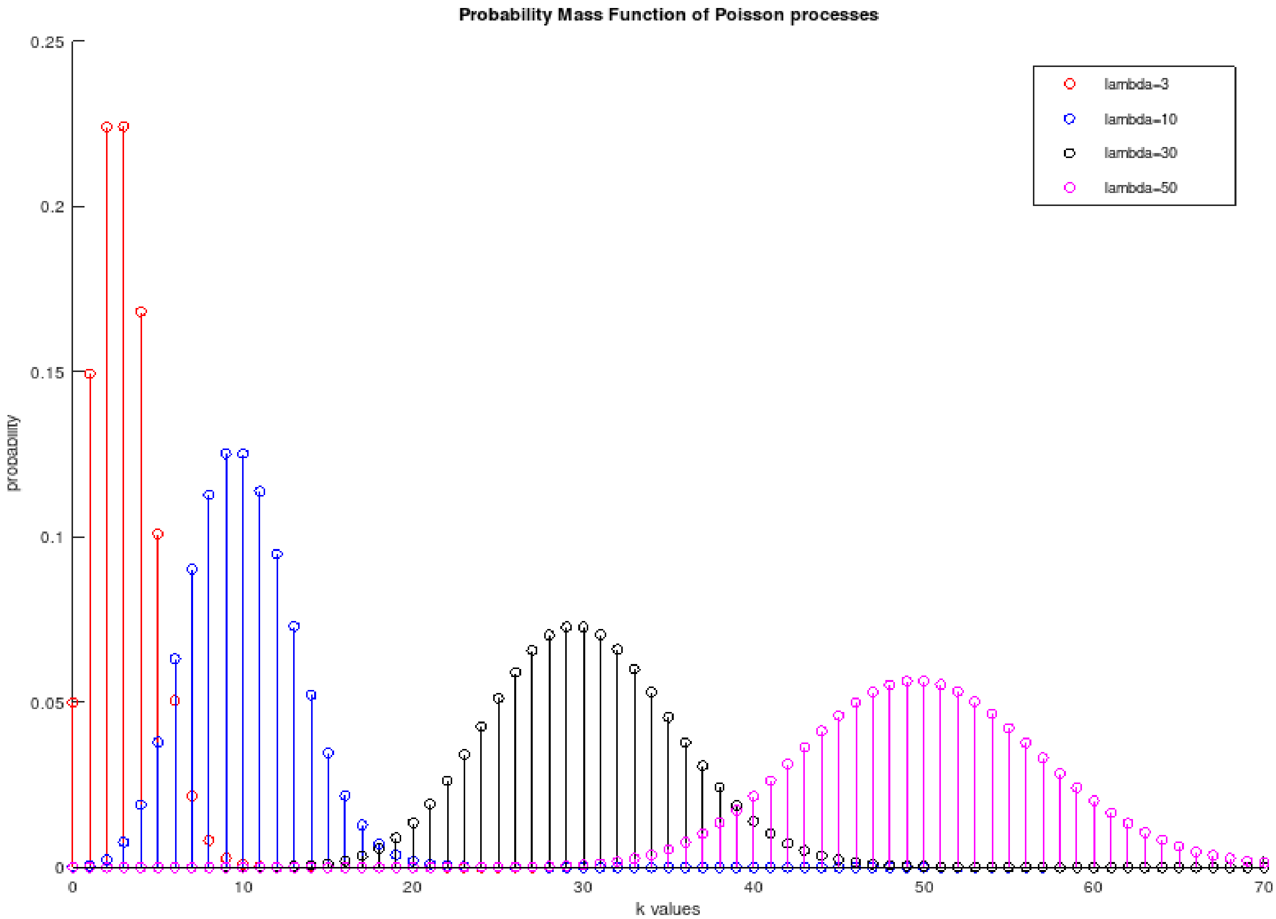
Συστήματα Αναμονής (Queuing Systems)
Πρώτη εργαστηριακή άσκηση

