ICP #5

(A Dry Run for Final Exam)

Group Member Names:Bilal Berat Postalcıoğlu, Berat Burak Tetik, Emre Fişek, Mert Can Geyik, Ceyda Düzgeç

Q1) Sweep & Prune (SAP) algorithm is used for broad phase collision detection.

a. Why do we need broad phase collision detection?
b. SAP can operate on 1, 2 or 3 axes depending on your choice. Compare the advantages and disadvantages of using just 1 axis versus using all 3 axes in SAP.
c. Assume you want to work with just 1 axis. What metric(s) can be used to choose which axis?
d. How would you design an experiment to evaluate how well your approach performs using a single axis (when compared to 3 axes).

ANS:

a. In the game there are multiple objects which might or might not collide. If the collision possibility of all the objects would be calculated it would need too much processing power. Instead, sweep and prune can be used which processes collision of only the possible object collisions by sorting them on axis and checking the intersections of their coordinates.

b. Advantages of using only one axis is that the complexity would be less so it would need less processing power. But, it would detect less collisions or detect extra detections that would not happen. On the other hand, 3 axis detection would find collisions more accurately but would need more processing power.

c. The object number intensity over the coordinates, lower bound and upper bound coordinates over the axis, the speed of objects related to axis, intersecting object number

d. We would create several objects in the 3D environment and count the real detection number. Compare this number with the 1-axis and 3-axis detection SAP to find their accuracy. Also we would compare their usage of processing power.

Q2) What is the tunneling problem? Explain it verbally. If you happen to increase the physics update – i.e. FixedUpdate()- period on Unity would it have any effect on the tunneling problem? (Make it worse? Solve the tunneling problem? No effect?). Explain why?

In video games all objects have a specific position which is updated by the physics engine in every frame. When the engine computes the position of objects it should also check for the collisions and respond to them. If an object is moving too fast or the frame rate is very low the physics engine may compute the new position of the object on the other side of the collided object without detecting collision. This is called tunneling.

Increasing physics update may prevent tunneling as it can detect collisions more precisely.

Q3) Assume that there is a broad phase collision detection algorithm called ALG485. Here are some information about ALG485:

- When ALG485 signals a possible collision between two collidable game objects, the probability that they actually collide is 0.88.
- ALG485 misses 0.08 of the actual collisions.
- For ALG485, computation cost of a single (potential)collision detection C1.
- For S&P, the computation cost of a single (potential) collision detection is 1.15 C1.
- Cost of narrow phase collision detection is C2= 10^3 C1

Comment on the properties of ALG485. As a game programmer would you use it for broad phase collision detection purposes in your games? Why? (Max 4 sentences)

I would use ALG485 if I am going to deal with a high number of game objects and some collision detection misses not so critical for the game itself. The algorithm detects nearly 10% of detections as false positives. So, if detecting the collisions right is crucial and there are limited gameobjects I would not use ALG485. Since cost of narrow phase collision high (10^3*C1) also I would not use ALG485 if the performance of the game is crucial.

Q4) Indicate whether the given statements are True or False (Circle the letter T or F). Write the reason for your choice. Credit will be given only if your explanation is correct.

		TRUE	**FALSE**
1)	Shortest path algorithm Dijkstra can <u>never</u> be used for path finding purposes in games.	T	[✓ F]
	Reason: False, The objects in the games can be represented as nodes in a graph, and since Dijkstra's shortest path algorithm is used to find shortest path between two nodes it can be used in path finding purposes in games as well.		
2)	Insertion sort is a good candidate to keep track of object locations in Sweep and Prune algorithm	[✓ T]	F
	Reason:		

Insertion sort is fast in average over the not so distinctive element sets. It’s best-case complexity is O(n)		
3) On a game scene where game objects are concentrated on certain locations Multi SAP performs better than plain SAP	T	F
Reason: As the number of objects increases in the SAP structure, updating it for a single object takes longer time.		