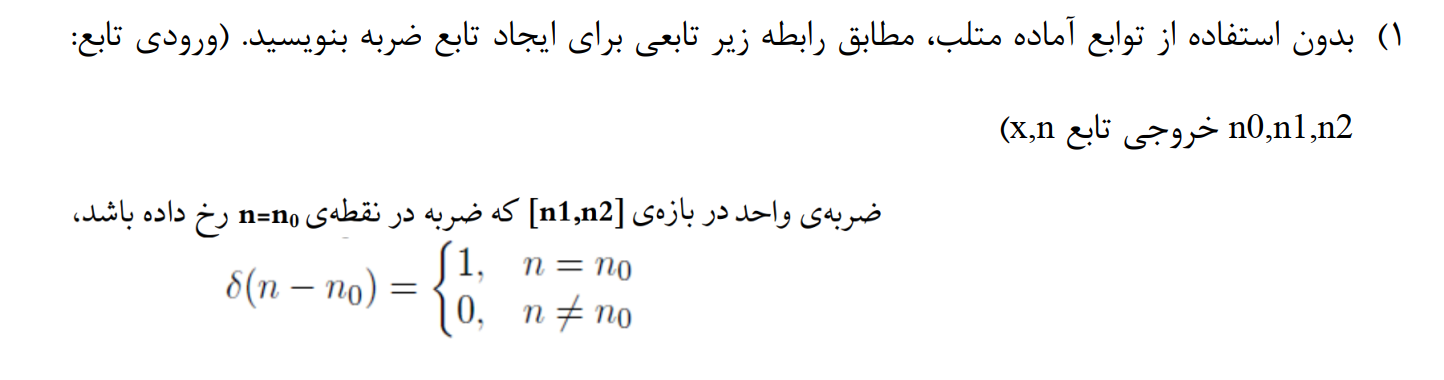


آزمایش 4

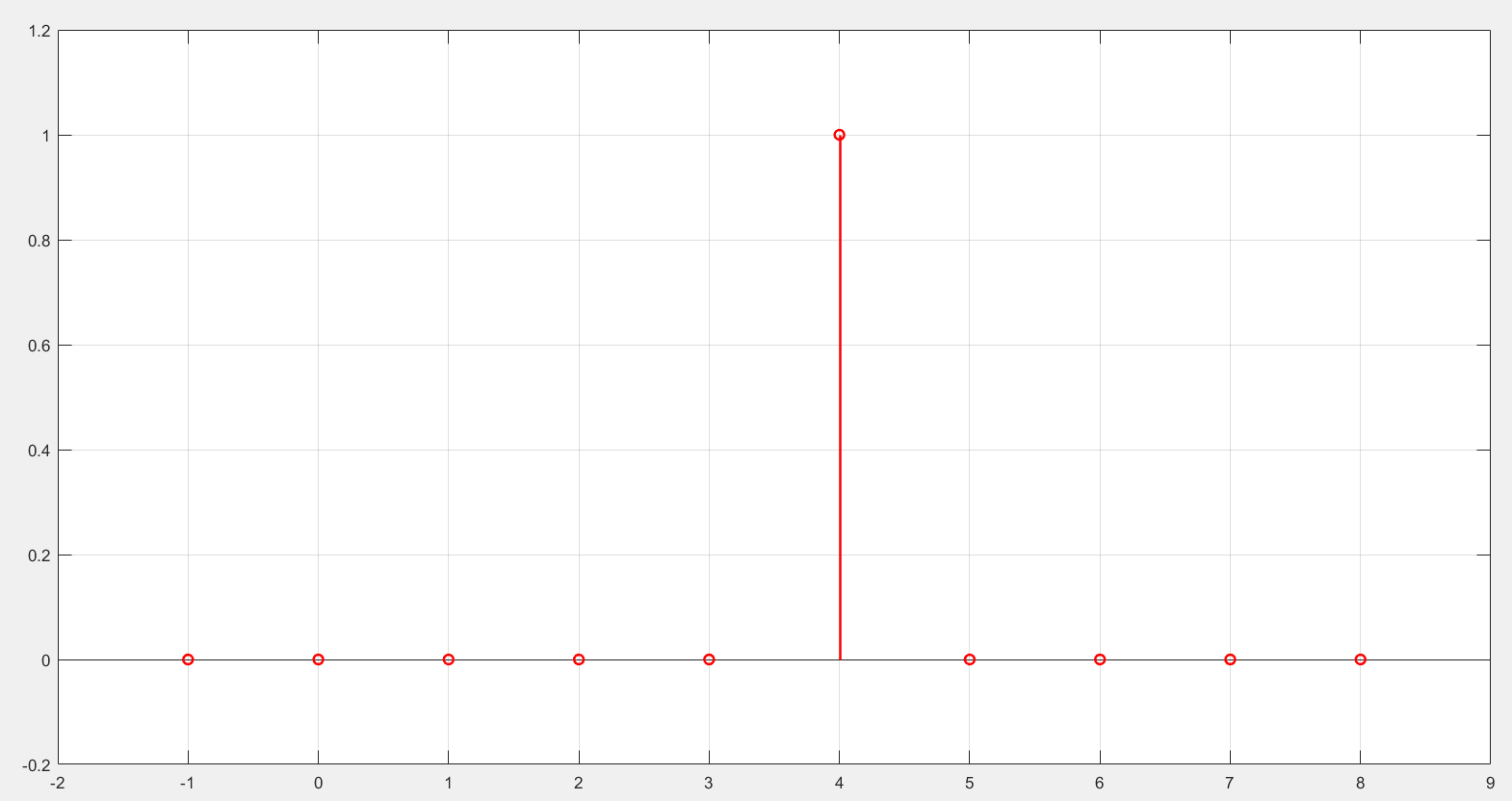
نام استاد : جناب دکتر مقیمی

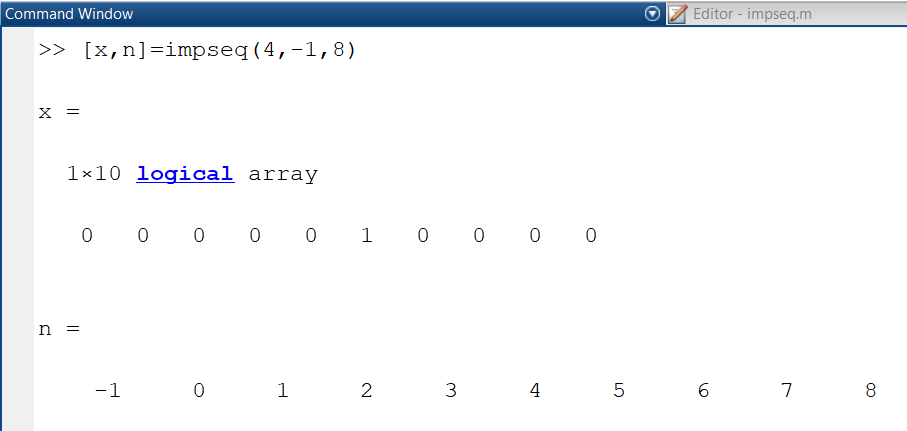
نام دانشجو : محمد توزنده جانی

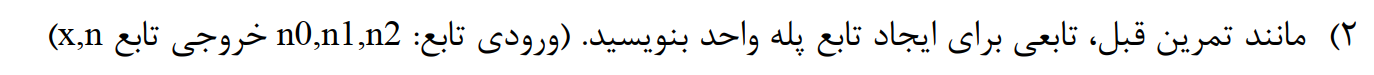
