

Συστήματα Αναμονής

Εργαστηριακές Ασκήσεις

1η Ομάδα Ασκήσεων

8 ^ο Εξάμηνο Η.Μ.Μ.Υ.	Αγγλογάλλος Αναστάσιος	031 18641
2021 - 2022		

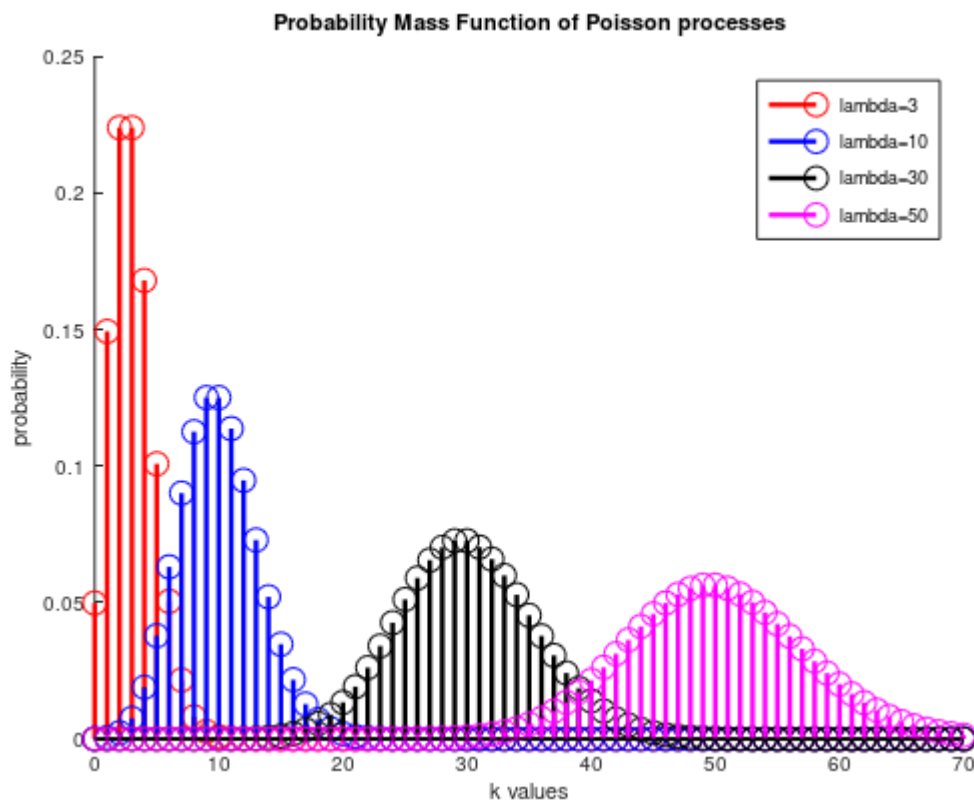
Κατανομή Poisson

Ερώτημα 1.A

Κώδικας Ερωτήματος :

```
# -> Question 1.A <-  
  
k = 0:1:70;  
lambda = [3, 10, 30, 50];  
  
for i = 1 : columns(lambda)  
    poisson(i, :) = poisspdf(k, lambda(i));  
endfor  
  
colors = "rbkm";  
figure(1);  
hold on;  
for i = 1 : columns(lambda)  
    stem(k, poisson(i, :), colors(i), "linewidth", 1.2);  
endfor  
hold off;  
  
title("Probability Mass Function of Poisson processes");  
xlabel("k values");  
ylabel("probability");  
legend("lambda=3", "lambda=10", "lambda=30", "lambda=50");
```

Παραγόμενο Διάγραμμα :



Γνωρίζουμε από την θεωρία ότι για την κατανομή Poisson ισχύει ότι η μέση τιμή της, καθώς και η διακύμανσή της ισούται με λ :

$$(\text{Var}[X] = E[X] = \lambda)$$

Παρατηρούμε ότι καθώς το λ αυξάνεται είναι αναμενόμενο η συνάρτηση να «απλώνει» και το σημείο μέσης τιμής να μετατοπίζεται σε μεγαλύτερες τιμές. Επιπλέον, καθώς το άθροισμα όλων των επιμέρους τιμών πρέπει να ισούται με τη μονάδα (πιθανότητες), ελαττώνεται και η μέγιστη τιμή της συνάρτησης.

Ερώτημα 1.B

Κώδικας Ερωτήματος :

```
# -> Question 1.B <-  
  
index = find(lambda == 30);  
chosen = poisson(index, :);  
mean_value = 0;  
for i=0:(columns(poisson(index, :)) - 1)  
    mean_value = mean_value + i .* poisson(index,i+1);  
endfor  
  
display("mean value of Poisson with lambda 30 is");  
display(mean_value);  
  
second_moment = 0;  
for i = 0 : (columns(poisson(index, :)) - 1)  
    second_moment = second_moment + i .* i .* poisson(index, i + 1);  
endfor  
  
variance = second_moment - mean_value .^ 2;  
display("Variance of Poisson with lambda 30 is");  
display(variance);
```

