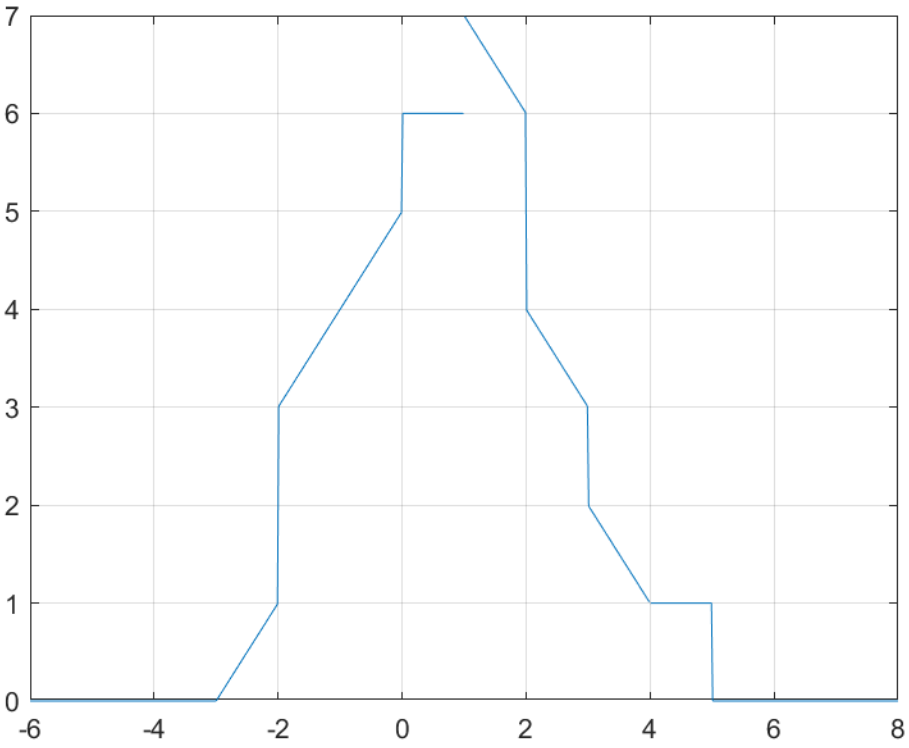
3-b code

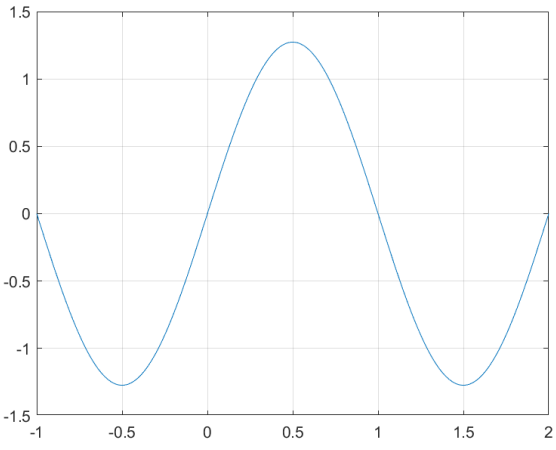
```
syms t;
syms tau;

x(t)= rectangularPulse(-2,2,t)+dirac(t-1);
h(t)= rectangularPulse(-1,2,t)+2*dirac(t)+dirac(t-3);
y(t)= int(x(tau)*h(t-tau),tau,-inf,inf);


 t=-6:0.01:8;
plot(t,y(t))
grid on
```