9720783

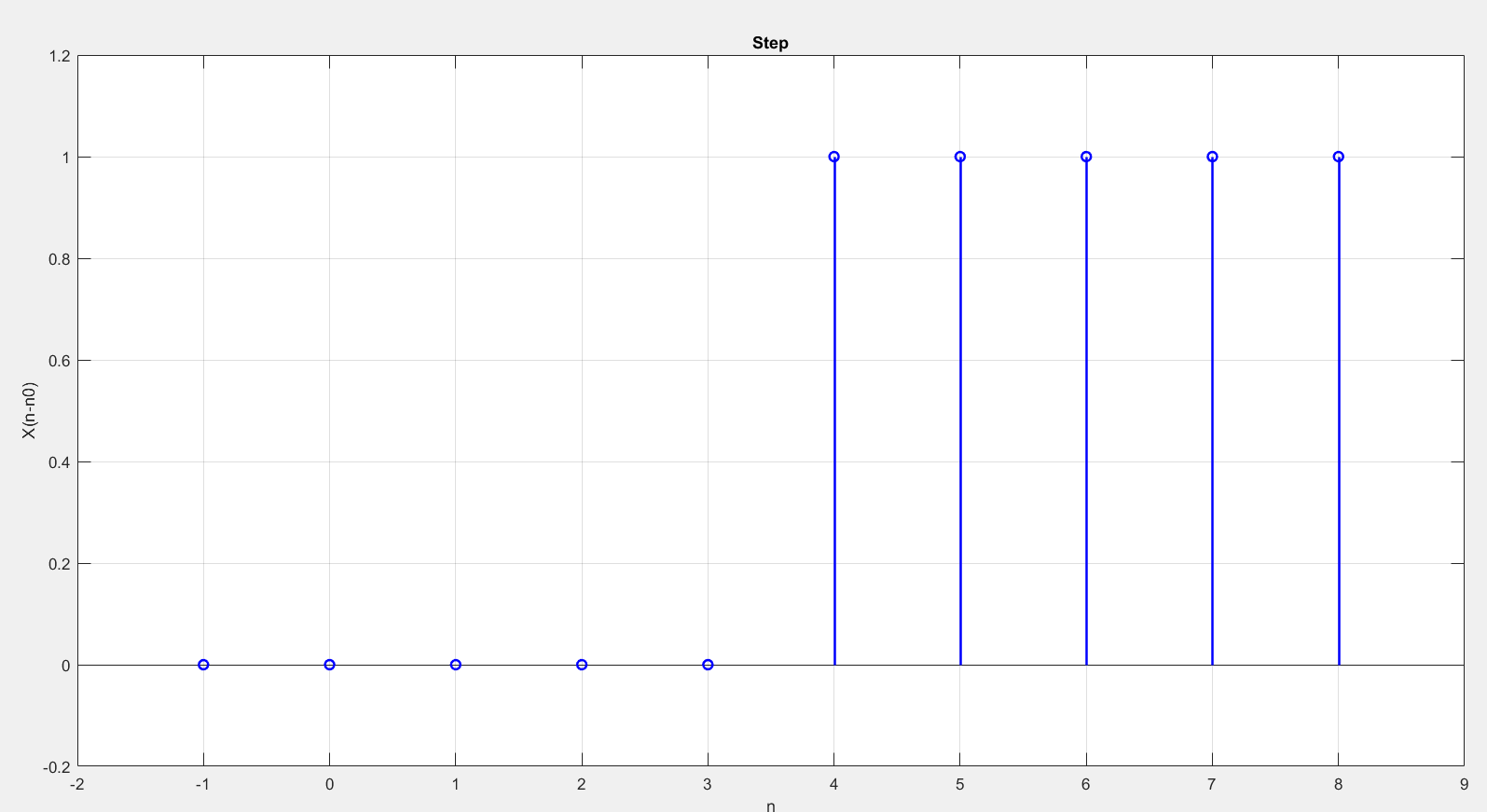


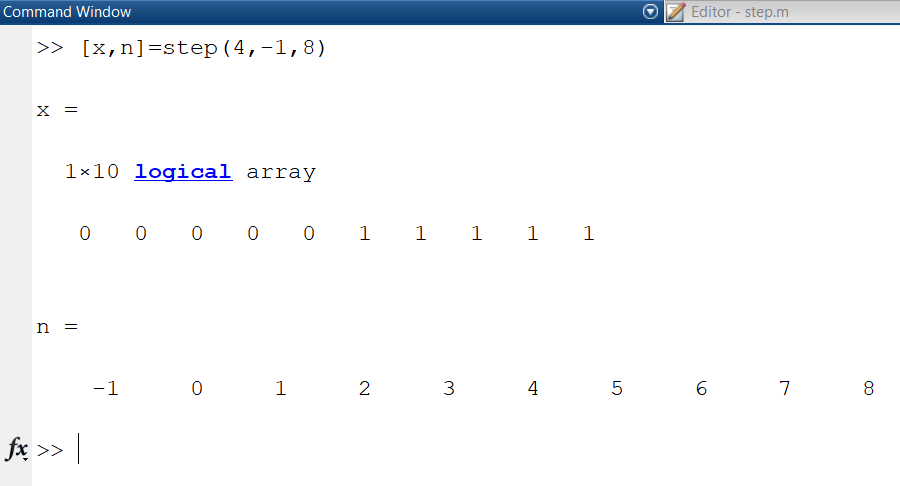
* کدهای متلب به صورت زیر می باشد:
* function [x,n]=impseq(n0,n1,n2)
* n=n1:n2;
* x=[(n-n0)==0]; %Output(x) is Logical
* stem(n,x,'r','linewidth',1.5)
* End
* خط x=[(n-n0)==0]; ماتریس منطقی از به نام x می سازد که در تمام اعداد صحیح بین بازه ی n1 و n2 مقدار صفر دارد جز در n=n0 که مقدار این ماتریس 1 می باشد که برابر دامنه تابع ضربه است.
* مثال: نمایش تابع ضربه در بازه ی [-1 , 8]