Όνομα: Αλεξόπουλος Ιωάννης
ΑΜ: 0317001

Κατανομή Poisson

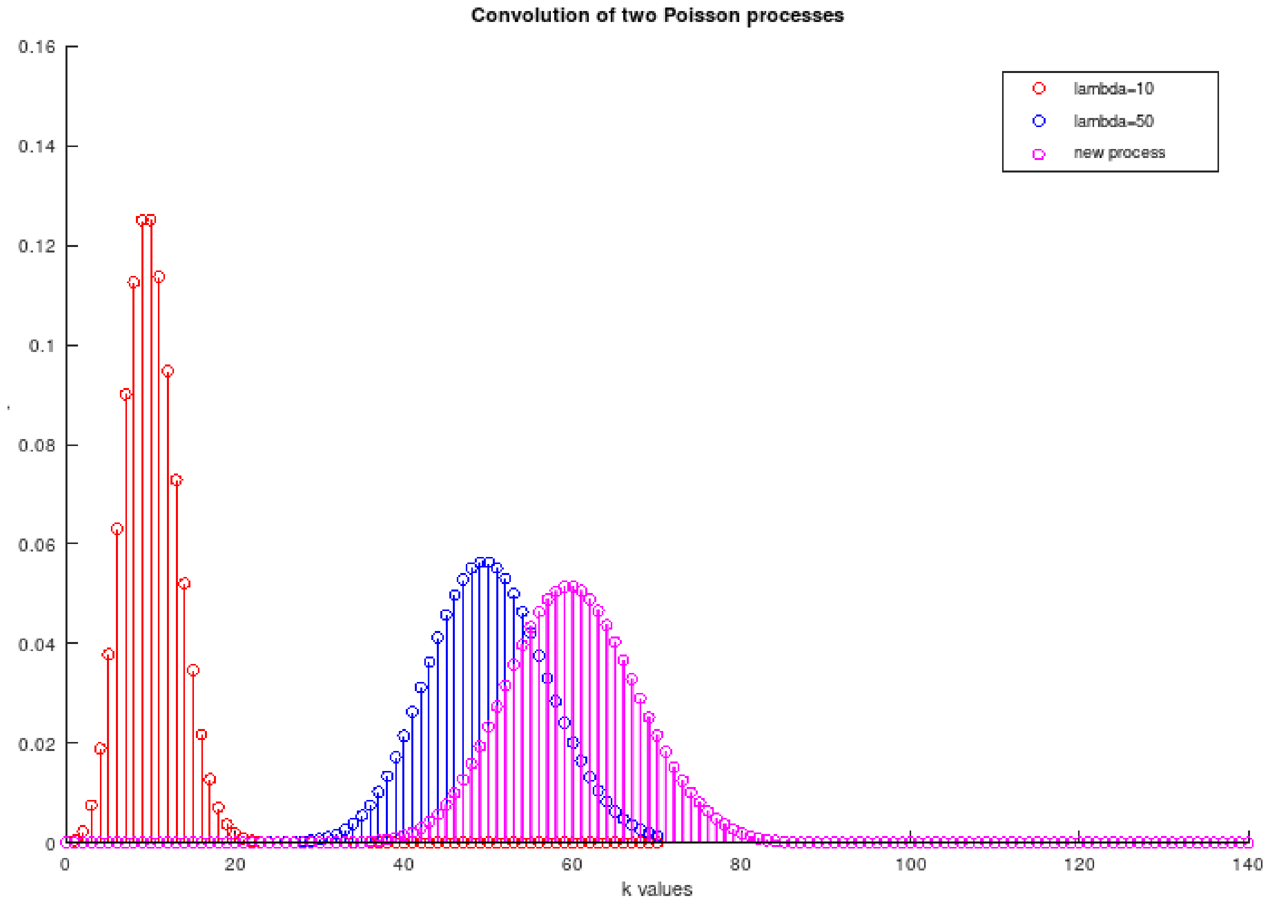
A)



Παρατηρούμε ότι καθώς αυξάνεται η τιμή της παραμέτρου λ, η συνάρτηση μάζας πιθανότητας μετατοπίζεται στον οριζόντιο άξονα ώστε να έχει μέγιστη τιμή για κ = λ. Κατά την μετατόπιση αυτή, η μέγιστη τιμή της κατανομής μειώνεται και οι τιμές καταλαμβάνουν μεγαλύτερο έυρος τιμών, δηλαδή αυξάνεται το πλάτος της συνάρτησης.