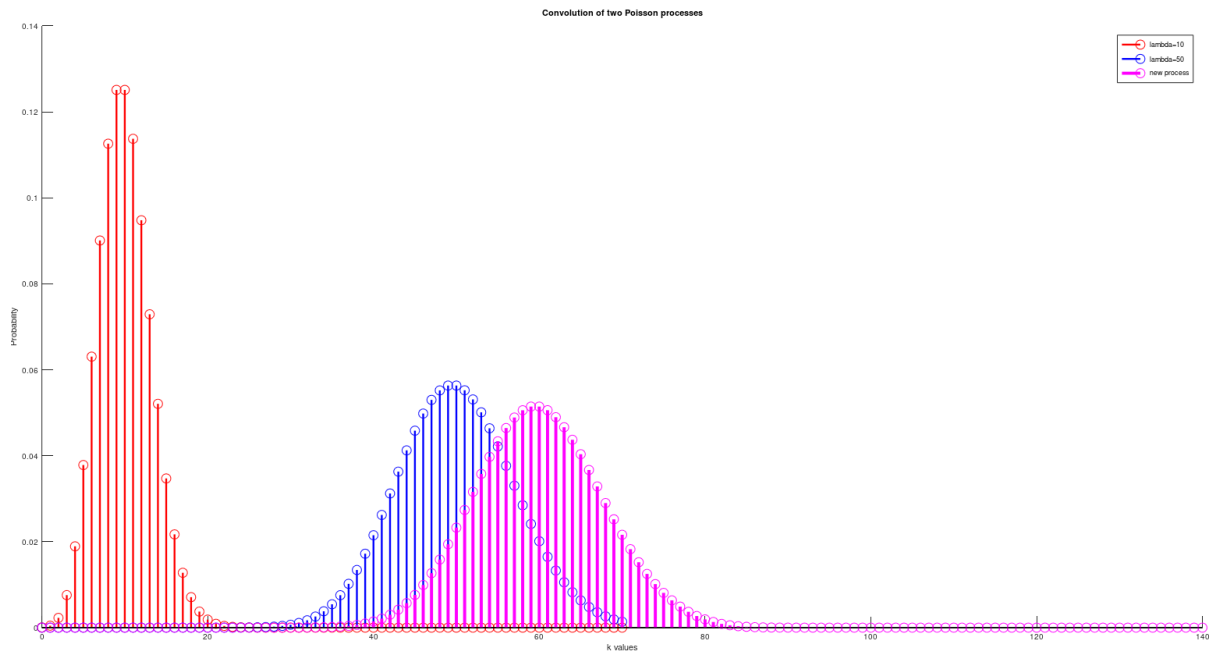
Αποτέλεσμα (Output) :

```
mean value of Poisson with lambda 30 is  
mean_value = 30.000  
Variance of Poisson with lambda 30 is  
variance = 30.000
```

Παρατηρούμε ότι για την κατανομή Poisson με παράμετρο $\lambda=30$, υπολογίζεται ότι $E[X] = 30$ και $\text{Var}[X] = 30$. Όπως αναμενόταν οι δύο τιμές αυτές είναι ίσες.

Ερώτημα 1.Γ

Με την υπέρθεση των ζητούμενων κατανομών προκύπτει οι εξής κατανομή (χρώμα : μωβ , όνομα : 'new process'):



Κώδικας Ερωτήματος :

```
first = find(lambda == 10);
second = find(lambda == 50);
poisson_first = poisson(first, :);
poisson_second = poisson(second, :);

composed = conv(poisson_first, poisson_second);
new_k = 0 : 1 : (2 * 70);

figure(2);
hold on;
stem(k, poisson_first(:), colors(1), "linewidth", 1.2);
stem(k, poisson_second(:), colors(2), "linewidth", 1.2);
stem(new_k, composed, "mo", "linewidth", 2);
hold off;
title("Convolution of two Poisson processes");
xlabel("k values");
ylabel("Probability");
legend("lambda=10", "lambda=50", "new process");
```

Παρατήρηση :

Η καινούρια κατανομή φαίνεται ότι είναι και αυτή κατανομή Poisson. Ωστόσο, όντας η συνέλιξη των δύο προηγούμενων είναι διπλάσια σε μήκος, ενώ έχει παράμετρο $\lambda = \lambda_1 + \lambda_2 = 10 + 50 = 60$.

$$\bullet \sum_{i=1}^n \text{Poisson}(\lambda_i) \sim \text{Poisson} \left(\sum_{i=1}^n \lambda_i \right) \quad \lambda_i > 0$$

Γενικότερα ισχύει ότι :

Απαραίτητη προϋπόθεση για να συμβαίνει αυτό για τη συνέλιξη δύο ή περισσότερων συναρτήσεων είναι αυτές να είναι ανεξάρτητες μεταξύ τους.

Ερώτημα 1.Δ

Για να ληφθεί μία κατανομή Poisson παραμέτρου λ ως το όριο μιας διωνυμικής (binomial) κατανομής παραμέτρων n και p αρκεί να διαιρέσουμε το διάστημα t το οποίο εξετάζουμε σε αρκετά μικρά n υποδιαστήματα Δt με τέτοιο τρόπο ώστε σε ένα υποδιάστημα να μπορεί να εμφανιστεί μόνο ένα (ή κανένα) γεγονός.

Με αυτόν τον τρόπο υπάρχουν δύο περιπτώσεις:

Υπαρξη εμφάνισης γεγονότος και μη-ύπαρξη εμφάνισης γεγονότος.

Έτσι, πραγματοποιώντας n ανεξάρτητες δοκιμές Bernoulli (μία σε κάθε υποδιάστημα), η ύπαρξη έχει πιθανότητα $p = \lambda \Delta t$ και η μη-ύπαρξη $1 - p$. Η πιθανότητα k επιτυχιών, λοιπόν σε n ανεξάρτητες δοκιμές δίνεται από τη Διωνυμική Κατανομή:

