van internet, waarna ik ze geëdit heb om ze passend te maken.

Deze game voldoet aan alle punten:
1. Een vloeiende scrollende achtergrond.
2. Een schip wat je bestuurd.
3. Snel schieten van bullets naar objecten en tegenstanders.
4. Actieve tegenstanders die in patronen het beeld in komen. (Aliens)
5. Passieve tegenstanders. (Vliegtuigen)
6. Explosies bij impact.
7. Gebruik van Object Pooling. (zie Scripts > Object Pooling)