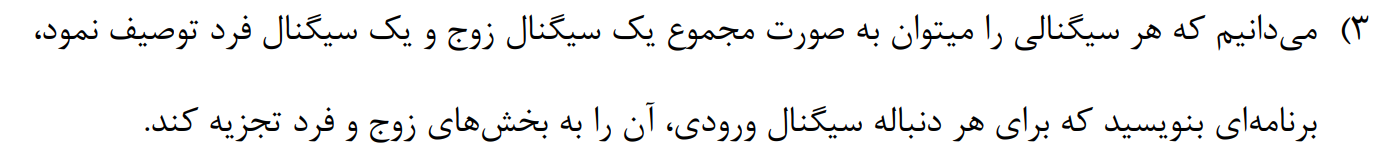






* کد های متلب به صورت زیر است :
* function [x,n]=step(n0,n1,n2)
* n=n1:n2;
* x=(n-n0)>=0; %Output(x) is Logical
* stem(n,x,'b','linewidth',1.5)
* title('Step')
* ylabel('X(n-n0)')
* xlabel('n')
* xlim([n1-1 n2+1])
* ylim([-0.2 1.2])
* grid on
* end
* مثال: نمایش تابع پله در بازه ی [-1 , 8] :





* هر سیگنال از یک بخش فرد و یک بخش روج تشکیل شده است که برای بدست آوردن این بخش ها از دو رابطه ی زیر به صورت مجزا استفاده می شود:
* کد های متلب به صورت زیر می باشد :

function [Xe,Xo,Sum,n1]=EvenOdd(x,n)

%------------------------------

x1=fliplr(x); %Create x1=x(-n)

n1=fliplr(-n); %Create new axis

Xe=0.5\*(x+x1);

Xo=0.5\*(x-x1);

Sum=Xe+Xo;

%------------------------------

figure(1)

subplot(2,1,1)

stem(n1,Xe,'r','linewidth',2.2)

title('X\_E')

xlabel('n')

grid minor

subplot(2,1,2)

stem(n1,Xo,'b','linewidth',2.2)

title('X\_O')

xlabel('n')

grid minor

%------------------------------

figure(2)

subplot(2,1,1)

stem(n,x,'g','linewidth',2.2)

title('X(n)')

xlabel('n')

grid minor

subplot(2,1,2)

stem(n,Sum,'k','linewidth',2.2)

title('X\_E+X\_O')

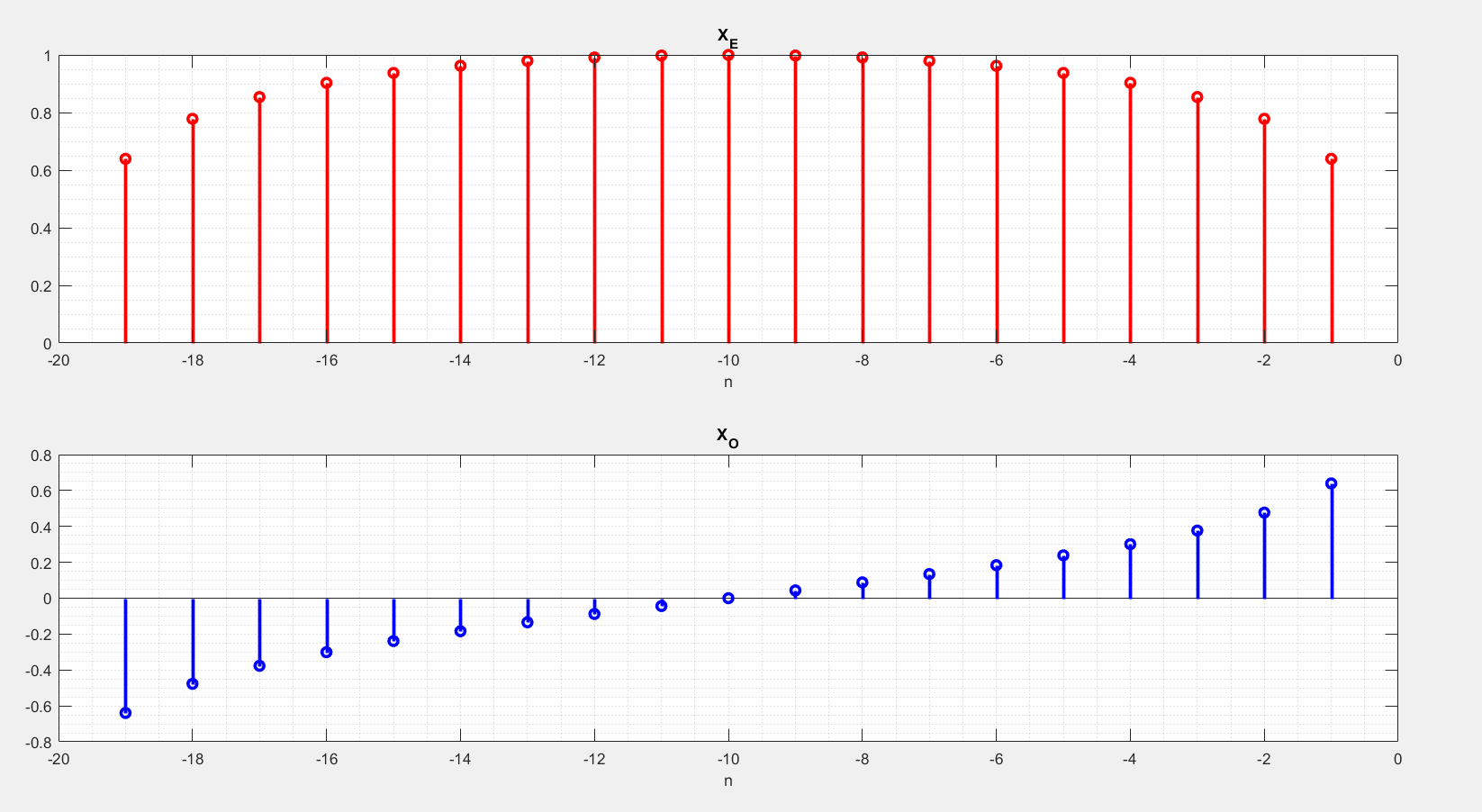
xlabel('n')