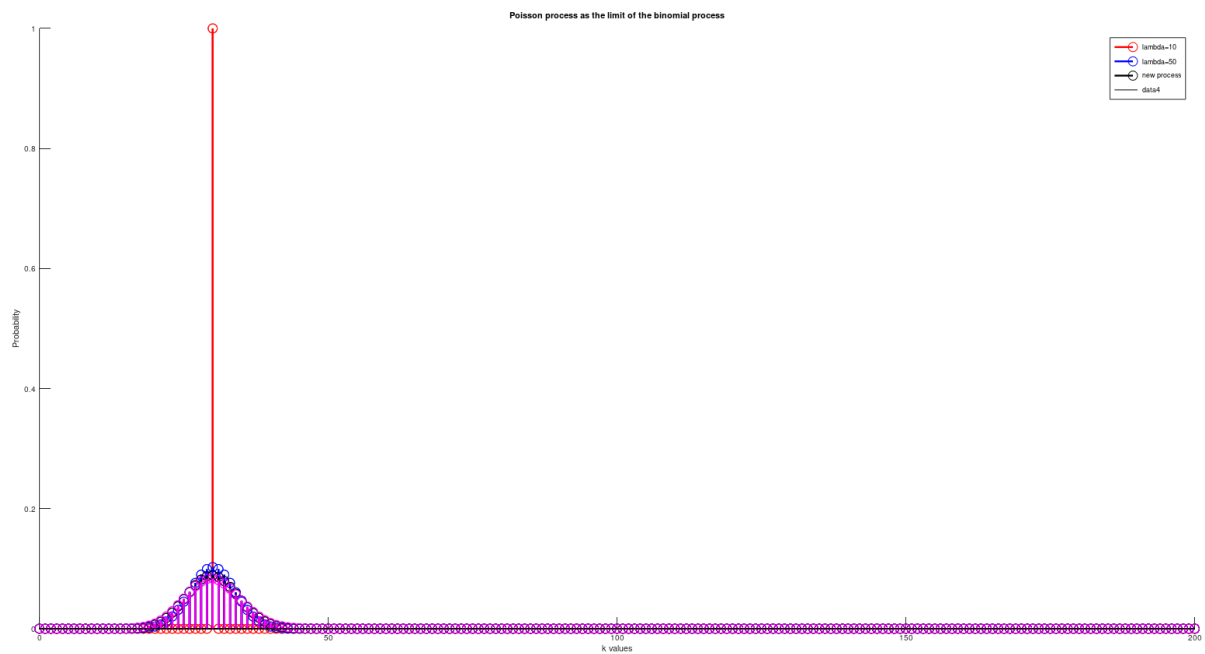
$$P[N(t) = k] = \binom{n}{k} p^k (1 - p)^{n-k}, \quad k = 0, 1, \dots, n$$

$$P[N(t) = k] = \binom{n}{k} (\lambda \Delta t)^k (1 - \lambda \Delta t)^{n-k} = \binom{n}{k} \left(\frac{\lambda t}{n} \right)^k \left(1 - \frac{\lambda t}{n} \right)^{n-k}$$

Στο όριο $\Delta t \rightarrow 0$, $n \rightarrow \infty$, $t = n \Delta t$ έχουμε $\frac{n!}{(n-k)!} \rightarrow n^k$, $\left(1 - \frac{\lambda t}{n} \right)^{n-k} \rightarrow e^{-\lambda t}$ και

$$P[N(t) = k] = \frac{n!}{k! (n-k)!} \left(\frac{\lambda t}{n} \right)^k \left(1 - \frac{\lambda t}{n} \right)^{n-k} \rightarrow \frac{(\lambda t)^k}{k!} e^{-\lambda t}$$

Παραγόμενο διάγραμμα :



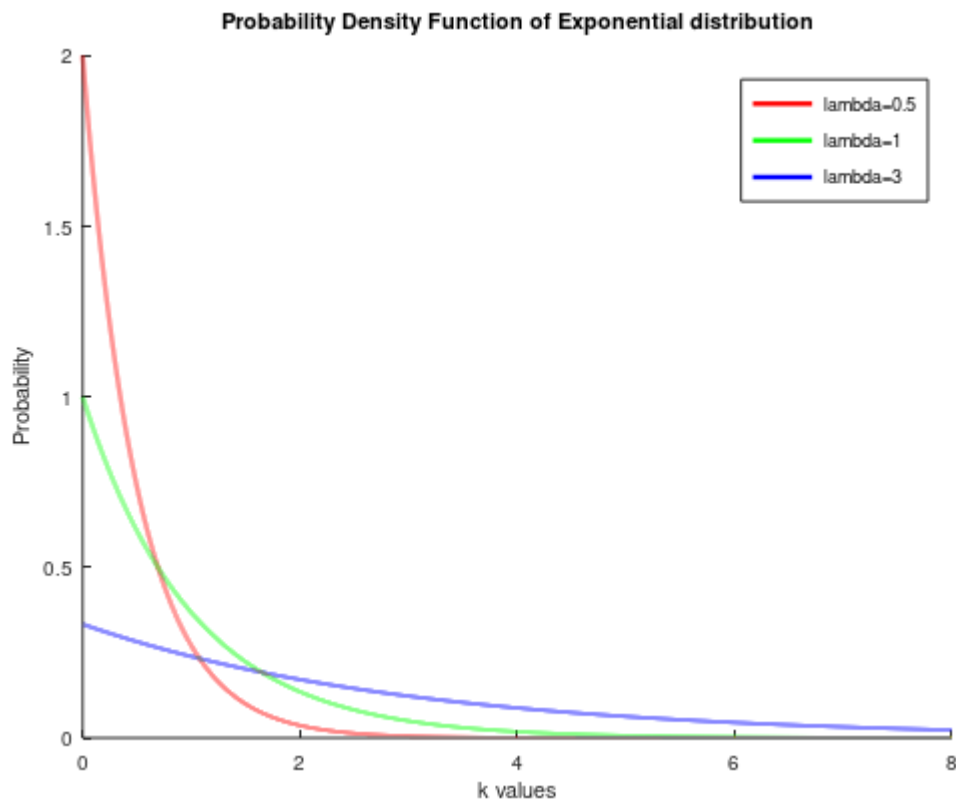
Κώδικας Ερωτήματος :

```
# -> Question 1.D <-  
  
# TASK: show that Poisson process is the limit of the binomial distribution.  
k = 0 : 1 : 200;  
# Define the desired Poisson Process  
lambda = 30;  
i = 1 : 1 : 5;  
n = lambda .* i;  
p = lambda ./ n;  
  
figure(3);  
title("Poisson process as the limit of the binomial process");  
xlabel("k values");  
ylabel("Probability");  
hold on;  
for i = 1 : 4  
    binomial = binopdf(k, n(i), p(i));  
    stem(k, binomial, colors(i), 'linewidth', 1.2);  
endfor  
hold off;
```

