# Generalised model

* Giulo handed you a bunch of puzzle pieces, make them work
* Happiness is established via a function; strip it out from agent\_step! And replace it with calling the function
* Should get the function to create the vectors of same/other group? Or is there a way to optimise by only creating them when needed (making friends). Hard to quick-check what group a neighbour is in because it has to move through the graph layer back down to the agent layer

# Eight-friends

Model in which every agent has at least eight friends. Connections with the other group are broken until they drop below eight, then the agent makes a new connection to bring them back up to eight. Eight is chosen because it is the most likely number of neighbours an agent would have in a Schelling segregation physical space grid.

* Set up architecture for collecting data as the model runs – look at schelling & schoolyard examples but also talk to Giulio
* Strip out the redundant grid space initialisation in the model

# Gathering data

* Use model\_step to take the dimensionality and output it to a csv?