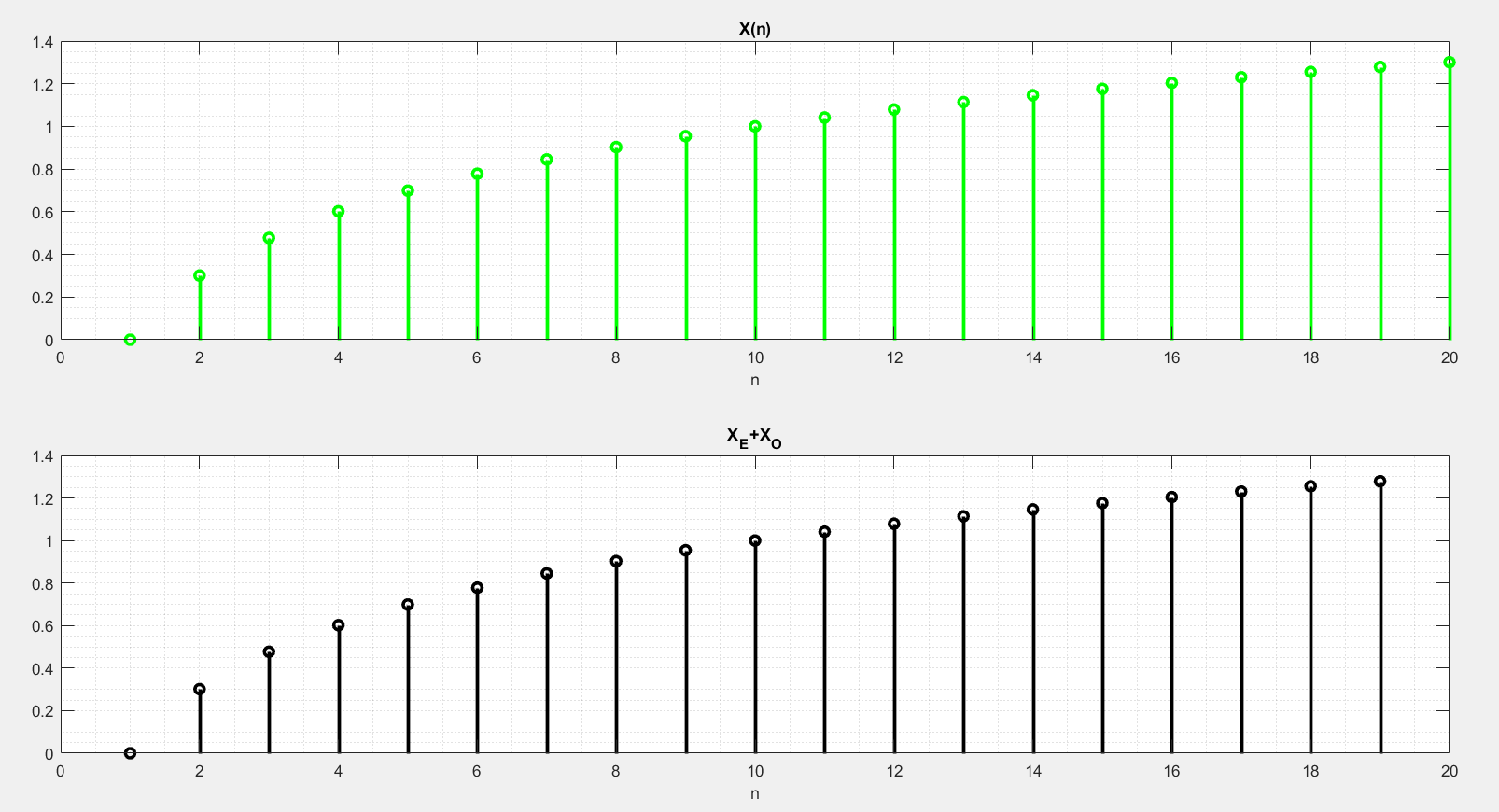
grid minor

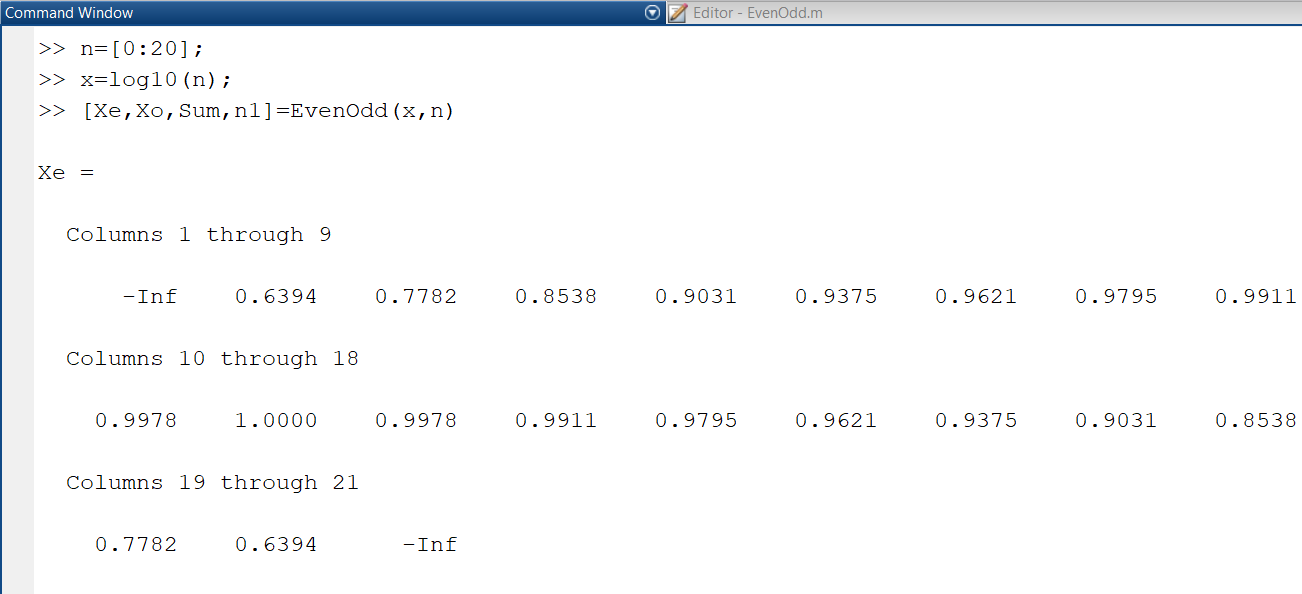
%------------------------------

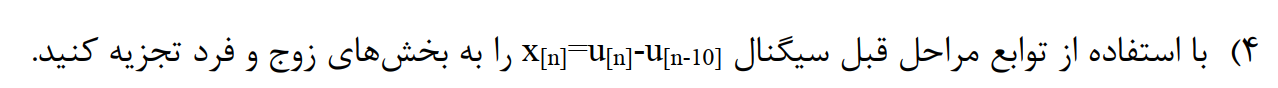
end

* مثال: نمایش قسمت فرد و زوج سیگنال در بازه ی [0 , 20]