Εκθετική Κατανομή

Ερώτημα 2.Α

Στο παρακάτω σχήμα φαίνονται οι ζητούμενες συναρτήσεις πυκνότητας πιθανότητας για τις εκθετικές κατανομές με παραμέτρους $\lambda=0.5$ (κόκκινο), $\lambda=1$ (πράσινο) και $\lambda=3$ (μπλε):



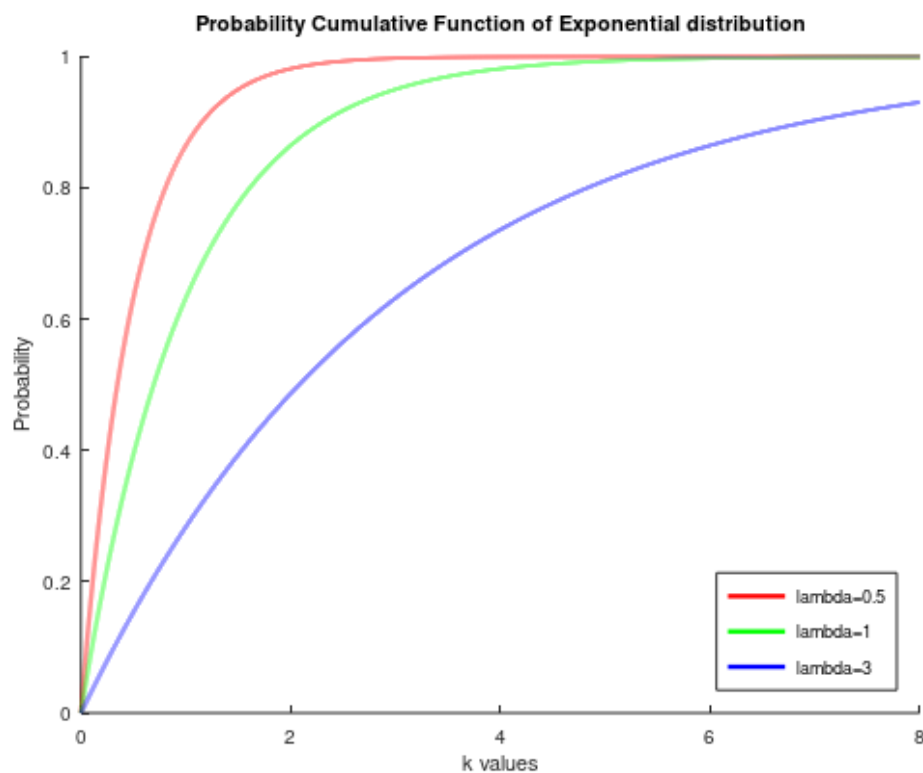
Σημείωση: Το πρόγραμμα Octave δεν ανταποκρίνονταν καθώς ο υπολογιστής δεν μπορούσε να επεξεργαστεί το μέγεθος του διανύσματος $k = 0:0.00001:8$, έτσι όπως υποδείχθηκε χρησιμοποιήσα στην άσκηση το διάνυσμα $k = 0:0.0001:8$

Κώδικας Ερωτήματος :

```
# -> Question 2.A <-  
  
k = 0:0.0001:8;  
lambda = [0.5,1,3];  
  
for i=1:columns(lambda)  
    exp(i,:) = exppdf(k,lambda(i));  
endfor  
  
colors="rgb";  
figure(1);  
hold on;  
for i=1:columns(lambda)  
    plot(k,exp(i,:),colors(i),"linewidth",1.2);  
endfor  
  
hold off;  
  
title("Probability Density Function of Exponential distribution");  
xlabel("k values");  
ylabel("Probability");  
legend("lambda=0.5","lambda=1","lambda=3");
```

Ερώτηση 2.B

Παραγόμενο διάγραμμα :



Θα χρησιμοποιήσουμε την συνάρτηση `expcdf()`, ώστε να σχεδιάσουμε την αθροιστική συνάρτηση κατανομής των εκθετικών κατανομών του προηγούμενου ερωτήματος σε ένα κοινό διάγραμμα.

Κώδικας Ερωτήματος :

```
# -> Question 2.B <-

k = 0:0.0001:8;
lambda = [0.5,1,3];

for i=1:columns(lambda)
    exp(i,:) = expcdf(k,lambda(i));
endfor

colors="rgb";
figure(2);
hold on;
for i=1:columns(lambda)
    plot(k,exp(i,:),colors(i),"linewidth",1.2);
endfor

hold off;

title("Probability Cumulative Function of Exponential distribution");
xlabel("k values");
ylabel("Probability");
h = legend("lambda=0.5","lambda=1","lambda=3");
legend(h,"location","southeast");
```

Ερώτημα 2.C

Όπου $P = P(X > 30000)$ και $Pr = P(X > 50000 | X > 20000)$.

```
P = 0.8869
Pr = 0.8869
```

Παρατηρούμε ότι οι πιθανότητες είναι ίσες. Αυτό συμβαίνει εξαιτίας του χαρακτηριστικού έλλειψης μνήμης της εκθετικής κατανομής. Γενικότερα ισχύει για την εκθετική κατανομή ότι $P(X > t+s | X > t) = P(X > s)$, το οποίο σημαίνει ότι στην εκθετική κατανομή δεν έχει σημασία το παρελθόν για την πιθανότητα, αλλά μόνο το διάστημα.

