**Weapon Lock Algorithm – Dry Run**

**Case:**

|  |  |  |
| --- | --- | --- |
| **Enemy Number** | **x** | **y** |
| 0 | 140 | 600 |
| 1 | 60 | 200 |
| 2 | 30 | 40 |

Weapon 1 Position – (40, 60)

Weapon 2 Position – (80, 60)

Radius of 100 each

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Line No.** | **i** | **totalEnemies** | **weaponNumber** | **Weapons[weaponNumber] .radius > Enemies[i].position?** | **Return** |
| 3 |  | 3 | 1 |  |  |
| 4 | 0 |  |  |  |  |
| 6 |  |  |  | FALSE |  |
| 4 | 1 |  |  |  |  |
| 6 |  |  |  | FALSE |  |
| 4 | 2 |  |  |  |  |
| 6 |  |  |  | TRUE |  |
| 8 |  |  |  |  | 2 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3 |  | 3 | 2 |  |  |
| 4 | 0 |  |  |  |  |
| 6 |  |  |  | FALSE |  |
| 4 | 1 |  |  |  |  |
| 6 |  |  |  | FALSE |  |
| 4 | 2 |  |  |  |  |
| 6 |  |  |  | TRUE |  |
| 8 |  |  |  |  | 2 |

isInRange() fed into by LockOnEnemy() algorithm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Line** | **dx** | **dy** | **(dx^2 + dy^2)** | **Sqrt(dx^2 –dy^2)** | **Return** |
| 2 | 100 |  |  |  |  |
| 3 |  | 540 |  |  |  |
| 5 |  |  | 301600 |  |  |
| 6 |  |  |  | 549 |  |
| 9 |  |  |  |  | FALSE |
| 2 | 20 |  |  |  |  |
| 3 |  | 140 |  |  |  |
| 5 |  |  | 19600 |  |  |
| 6 |  |  |  | 140 |  |
| 9 |  |  |  |  | FALSE |
| 2 | -10 |  |  |  |  |
| 3 |  | 20 |  |  |  |
| 5 |  |  | 500 |  |  |
| 6 |  |  |  | 22 |  |
| 12 |  |  |  |  | TRUE |
| 2 | 60 |  |  |  |  |
| 3 |  | 540 |  |  |  |
| 5 |  |  | 295200 |  |  |
| 6 |  |  |  | 543 |  |
| 9 |  |  |  |  | FALSE |
| 2 | -20 |  |  |  |  |
| 3 |  | 140 |  |  |  |
| 5 |  |  | 20000 |  |  |
| 6 |  |  |  | 141 |  |
| 9 |  |  |  |  | FALSE |
| 2 | 50 |  |  |  |  |
| 3 |  | 20 |  |  |  |
| 5 |  |  | 2900 |  |  |
| 6 |  |  |  | 53 |  |
| 12 |  |  |  |  | TRUE |

This works as expected. The two weapons both choose to lock onto Enemy Number 3 and no other enemy as this is the only enemy within their range.

**Case:**

|  |  |  |
| --- | --- | --- |
| **Enemy Number** | **x** | **y** |
| 0 | 60 | 20 |
| 1 | 500 | 200 |
| 2 | 45 | 20 |
| 3 | 250 | 20 |

Weapon Position – (40, 20)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Line No.** | **i** | **totalEnemies** | **weaponNumber** | **Weapons[weaponNumber] .radius > Enemies[i].position?** | **Return** |
| 3 |  | 4 | 1 |  |  |
| 4 | 0 |  |  |  |  |
| 6 |  |  |  | TRUE |  |
| 8 |  |  |  |  | 0 |
| 4 | 1 |  |  |  |  |
| 6 |  |  |  | FALSE |  |
| 4 | 2 |  |  |  |  |
| 6 |  |  |  | TRUE |  |
| 8 |  |  |  |  | 5 |
| 4 | 3 |  |  |  |  |
| 6 |  |  |  | FALSE |  |

isInRange() fed into by LockOnEnemy() algorithm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Line** | **dx** | **dy** | **(dx^2 + dy^2)** | **Sqrt(dx^2 –dy^2)** | **Return** |
| 2 | 20 |  |  |  |  |
| 3 |  | 0 |  |  |  |
| 5 |  |  | 400 |  |  |
| 6 |  |  |  | 20 |  |
| 12 |  |  |  |  | TRUE |
| 2 | 460 |  |  |  |  |
| 3 |  | 180 |  |  |  |
| 5 |  |  | 244000 |  |  |
| 6 |  |  |  | 493 |  |
| 9 |  |  |  |  | FALSE |
| 2 | 5 |  |  |  |  |
| 3 |  | 0 |  |  |  |
| 5 |  |  | 25 |  |  |
| 6 |  |  |  | 5 |  |
| 12 |  |  |  |  | TRUE |
| 2 | 210 |  |  |  |  |
| 3 |  | 0 |  |  |  |
| 5 |  |  | 44100 |  |  |
| 6 |  |  |  | 210 |  |
| 9 |  |  |  |  | FALSE |

This does not work as expected. This is because the algorithm finds two potential Enemies to lock onto, and instead locks onto the enemy that is furthest away. This can be fixed by putting in a break case on line 9 and so the weapon only locks onto the first enemy within its radius.