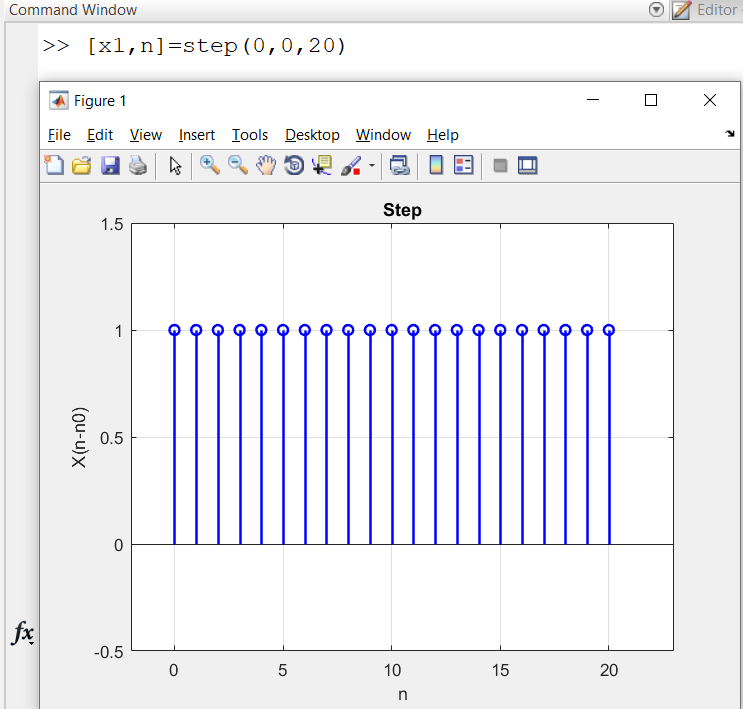




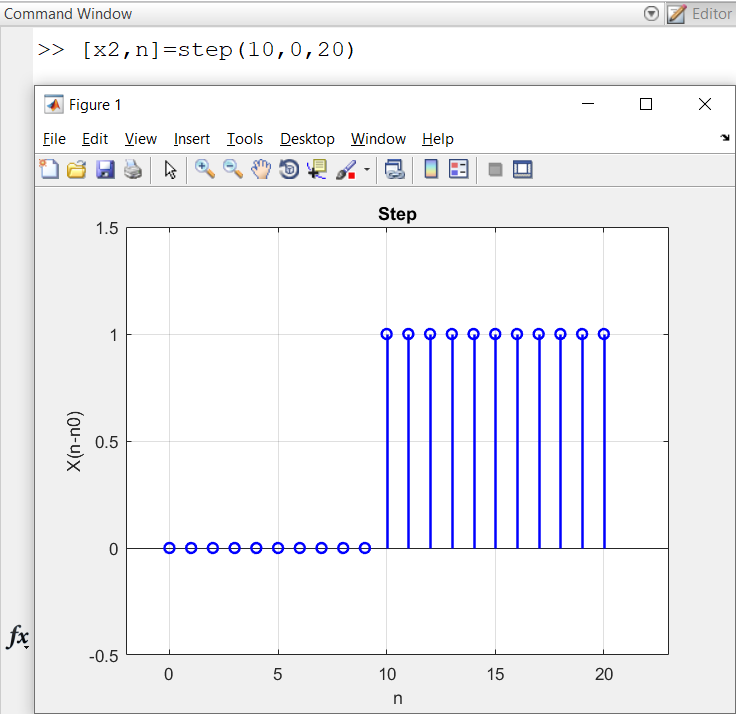
* همانطور که در شکل فوق مشخص است حاصل جمع دو سیگنال فرد و زوج برابر سیگنال اصلی است بنابراین تابع نوشته شده درست عمل می نماید.

