- Local search operators to try out
    - Need one general operator but it should generate reasonably sized moveset
    - Is it required that every tour in the moveset is a better solution than the current tour?
    - Why do we need a set of states as moveset? A single state to move next would suffice right?
        - Use the operation specified here http://toddwschneider.com/posts/traveling-salesman-with-simulated-annealing-r-and-shiny/ and generate a moveset of fixed size
        - Just realized that its better to use an operator that with greater probability lead to better choices, but can also generate bad ones
            - Like choosing the longest 2 edges and crossing them up
        - Generating an initial move to start with ?
            - Start with city A and move to the nearest unvisited city of the current tour. This is greedy approach
- When to stop simulated annealing ?
    - When after a certain number of steps, and when you stop seeing improvement in the solution
    - No need of random restart for this assignment
- Cooling strategies
    - Exponential schedule
        - $T (t) = T0\alpha t$ where $0 < \alpha < 1$
    - Linear schedule
        - $T (t) = T0 - \eta t$
    - Logarithmic schedule
        -
- How do we implement ‘selecting with probability P’?
- Resources
    - http://toddwschneider.com/posts/traveling-salesman-with-simulated-annealing-r-and-shiny/
    - https://github.com/chncyhn/simulated-annealing-tsp
    - For cooling strategy http://iopscience.iop.org/article/10.1088/0305-4470/31/41/011/pdf
    -