Assignment 4

1. Traffic is modelled as an inhomogeneous Poisson process at a crossing where vehicles arrive at rates that depend on the time of the day. The rates are measured averages over time-intervals as follows:

Time-Interval	$\lambda(t)$ **(cars per minute)**
00:00 – 06:00	0.2
06:00 – 08:00	0.9
08:00 – 12:00	0.5
12:00 – 15:00	0.6
15:00 – 17:00	1.0
17:00 – 19:00	0.6
19:00 – 24:00	0.5

Implement the two methods that will give the event times in order: **the revised method and the thinning approach**. **Simulate the described arrival process using these methods.**

Using both methods, simulate 1000 Poisson processes. Present the results separately, first for the revised method under **1. a)** and second for the thinning approach under **1. b**).
(i) (w=2) For each such simulation, **plot the average number of cars** that arrived at (and obviously left) the crossing **per minute over the period of 24 hours.**
(ii) (w=2) **Also plot the cumulative number of cars that arrived per minute over the period of 24 hours**. As numerical results,
(iii) (w=1) **print the average number of cars that arrived in one day** and
(iv) (w=1) **the average time when the first car arrived**.

By the above plots and numbers you can check that your algorithms simulating the inhomogeneous Poisson process work right, since you know what you should get.

Please note that in the revised algorithm you need to sample the different numbers of events for the different realisations of the Poisson process (the first row in the pseudo algorithm:

$m \sim Po(U_\lambda T)$. You can use your own algorithm for generating numbers from the Poisson distribution or use the library function numpy.random.poisson

2. In Assignment 3 you implemented and simulated homogeneous and inhomogeneous Poisson process of radioactive decay by applying inversion distribution method to inter-event time distribution. Here you will use the Poisson distribution for the number of events. Implement both processes by drawing samples from a Poisson distribution to obtain number of events in appropriate time intervals. Use $\Delta t = 0.01$ as the magnitude of one time step. So, elapsed time = steps × Δt. The number of nuclei is initially $N_0 = 10\ 000$. The decay (fission) rate for one nucleus is $\lambda_c = 0.3$ per second. (Note that this λ_c value is different from what was used in Assignment 3.)

a) Consider an ensemble of $N_0 = 10\ 000$ nuclei, where the number of undecayed nuclei is kept constant; in other words, every time a nucleus decays it is replaced by an undecayed nucleus. The differential equation for the process is $\frac{dN}{dt} = -\lambda_c N_0$. Accordingly, the constant decay rate is $\lambda = \lambda_c N_0$. Draw samples from the Poisson distribution by using numpy.random.poisson. This is the number of events in a time interval.
   (i) (w=2) Implement the algorithm. Indicate clearly the different parts and functions.
   (ii) (w=2) Compute mean values by averaging over 1000 simulations of the following: The half time, i.e., the time it takes for $N_0/2$ nuclei to decay, and the variance of this half time.

b) Next, consider an ensemble of $N_0 = 10\ 000$ nuclei, where decayed nuclei are not replaced. The process becomes inhomogeneous as the decay rate will depend on time. The differential equation describing this process reads as $\frac{dN}{dt} = -\lambda_c N(t)$, where the number of undecayed nuclei decreases exponentially, $N(t) = N_0 e^{-\lambda_c t}$. The inter-event times are different from the homogeneous case in a). Accordingly, you need to sample number of events in appropriate time interval using numpy.random.poisson.
   (i) (w=2) Implement the algorithm. Indicate clearly the different parts and functions.
   (ii) (w=2) Compute mean values by averaging over 1000 simulations of the following: The half time, i.e., the time it takes for $N_0/2$ nuclei to decay, and the variance of this half time.

Note. Above, $N = \langle N \rangle$, i.e. an ensemble average. Likewise, $N(t) = \langle N(t) \rangle$, i.e. an ensemble average at time t.