Κώδικας Ερωτήματος :

```
# -> Question 2.C <-  
  
k = 0:0.00001:8;  
  
exp = expcdf (k,2.5);  
P = 1 - exp(30000);  
display(P);  
Pr = (1-exp(50000)) ./ (1-exp(20000));  
display(Pr);
```

Διαδικασία Καταμέτρησης Poisson

Ερώτημα 3.A

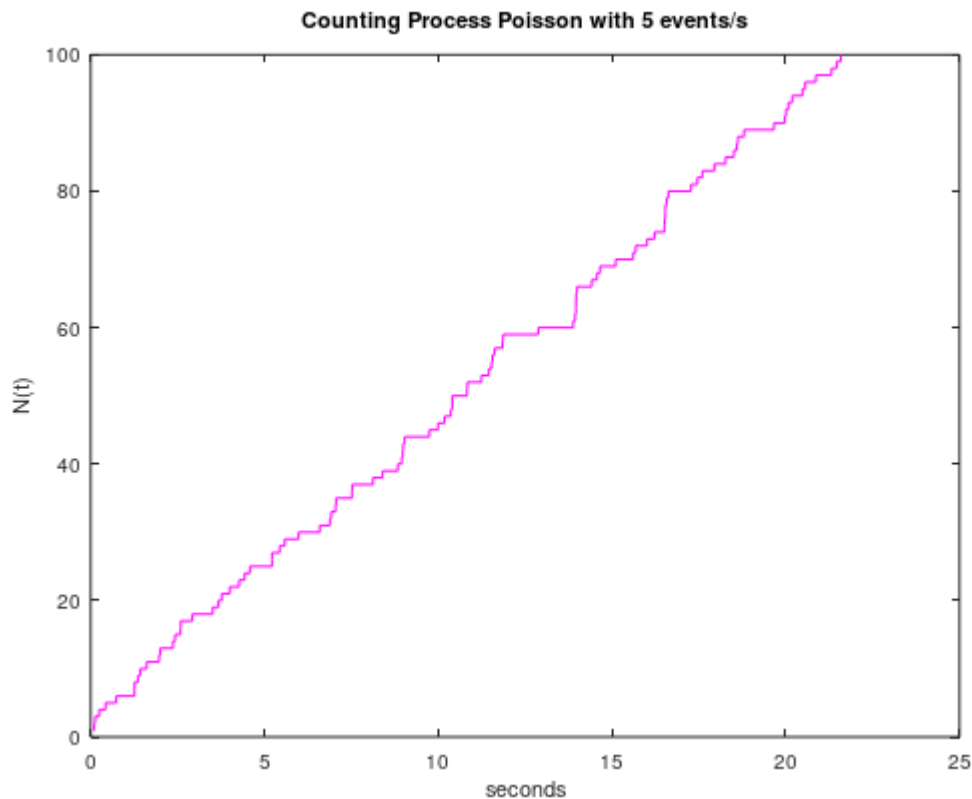
Οι χρόνοι που μεσολαβούν ανάμεσα στην εμφάνιση δύο διαδοχικών γεγονότων Poisson ακολουθούν την εκθετική κατανομή με μέση τιμή $1/\lambda$. Μπορεί να αποδειχθεί θέτοντας στην παρακάτω εξίσωση $k=1$, καθώς αναμένουμε μόνο ένα γεγονός, το επόμενο.

$$P_t[v = k] \triangleq P[N(t) = k] = \frac{(\lambda t)^k}{k!} e^{-\lambda t}, \quad k = 0, 1, 2, \dots$$

Θα χρησιμοποιήσουμε την εντολή `exprrnd()` ώστε να δημιουργήσουμε έναν πίνακα με 100 διαδοχικά τυχαία γεγονότα, τα οποία ακολουθούν την εκθετική κατανομή με μέση τιμή $1/\lambda=1/5=0.2$

Έπειτα θα αποθηκεύσουμε στον πίνακα x το άθροισμα $N(t)$ αυτών των τυχαίων γεγονότων και με την χρήση της συνάρτησης `stairs()` θα προκύψει το παρακάτω διάγραμμα.

Ακολουθεί η ζητούμενη διαδικασία καταμέτρησης Poisson:



Κώδικας Ερωτήματος :

```
# -> Question 3.A <-
x = exprnd(0.2,1,100);
y = ones(100,1);
for i=1:99
    x(i+1)=x(i+1)+x(i);
    y(i+1)=y(i+1)+y(i);
endfor
figure(1);
stairs(x,y, color= 'm');
title("Counting Process Poisson with 5 events/s");
xlabel("seconds");
ylabel("N(t)");
```