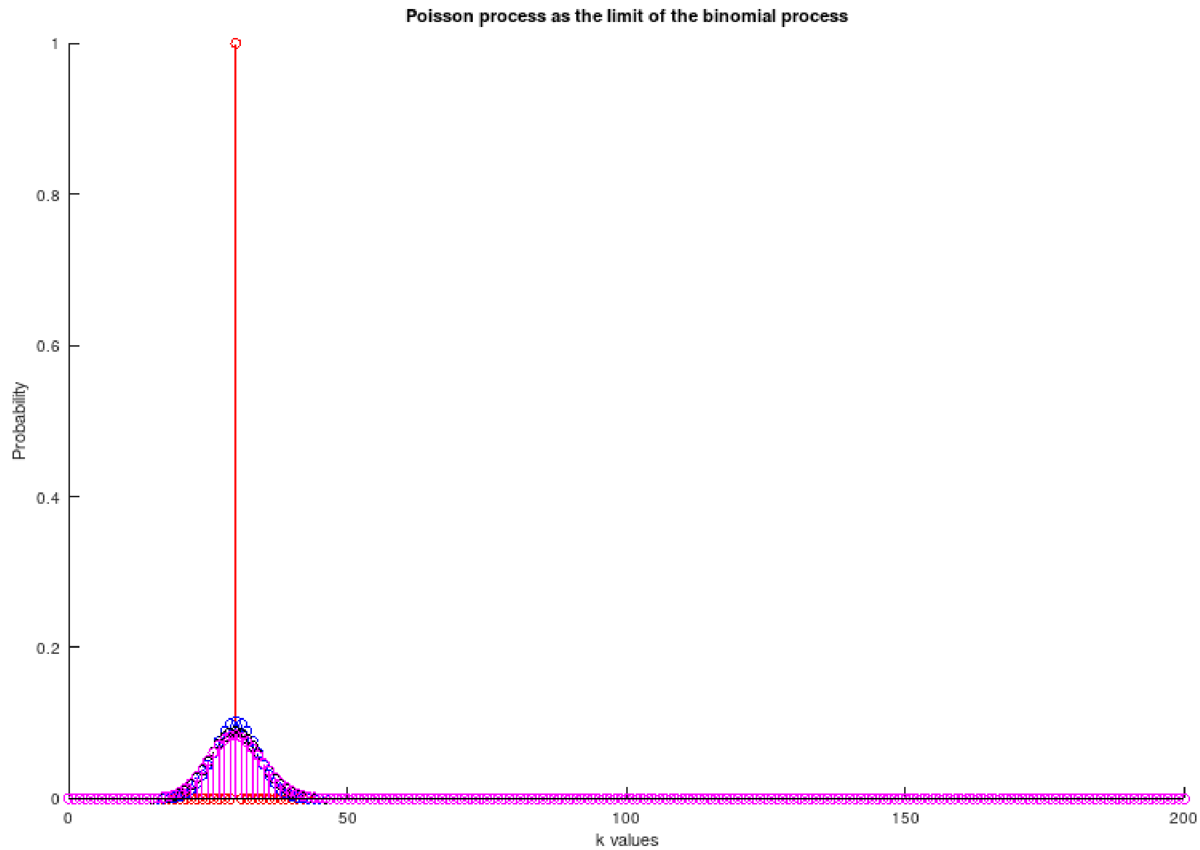
B)
Επαληθεύουμε, όπως είναι γνωστό από την θεωρία, ότι η μέση τιμή και η διακύμανση της κατανομής Poisson ταυτίζονται με την παράμετρο λ, δηλαδή στην συγκεκριμένη περίπτωση έχουν την τιμή λ=30.

Γ)



Όπως παρατηρείται στο παραπάνω διάγραμμα, η συνέλιξη των δύο κατανομών Poisson με παραμέτρους λ=10 και λ=50 αντίστοιχα, έχει ως αποτέλεσμα μια νέα κατανομή Poisson με παράμετρο που ισούται με το άθροισμα των παραμέτρων των αρχικών κατανομών, δηλαδή στην προκείμενη περίπτωση λ_{new}= 10 + 50 = 60. Απαραίτητη προϋπόθεση αποτελεί η ανεξαρτησία των τυχαίων μεταβλητών που ακολουθούν τις δύο αρχικές κατανομές Poisson.