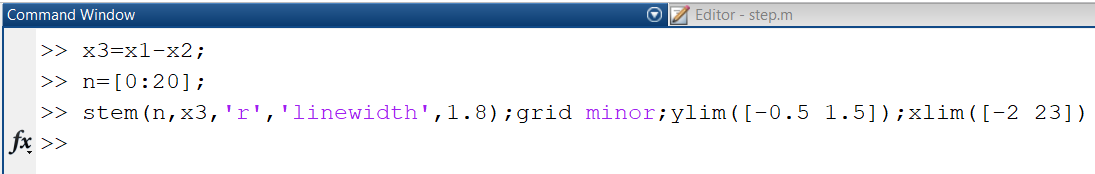
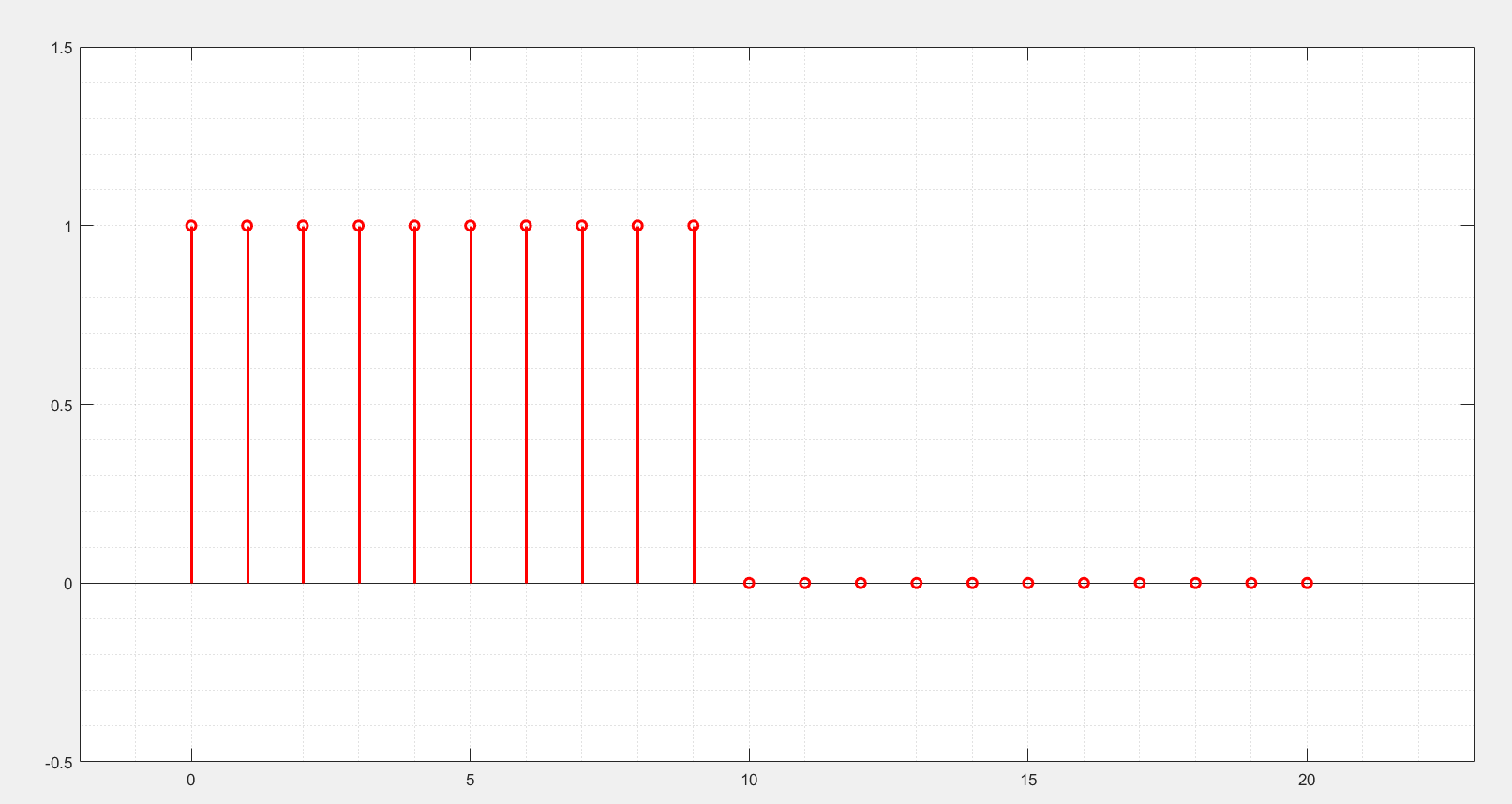


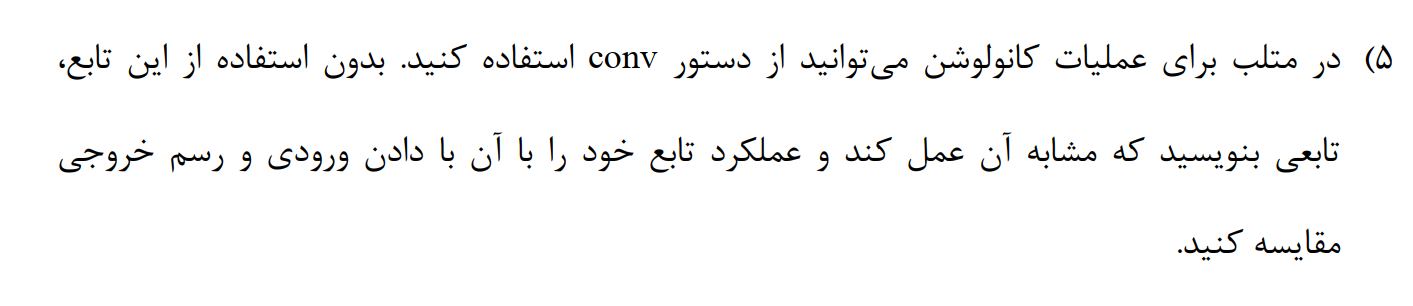
* سیگنال U[n] را از بازه 0 تا 20 ایجاد می کنیم :



* سیگنال U[n-10] را از بازه 0 تا 20 ایجاد می کنیم :



* حال سیگنال را با تفریق دو سیگنال فوق ایجاد کرده و با دستور stem رسم می کنیم :



function Y = Conv(X,H)

z = [];

for i = 1:length(X)

g = H.\*X(i);

z = [z;g];

end

[r c]=size(z);

k=r+c;

t=2;

Y=[];

cd=0;

while(t<=k)

for i=1:r

for j=1:c

if((i+j)==t)

cd = cd + z(i,j);

end

end

end

t=t+1;

Y=[Y cd];

cd=0;

end

end