Ερώτημα 3.B

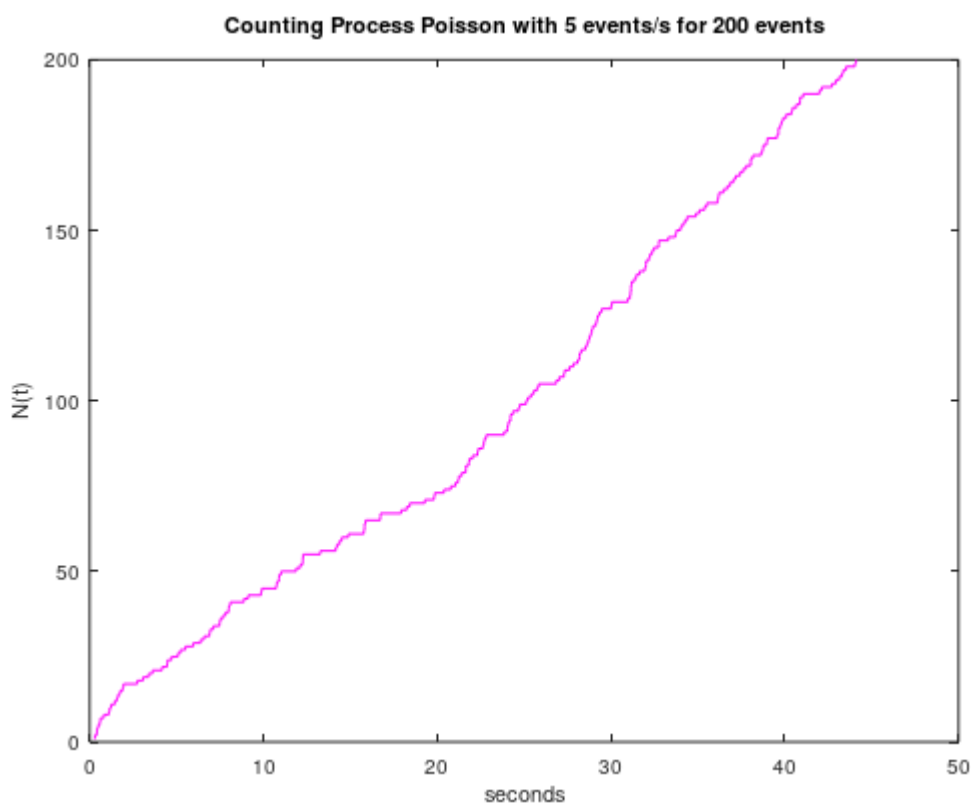
Από την θεωρία είναι γνωστό ότι ο αριθμός γεγονότων σε ένα χρονικό παράθυρο $\Delta T = t_1 - t_2$ ακολουθεί την κατανομή Poisson.

Μάλιστα, ο μέσος αριθμός γεγονότων σε αυτό το διάστημα είναι ανάλογος του διαστήματος και ίσος με $\lambda \times (t_1 - t_2)$, όπου η σταθερά λ ορίζει το τον μέσο ρυθμό εμφανίσεων. Για τους ζητούμενους αριθμούς γεγονότων υπολογίστηκαν οι εξής μέσες τιμές:

Αριθμός Γεγονότων	Μέσος Αριθμός Γεγονότων στη Μονάδα του Χρόνου
100	4,8825
200	4.5349
300	5.2313
500	5.1699
1000	5.4073
10000	5.0377

Παρατηρείται ότι όσο αυξάνουμε τα δείγματα που λαμβάνουμε από την εκθετική κατανομή με τη συνάρτηση `exprnd()` τόσο ο μέσος αριθμός γεγονότων στη μονάδα του χρόνου πλησιάζει την τιμή του λ (5 γεγονότα ανά δευτερόλεπτο).

Διάγραμμα <i> για Αριθμό Γεγονότων = 200 :



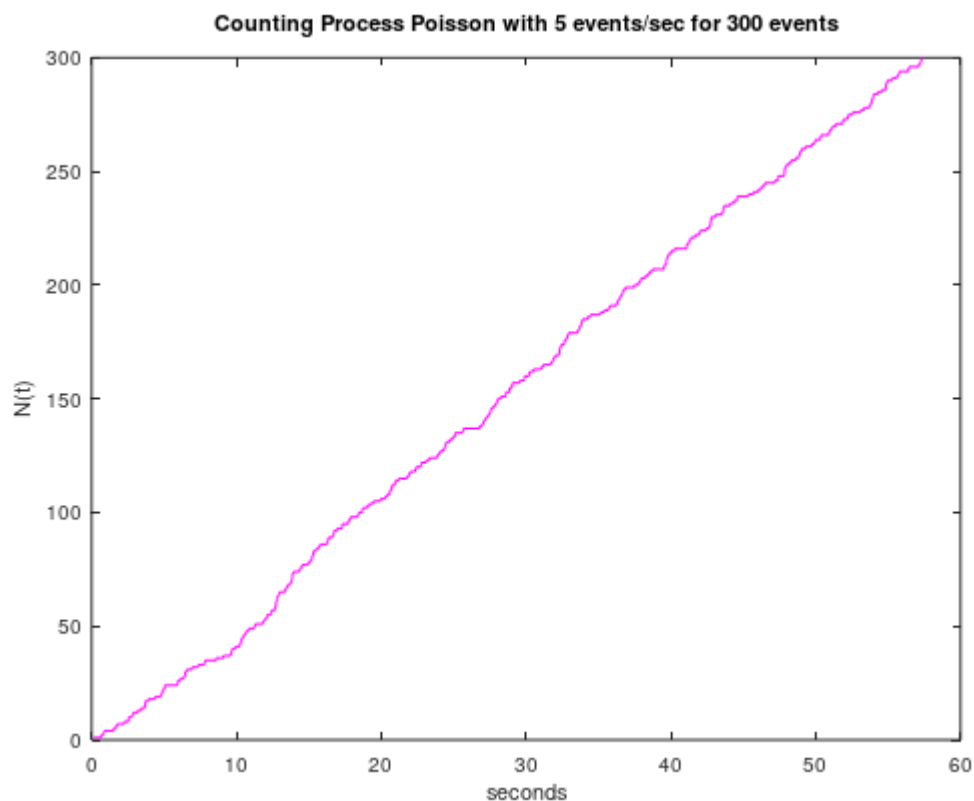
Μέσος Αριθμός Γεγονότων στην Μονάδα Χρόνου :

```
200/x(200)=  
4.5349
```

