METHODS:

~~generates list of cities~~

~~shuffle list~~

hill climber (accept T as a parameter - T = 0 is basic hill climber)

~~- will keep track of how many times its attempted an operator and not found a better tour(once a better tour is found, we can reset this to 0) will terminate at a set maximum~~

~~- cooling schedule as parameter (multiply by .8 every 200 iterations, for example)~~

~~-for each operator applied keep track of E(s') - E(s), so that the average cost of a bad move can be tracked~~

~~this can be used to find a good initial T by running the basic hill climber with this~~

-~~need to keep track of the number of permutations attempted~~

~~OPERATOR 1: swap two random cities operator~~

~~OPERATOR 2: reverse a sub sequence~~

~~cost function (Euclidean distance) E(s)~~

~~accept with probability~~

~~probability function~~

dynamically visualize tour as it changes

(probably want one screen to display current & one to just display the best found so far)

~~B & B algorithm - to know what the best tour actually is~~

GRAPHS:

plot for 10 random problems on each of the 4 algorithms (1 graph for each algorithm, 10 data points) - each of the 4 need to be run against the same 10

X: the number of states explored

Y: the terminal solution quality

best solution found / optimal solution, found by B&B \* 100

graph 1 generator

foreach 10 random solutions

run the 4 complete solution aglorithms, and the B &B

foreach of the 4:

X - # of states explored by it, y is the terminal solution equality

graph 2 generator : same as 1 except this time only using best simulated annealing with best operator

try 4 different cooling schedules