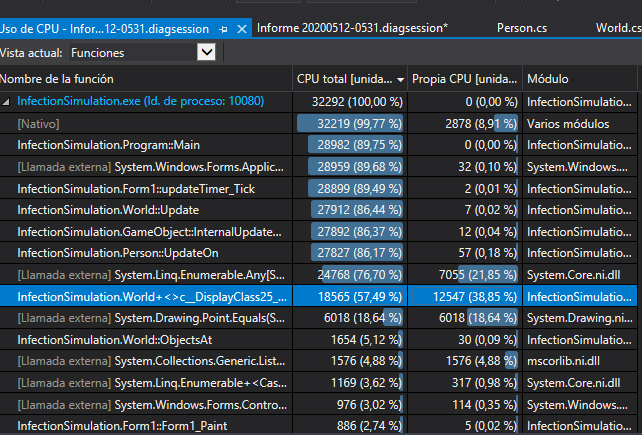
Hipotesis

Consignas:

1Realizar mediciones de performance que permitan diagnosticar el problema.  
2. Escribir qué problemas encontró en la simulación en función de las mediciones.  
3. Proponer una solución al problema encontrado, describirla en sus palabras con el mayor detalle posible.  
4. Implementar la solución planteada haciendo los cambios en el código que sean necesarios.  
5. Validar la implementación realizando nuevas mediciones. ¿Se resolvió el problema? En caso negativo, ¿qué otras soluciones alternativas se le ocurren?

1. Utilice como medición un informe de diagnóstico de unity sobre el CPU.
2. Durante 30 seg que se ejecutó el programa, encontré que la función public IEnumerable<GameObject> ObjectsAt(Point pos) que se encuentra en la clase WORLD, es llamada muchas veces y ocupa un 39% del CPU.



3) El programa se Actualiza en bases a tics por segundo, por cada actualización el programa calcula la posición de la una persona. (Persona 1 de 5000)

Esta persona 1 Recorre las 5000 personas para ver cual está tocando y revisa si está infectada o no.

Luego la persona 2 (2 - 5000) realiza el mismo calculo.

Este es el problema del programa, que cuantas mas personas se encuentre, más cálculos se tienen que realizar y se ralentiza el programa

La solución que planteo es, cada ves que se actualice el programa, recorrer las 5000 personas y guardar la posición de los infectados.

Luego se recorrerán las 5000 personas nuevamente y se mirara una por una quien toca y quien no toca al infectado.

4)La implementación ya fue puesta a prueba.

5)El programa mejoro mucho, pero se sigue ralentizando, esta optimización fue positiva ya que el programa mejoro notablemente.