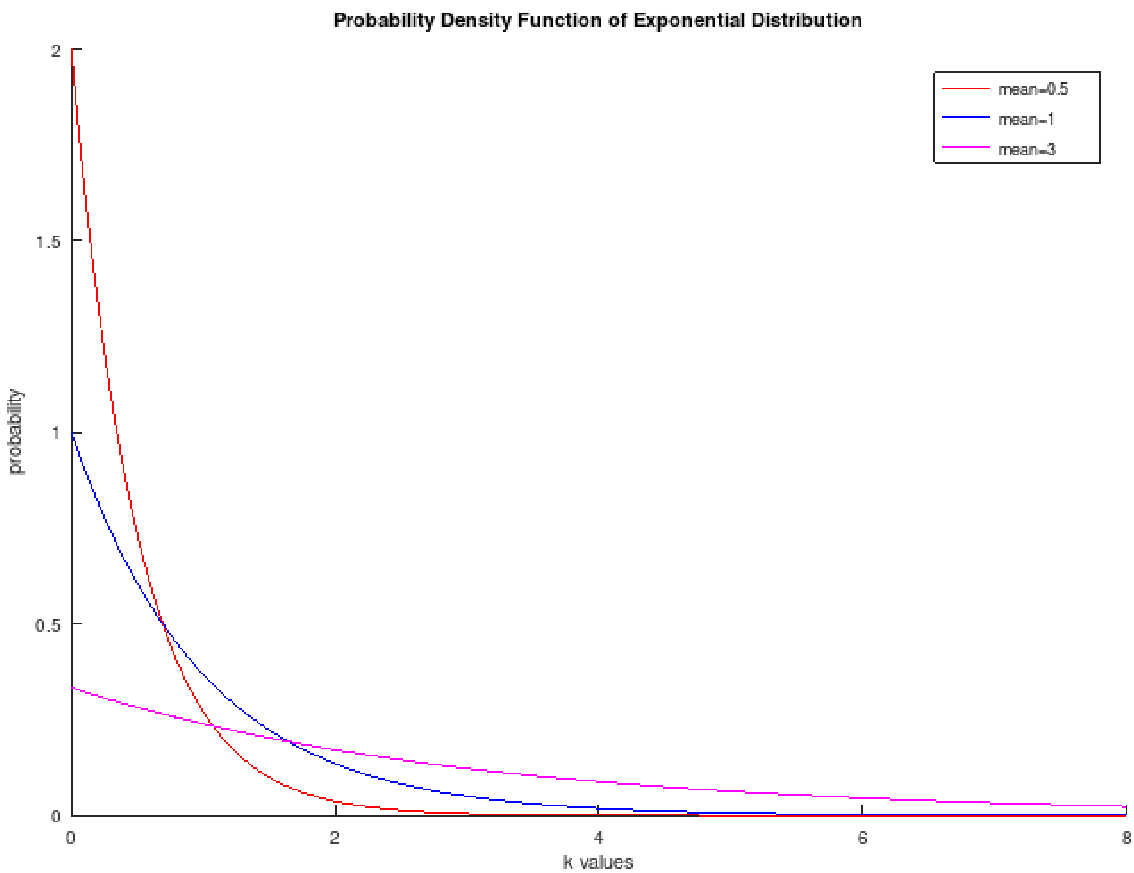
Δ)



