2. Theoretical Tasks: AI (5 Points)

2.1 Steering behaviours

You are given two objects, A and B, with the given positions and speeds. Fill out the following table of steering outputs if A is given B as its target. The steering output could be requested velocity or acceleration. In both cases, it is a two-dimensional vector.

The maximal speed of A should be 2, the maximal acceleration of A should be 0.5.

Steering behavior	Steering output (vec2)
Steer (kinematic)	Vector to target with max speed Normalize(3, -4) * 2
Flee (kinematic)	Vector away from target with max speed Normalize(3, -4) * -2
Steer (dynamic)	Vector to target with max accel. Normalize(3, -4) * 0.5
Flee (dynamic)	Vector away from target with max accel. Normalize(3, -4) * -0.5
Pursue	How long do we take? Length of the way: Length of (3, -4) = 5 How long do we take? 5 / 2 = 2.5 New Position of B: (8, 2) + (3, 4) * 2.5 = (8,2) + (7.5, 10) = (15.5, 12) Steering output: Normalize((15.5, 12) – (5, 6)) * 0.5
Evade	How long do we take? Length of the way: Length of (3, -4) = 5 How long do we take? 5 / 2 = 2.5 New Position of B: (8, 2) + (3, 4) * 2.5 = (8,2) + (7.5, 10) = (15.5, 12) Steering output: Normalize((15.5, 12) – (5, 6)) * -0.5

2.2 Interruptible AI algorithms, Anytime algorithms

a) Can A* be interrupted during its execution and continued at a later time? If yes, describe why. If not, give a counterexample.

Yes. If we put the data (open-node-list, …) on the heap, we can continue at another time.

b) Imagine that an interruptible pathfinding algorithm is interrupted and continues 1 second later. Are there problems to be expected in this case? If so, what could the problems be?

The path could be invalid (terrain or buildings changed/destroyed), the move command could be canceled by the player, weights can have changed, …

c) If a) is true, does this make A* an Anytime algorithm? (i.e. Would the implementation be able to provide a valid path after being interrupted?)

Not without changes. If the algorithm is interrupted before it finished, it does not have a valid path to the target. You can search for "Anytime A*" to find algorithms with such changes.

2.3 AI LOD

Imagine a scene with many characters walking in random directions. In this scene, an AI LOD system is used. Nearby characters use collision avoidance, far away characters may freely interpenetrate.

What problems could arise if characters exploded when they collided?

If an AI stays in one of the LOD regions, everything is fine.

If AI's interpenetrate (= far LOD) and the player comes closer, they can suddenly explode as they get into the near LOD.

Depending on the implementation, characters on the far-side could wander into those close to the border.