Κώδικας <i> για Αριθμό Γεγονότων = 200 :

```
# <i>  
x = exprnd(0.2,1,200);  
y = ones(200,1);  
for i=1:199  
    x(i+1) = x(i+1) + x(i);  
    y(i+1) = y(i+1) + y(i);  
endfor  
figure(2);  
stairs(x,y, color= 'm');  
title("Counting Process Poisson with 5 events/s for 200 events");  
xlabel("seconds");  
ylabel("N(t)");  
display("200/x(200)=");  
display(200/x(200));
```

Διάγραμμα <ii> για Αριθμό Γεγονότων = 300 :



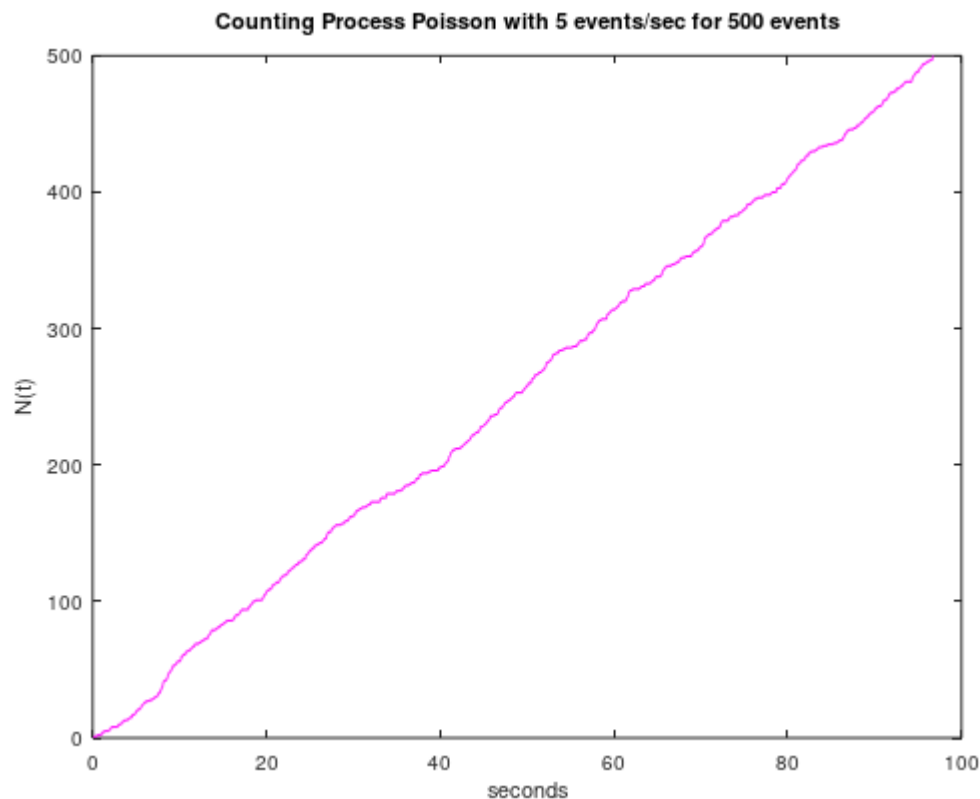
Μέσος Αριθμός Γεγονότων στην Μονάδα Χρόνου :

```
300/x(300)=  
5.2313
```

Κώδικας <ii> για Αριθμό Γεγονότων = 300 :

```
# <ii>  
x = exprnd(0.2,1,300);  
y = ones(300,1);  
for i=1:299  
    x(i+1) = x(i+1) + x(i);  
    y(i+1) = y(i+1) + y(i);  
endfor  
figure(3);  
stairs(x,y, color= 'm');  
title("Counting Process Poisson with 5 events/sec for 300 events");  
xlabel("seconds");  
ylabel("N(t)");  
display("300/x(300)=")  
display(300/x(300));
```

Διάγραμμα <iii> για Αριθμό Γεγονότων = 500 :



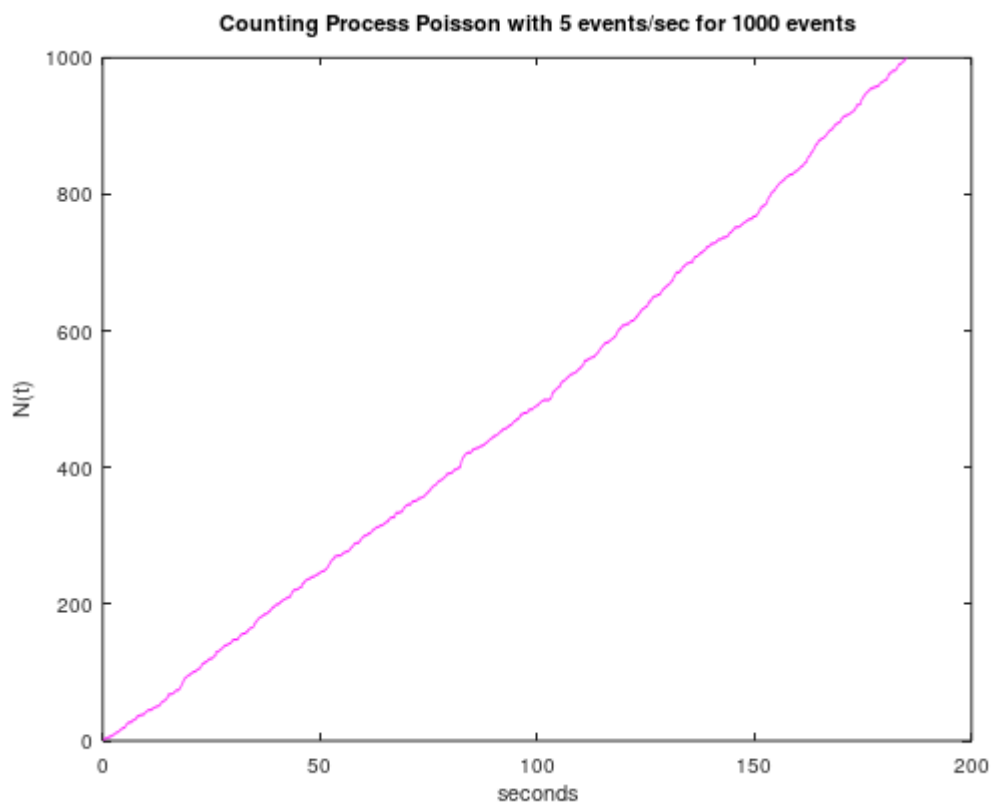
Μέσος Αριθμός Γεγονότων στην Μονάδα Χρόνου :

```
500/x(500)=  
5.1699
```