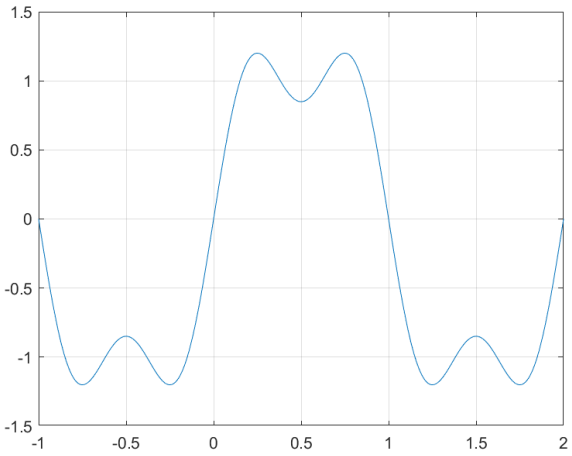
5-b code

```
syms t x N
t=-1:0.01:2;

x=0;
N=1; %/ N=1,N=3,N=5, N=50, N=1000 을 번갈아 넣으며 실행

for n=1:N
if rem(n,2) == 1
x=x+(4/(n*pi))*sin(n*pi*t);
end
end

plot(t,x)
grid on
xlim([-1,2])
```

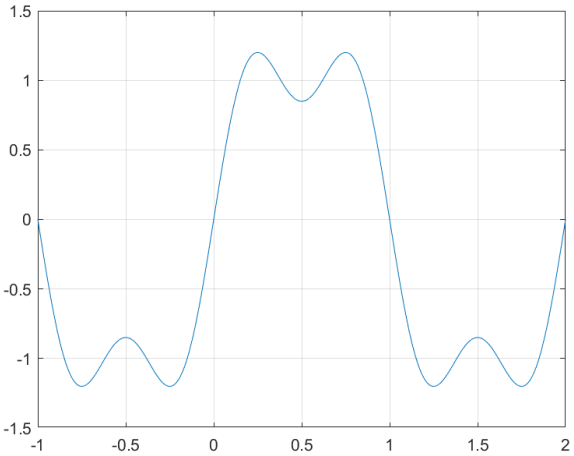
N=1



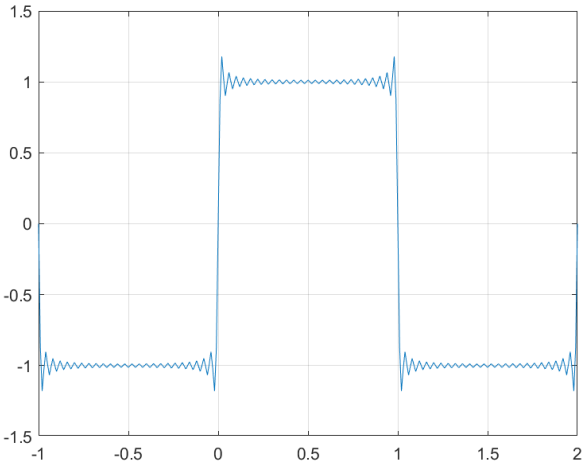
N=3



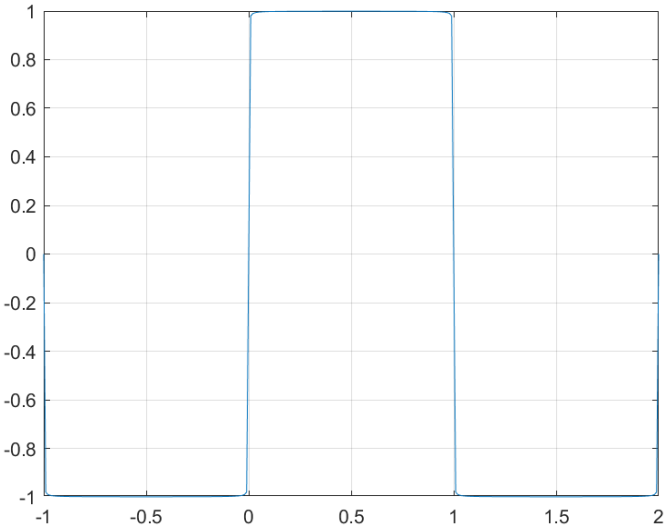
N=5


1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2

N=50


1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2

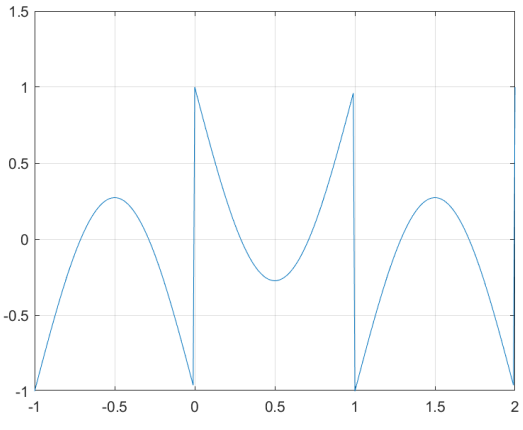
N=1000