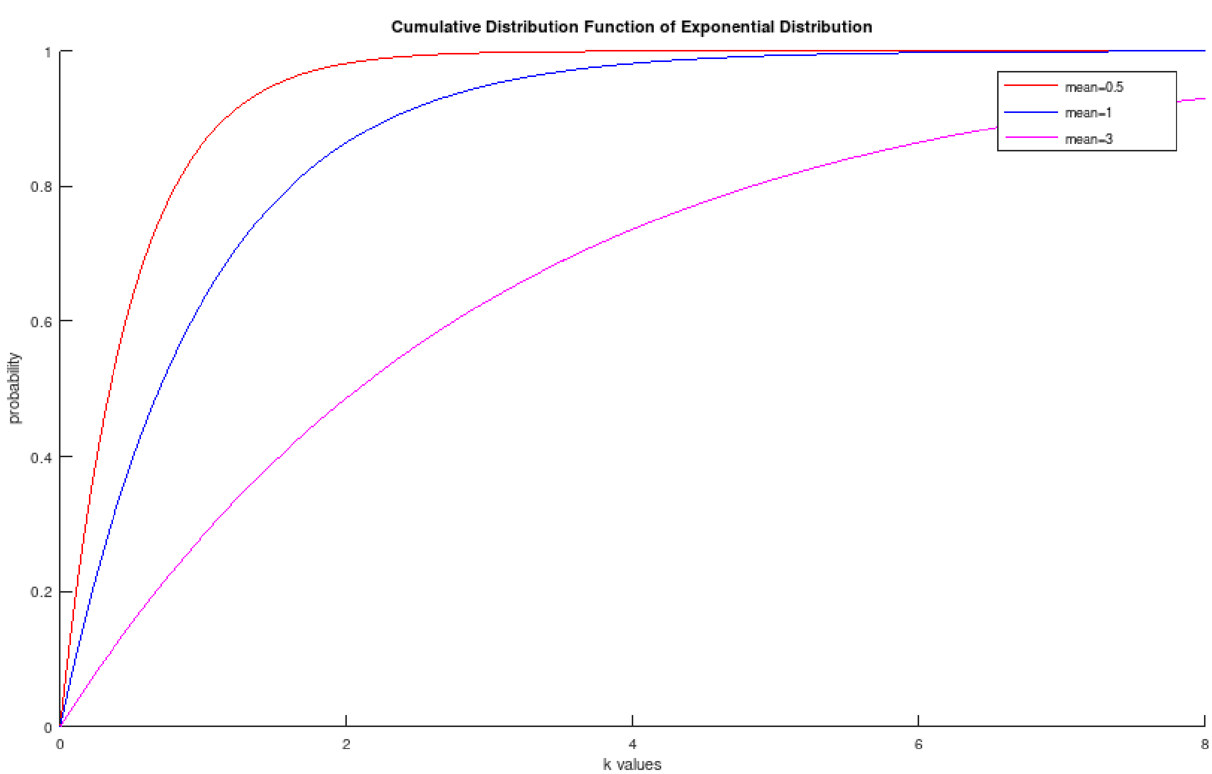
Σε αυτό το ερώτημα επαληθεύουμε ότι η κατανομή Poisson προκύπτει οριακά από την διωνυμική κατανομή όταν επιλέγουμε μεγάλο αριθμό δοκιμών n και μικρή πιθανότητα επιτυχίας p. Ρυθμίζουμε επίσης το γινόμενο np ώστε να ισούται με την παράμετρο λ της επιθυμητής κατανομής Poisson. Για μικρή τιμή του n παρατηρείται singularity αφού το p της διωνυμικής ισούται με την μονάδα ενώ με την αύξηση των δειγμάτων, παρατηρείται καλύτερη προσέγγιση της κατανομής Poisson.

Εκθετική Κατανομή

A)



B)