Κώδικας <iii> για Αριθμό Γεγονότων = 500 :

```
# <iii>  
x = exprnd(0.2,1,500);  
y = ones(500,1);  
for i=1:499  
    x(i+1) = x(i+1) + x(i);  
    y(i+1) = y(i+1) + y(i);  
endfor  
figure(4);  
stairs(x,y, color= 'm');  
title("Counting Process Poisson with 5 events/sec for 500 events");  
xlabel("seconds");  
ylabel("N(t)");  
display("500/x(500)=");  
display(500/x(500));
```

Διάγραμμα <iv> για Αριθμό Γεγονότων = 1000 :



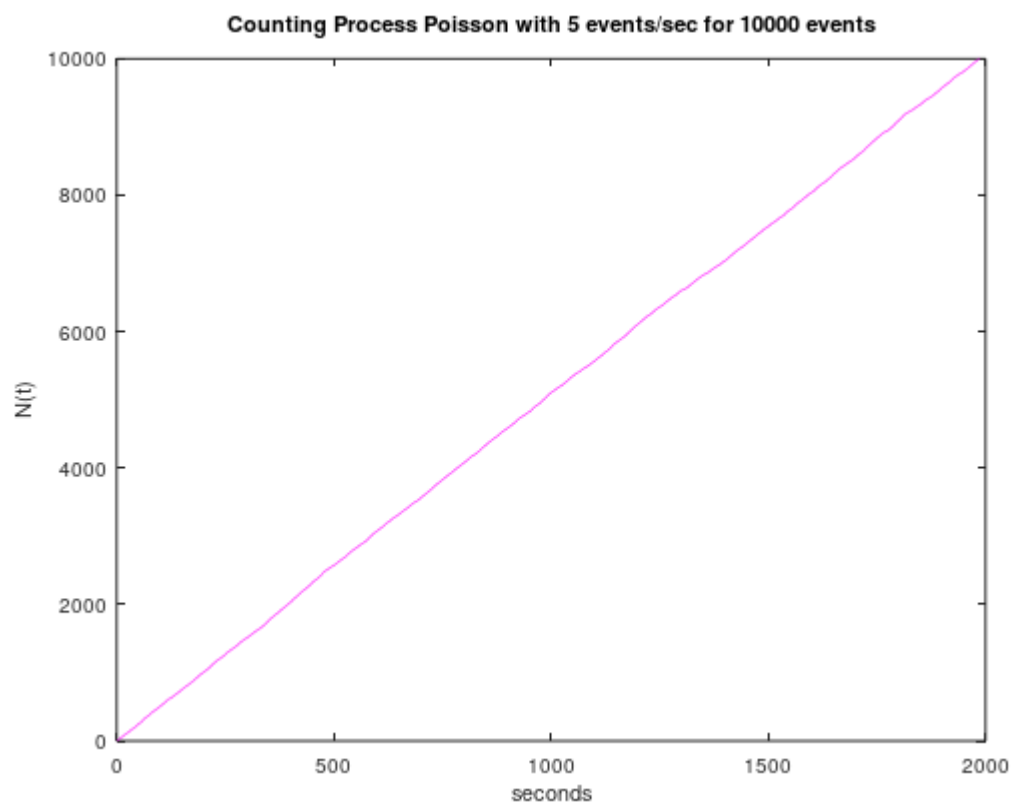
Μέσος Αριθμός Γεγονότων στην Μονάδα Χρόνου :

```
1000/x(1000)=  
5.4073
```

Κώδικας <iv> για Αριθμό Γεγονότων = 1000 :

```
# <iv>  
x = exprnd(0.2,1,1000);  
y = ones(1000,1);  
for i=1:999  
    x(i+1) = x(i+1) + x(i);  
    y(i+1) = y(i+1) + y(i);  
endfor  
figure(5);  
stairs(x,y, color= 'm');  
title("Counting Process Poisson with 5 events/sec for 1000 events");  
xlabel("seconds");  
ylabel("N(t)");  
display("1000/x(1000)=");  
display(1000/x(1000));
```

Διάγραμμα <n> για Αριθμό Γεγονότων = 10000 :



Μέσος Αριθμός Γεγονότων στην Μονάδα Χρόνου :

```
10000/x(10000)=  
5.0377
```

Κώδικας <v> για Αριθμό Γεγονότων = 10000 :

```
# <v>  
x = exprnd(0.2,1,10000);  
y = ones(10000,1);  
for i=1:9999  
    x(i+1) = x(i+1) + x(i);  
    y(i+1) = y(i+1) + y(i);  
endfor  
figure(6);  
stairs(x,y, color= 'm');  
title("Counting Process Poisson with 5 events/sec for 10000 events");  
xlabel("seconds");  
ylabel("N(t)");  
display("10000/x(10000)=");  
display(10000/x(10000));
```