|  |  |
| --- | --- |
| **Spec** | **Completed** |
| Discrete (grid based) world | X & Y coordinates of a panel make grid |
| 8 possible movement directions | Roll 0-7 from top left clockwise to pick direction  Roll 0-2 to pick left one, forward or right one |
| Ants created at random positions in the world | Ants spawn at random coordinates within panel  Displayed using rectangle brush |
| Find food | Check for food within 100 steps on either axis (diagonal count as 2)  *Picks closest food to nest* |
| Find nest | Check for nest within 100 steps on either axis (diagonal count as 2)  *Currently always takes newest nest within sight range* |
| Take food to nest | If have food & knows where nest is return to nest |
| Head towards food then nest when they have it | If have food & knows where nest is return to nest |
| Ask other ants for information   * Don’t tell “enemy” ants where food is | Compares colony information freely.  If ants in same colony then shares food location.  Compares closest food – ant will remember food with least steps between itself and colony |
| After food deposited, head back towards known food source   * If no food, wander for more | If have food & knows where nest is return to nest |
| Sight radius to detect food, nest & ants   * Different radii possible | Check for food/nest within 100 steps on either axis (diagonal count as 2) |
| Ants can stack on same tile | Do not program collision detection |
| Ask closest or all ants for info | *Currently cycles through ants in list order* |
| Click drops n units of food | Set as 8 units |
| Each ant removes m units of food from stack |  |
| m < n | Ants collect 1 as default |
| Alternate click drops nests | Done |
| Ant may forget food and/or nests at any time | Ant rolls random memory quality on spawn.  Ant rolls 0-memory quality every tick.  If 0 forget both, if 1 forget colony if 2 forget food. |
| Separate population of robber ants **\*EC\*** |  |
| Remember where they stole food **\*EC\*** |  |
| Return to try to find more ants with food **\*EC\*** |  |

*If Distance between ant and objective > panel.width/2*

*Use inverted case statement for movement (opposite directions)*

*Same with panel.height*

**Click drops n units of food**

*If food diminished flag all ants to not spread food location & remove from list*

*If at food & != has food then forget food*

**Each ant removes m units of food from stack**

*Change HasFood Boolean to Int. Ant picks up 1-3 food per time. Make random 1-3 on spawn?*

**m < n**

*Fucking done.*

**Alternate click drops nests**

*Done. Shift + click can drop thief ant nests?*

**Separate population of robber ants**

*Spawn like regular ants. Exactly the same with a few exceptions:*

*Robber nest*

*Rob ant with food*

**Remember where they stole food**

*Remember pixel coordinate like ants remember food.*

**Return to try to find more ants with food**

*Move to current objective like with regular ants.*