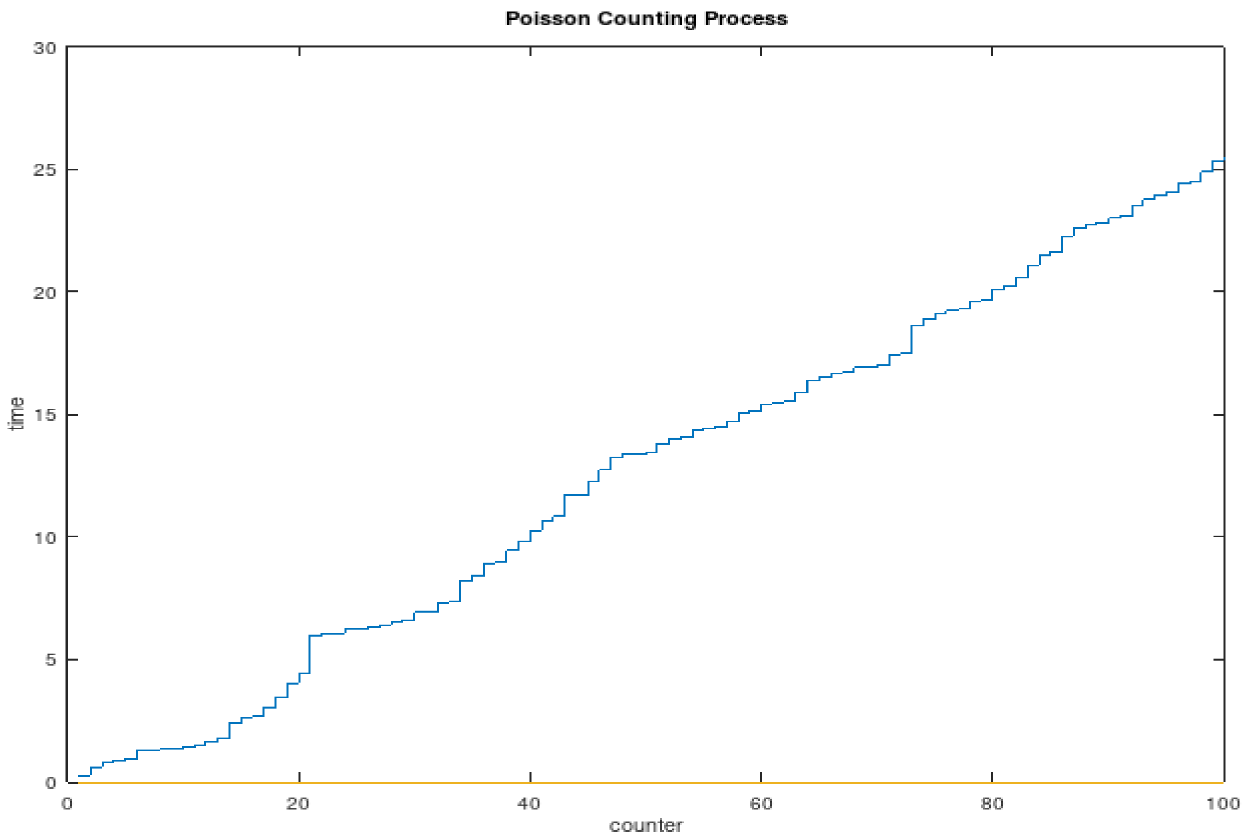


Γ) Στο ερώτημα επαληθεύεται η ιδιότητα έλλειψης μνήμης της εκθετικής κατανομής. Αποδεικνύεται δηλαδή ότι: P(X>t+s | X> t) = P(X>s) αφού οι τιμές που βρίσκουμε για τις δύο εκφράσεις είναι ίσες για s =30000, t=20000. Για την έκφραση της δεσμευμένης πιθανότητας χρησιμοποιούμε τον ορισμό: P(X>50000|X>20000) = P(X>50000 (intersection) X>20000) / P(X>20000) = P(X>50000)/P(X>20000) = (1-CDF(50000))/(1-CDF(20000))

<u>Διαδικασία Καταμέτρησης Poisson</u>

A)

Γνωρίζουμε ότι οι χρόνοι που μεσολαβούν ανάμεσα στην εμφάνιση δύο διαδοχικών γεγονότων Poisson ακολουθούν την εκθετική κατανομή. Εδώ για λ= 5 γεγονότα/sec, χρησιμοποιούμε την συνάρτηση exprnd με παράμετρο λ’=0.2 (μέση τιμή της εκθετικής κατανομής).



B)
Ο μέσος αριθμός γεγονότων σε ένα χρονικό παράθυρο ΔT γνωρίζουμε ότι ακολουθεί την κατανομή Poisson με μέση τιμή λt. Για τον μέσο αριθμό γεγονότων στη μονάδα του χρόνου για χρόνους μεταξύ γεγονότων που δημιουργήθηκαν με την διαδικασία που περιγράφηκε στο προηγούμενο ερώτημα έχουμε για 200,300,500,1000,10000 γεγονότα αντίστοιχα:

```
average_events_per_unit_time_1 =  5.6788
average_events_per_unit_time_2 =  4.6674
average_events_per_unit_time_3 =  4.9431
average_events_per_unit_time_4 =  5.3146
average_events_per_unit_time_5 =  4.9791
```

