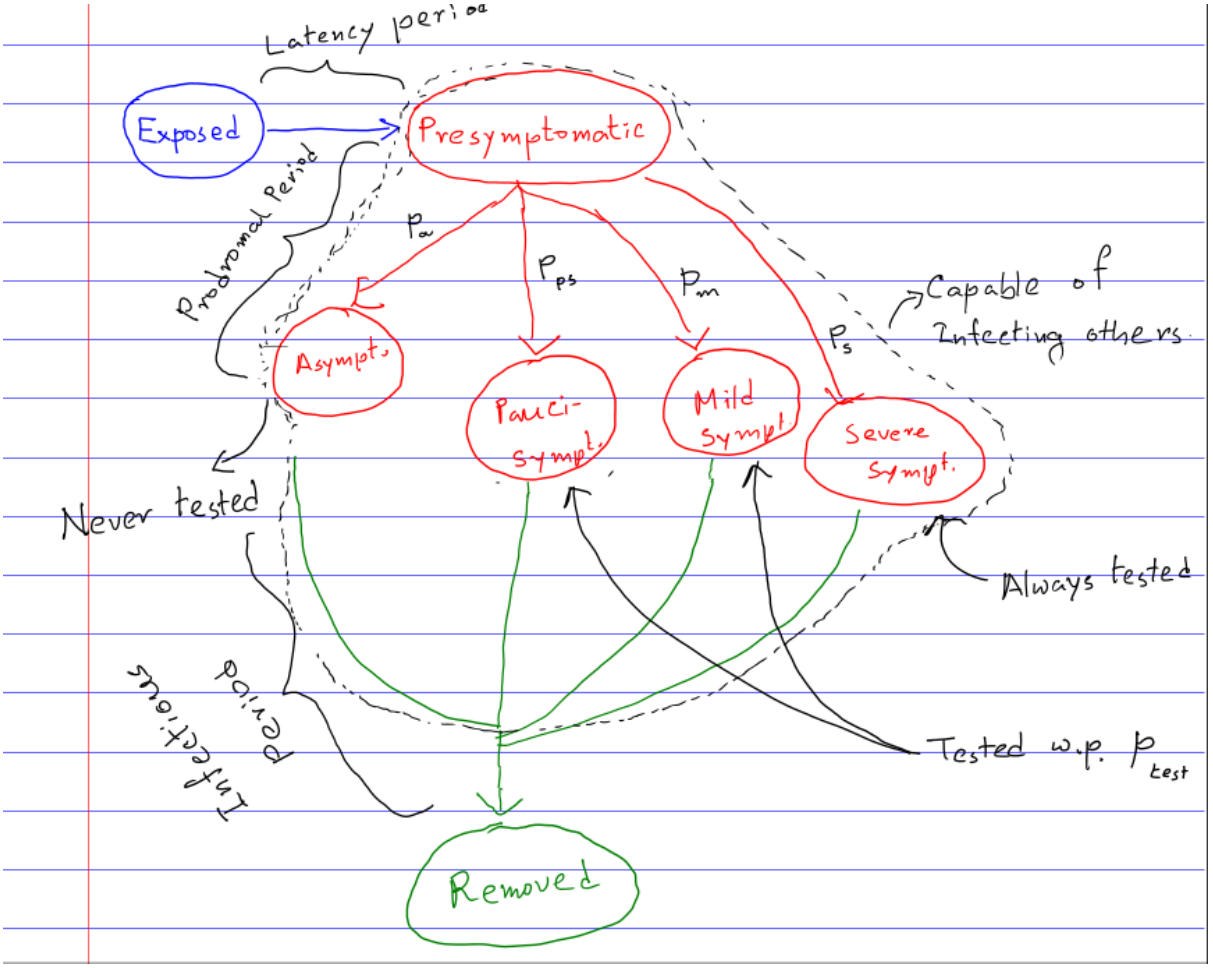
Parameters:

1. one day in seconds, `day=24*60*60`
2. time steps in data are 300 sec each, `timestep_in_data=300.0`
3. probability of using a tracking smartphone app, `p_app_d=0.00`
4. probability of getting tested if with mild symptoms (asymptomatic never get tested; those with severe symptoms always get), `p_tested_d=0.5`
5. probability of a contact being recalled correctly when contact tracing, set to 0 if no contact tracing, `p_traced_d=0.5`
6. probability of wearing a mask, `p_mask_d=0.0`
7. delay from onset of symptoms to getting test results, `test_delay_d=0.5*day`
8. additional delay of contact tracing before quarantining contacts, `trace_delay_manual_d=1.0*day`
9. Same as above, but for digital mode, `trace_delay_app_d=0.0`
10. total number of timeslots for contact to be considered for tracing, `manual_tracing_threshold=2`
11. Same as above, for app, `app_tracing_threshold=2`
12. how much transmission probability is reduced, if the infectious person wears a mask, `mask_reduction_out_d=0.6`
13. how much transmission probability is reduced, if the susceptible person wears a mask, `mask_reduction_in_d=0.9`
14. store contacts for this long, then remove, `tracelength_d=day*2`
15. duration of quarantine, `quarantine_length_d=day*14`
16. time from transmission to symptoms, `incubation_period=5.2*day`
17. infectiousness begins this many days before symptoms, `prodromal_period=1.5*day`
18. `latency_period=incubation_period-prodromal_period`
19. prob. of an infected individual being asymptomatic, `p_asymptomatic=0.2`
20. `p_paucisymptomatic=0.2*(1-p_asymptomatic)`
21. `p_mildsymptoms=0.7*(1-p_asymptomatic)`
22. `p_severesymptoms=0.1*(1-p_asymptomatic)`
23. `infectious_period=7.5*day-incubation_period`
24. for Ias,Ips and their pre-symptomatic period (Ip), `infectiousness_damping=0.51`
25. prob. of transmission from I to S in one timestep, `p_transmission=0.045`

Distributions of delays:

1. Time gap between exposure and beginning of presympt. stage ~ Normal(mean=`latency_period`, sd=`latency_period/10`)

2. Time gap between beginning of presympt. and beginning of symptoms(including asymptomatic) ~ Normal(mean=`prodromal_period`, sd=`prodromal_period/10`)
3. If the individual gets tested (assuming test results are always correct), then time gap between beginning of symptoms and beginning of quarantine ~ Normal(mean=`test_delay`, sd=`test_delay/10`)
4. Time gap between beginning of symptoms (i.e., ending of presympt. stage) and time for removal ~ Normal(mean=`infectious_period`, sd=`infectious_period/10`)
5. If someone gets a positive test result, his/her contacts (those who can be recalled. Each of them are recalled independently w.p. `p_traced`) are sent to quarantine after time Normal(mean=`trace_delay_manual`, sd=`trace_delay_manual/10`). If source and target both have apps (in that case, all such contacts are recalled), the delay is Normal(mean=`trace_delay_app`, sd=`trace_delay_app/10`)
6. Quarantine period is deterministic and equal to `quarantine_length`



Contact tracing: If someone is tested positive, the person tries to recall the contacts. The recalled ones are sent to quarantine (without testing).

Input file:

CSV file with 4 columns: timestamp (in seconds. multiple of 300), Person 1, Person 2, Contact strength. So if there is a row: `2700 16 23 10.0`

It implies that individuals with id 16 and 23 had contact with each other during the period 45m00s-49m59s, and contact strength value was 10.0. As of now, we are not using the strength value, but it may be used to characterize high-risk and low-risk contacts.

Questions:

1. What are the standard distributions and parameters of the disease? Is there a way to estimate those from our data?
2. Do we have to introduce age-specific parameters? For example, age-specific probability of developing symptoms, transmission prob. etc.
3. As of now, I am using a fixed input file obtained from the internet and it contains only 692 individuals. How do we generate contact events for arbitrary population size? (For example, in the science paper, we have seen that similar age-pairs have higher chances of transmission. How to simulate contact data so that these patterns still remain?)
4. What is the exact procedure of testing in the Indian context? What are their probabilities of producing correct test results?
5. Scalability:
    a. I have a feeling that in the worst case scenario, the time complexity is O(`p_transmission`*E*N), where E=total number of contact events and N=total number of individuals. (This is the worst case complexity. Otherwise, it is possibly O(`p_transmission`*E*C), where C denotes the expected/mean number of contacts per individual occurred in a timespan of `trace_length`). If we can fix some algorithm for generating contact events for a population of size 100K or 1M, we can try to run the code for that size. But I am not very optimistic about the performance of python. Is it a good idea to convert everything to C++?
    b. There are some tools for simulating epidemics: See here (page 6) for a comparison of 5 existing methods. Also see FLuTE , source code here, written in C++. But quite long, complex. In terms of algorithms, there is not much room for improvement, since we want to preserve individual level data. Only thing to do is to parallelize the computation with a smart approach. For these existing methods, I could not find well-written documentation (only found a code for FLuTE, nothing else). So adapting these tools to our scenario may become challenging.