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
2

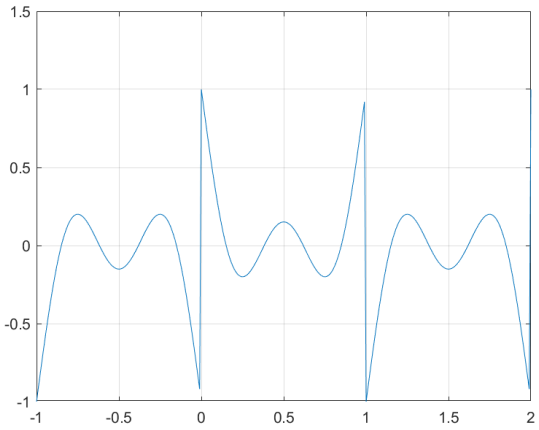
5-c code

```
syms x error N
t=-1:0.01:2;
x=0;
N=50; %/ N=1,N=3,N=5, N=50 을 번갈아 넣으며 실행

for n=1:N
if rem(n,2) == 1
x=x+(4/(n*pi))*sin(n*pi*t);
end
end

error=square(pi*t)-x;
plot(t,error)
grid on
xlim([-1,2])
```

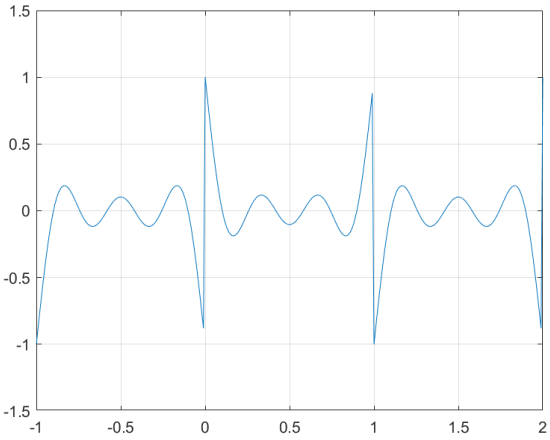
N=1



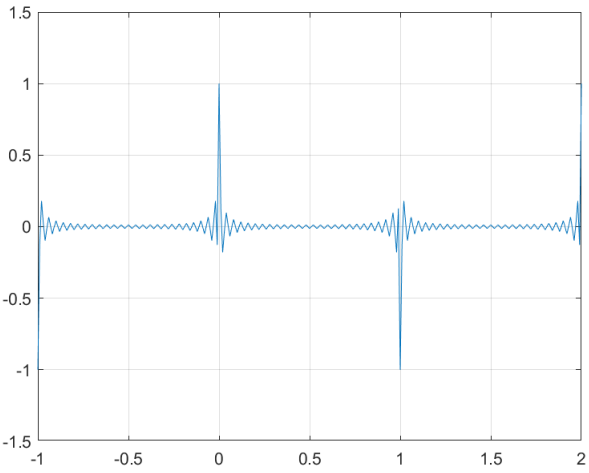
N=3



N=5



N=50



6-e code

Square 함수를 쓰기위해 라이브러리 추가설치 후 진행했다.

```
syms t x

t=-500:0.01:500;
x=square(pi*t);
h=exp(-t).*heaviside(t);

y=conv(x, h,'same');

plot(t,y)
xlim([-10,10])
grid on
```