<u>Κώδικας που χρησιμοποιήθηκε:</u>

1)

```
clc;
clear all;
close all;

# TASK: In a common diagram, design the Probability Mass Function of Poisson processes
# with lambda parameters 3, 10, 30, 50. In the horizontal axes, choose k parameters
# between 0 and 70.

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");

# TASK: regarding the poisson process with parameter lambda 30, compute its mean
# value and variance

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);

# TASK: consider the convolution of the Poisson distribution with lambda 20 with
# the Poisson distribution with lambda 30.

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");

# 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;


2)#TASK: PDF of exponential Distribution with mean values 1/λ = {0.5,1,3}. Use
#K = 0:0.00001:8
k = 0:0.00001:8;
means = [0.5,1,3];
for i=1:columns(means)
  exponential(i,:) = exppdf(k,means(i));
endfor

colors = "rbm";
figure(1);
hold on;
for i=1:columns(means)
  plot(k,exponential(i,:),colors(i),"linewidth",1.2);
endfor
hold off;

title("Probability Density Function of Exponential Distribution");
xlabel("k values");
ylabel("probability");
legend("mean=0.5","mean=1","mean=3");


#TASK CDF of exponential distributions
k = 0:0.00001:8;
means = [0.5,1,3];
for i=1:columns(means)
  cdfexponential(i,:) = expcdf(k,means(i));
endfor

colors = "rbm";
figure(1);
hold on;
for i=1:columns(means)
  plot(k,cdfexponential(i,:),colors(i),"linewidth",1.2);
endfor
hold off;

title("Cumulative Distribution Function of Exponential Distribution");
xlabel("k values");
```

```
ylabel("probability");
legend("mean=0.5","mean=1","mean=3");


#TASK : Memory Loss of Exponential Distribution
k = 0:0.00001:8;
new_means = 2.5;
new_cdf_exponential = expcdf(k,new_means);
first = 1 - new_cdf_exponential(30000);
second_1 = 1 - new_cdf_exponential(50000);
second_2 = 1 - new_cdf_exponential(20000);
second = second_1 / second_2 ;
display(first);
display(second);
```

3)

```
#TASK: Poission N(t)
mul = 0.2;
sz = [1,100];
r = exprnd(mul,sz);
count = zeros(columns(r));
counter = 0;
for i=1:columns(r)
  counter = counter + r(i);
  count(i) = counter;
endfor
stairs(count);
title("Poisson Counting Process");
xlabel("counter");
ylabel("time");


#TASK: number of events in a period of time dT = t1 - t2
mul = 0.2;
array_1 = exprnd(mul,1,200);
array_2 = exprnd(mul,1,300);
array_3 = exprnd(mul,1,500);
array_4 = exprnd(mul,1,1000);
array_5 = exprnd(mul,1,10000);
counter_1 = 0;
counter_2 = 0;
counter_3 = 0;
counter_4 = 0;
counter_5 = 0;
for i=1:columns(array_1);
  counter_1 = counter_1 + array_1(i);
  count_1(i) = counter_1;
endfor
for i=1:columns(array_2);
```

```
  counter_2 = counter_2 + array_2(i);
  count_2(i) = counter_2;
endfor
for i=1:columns(array_3);
  counter_3 = counter_3 + array_3(i);
  count_3(i) = counter_3;
endfor
for i=1:columns(array_4);
  counter_4 = counter_4 + array_4(i);
  count_4(i) = counter_4;
endfor
for i=1:columns(array_5);
  counter_5 = counter_5 + array_5(i);
  count_5(i) = counter_5;
endfor
average_events_per_unit_time_1 = 200/counter_1;
average_events_per_unit_time_2 = 300/counter_2;
average_events_per_unit_time_3 = 500/counter_3;
average_events_per_unit_time_4 = 1000/counter_4;
average_events_per_unit_time_5 = 10000/counter_5;
display(average_events_per_unit_time_1);
display(average_events_per_unit_time_2);
display(average_events_per_unit_time_3);
display(average_events_per_unit_time_4);
display(average_events_per_unit_time_5);
```