Problema 3

3.1 Defintión de la funcion de Convolución y la Función escalón unitario

La convolución de dos señales se defiene según

$$f * g\,(t) = \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau \tag{0.1}$$

en el caso continuo y

$$f * g(n) = \sum_{k=-\infty}^{\infty} f(k)\, g(n-k) \tag{0.2}$$

en el caso discreto.

Definicmo entonces las funciones

```
In[1]:= DConv := Module[{k},
          Assuming[#3 ∈ Integers, Sum[#1[k] * #2[-k + #3], {k, -Infinity, Infinity}]] &]
```

para la convolucion de señales discretas, y

```
In[2]:= CConv := Module[{τ},
          Assuming[#3 ∈ Reals, Integrate[#1[τ] * #2[#3 - τ], {τ, -Infinity, Infinity}]] &]
```

para las señales continuas. Además resulta útil definir la función escalón unitario dada por

$$u\,(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases} \tag{0.3}$$

entonces

```
In[3]:= u[n_] := Piecewise[{{1, n ≥ 0}}, 0]
```

Ahora aplicamos estas definiciones para resolver los ejecicios siguientes:

Ejecicio 3.1

Hacer la convolucion de las fuinciones

$$x_1(n) = \left(\frac{1}{2}\right)^n u(n-4) \tag{0.4}$$

$$h_1(n) = 4^n\, u(2-n) \tag{0.5}$$

Definimos entonces

```
In[4]:= x1[n_] := (1/2)^n u[n - 4]
```

```
In[5]:= h1[n_] := 4^n u[2 - n]
```

La convolución la calculamos con la funcion `DConv` según

```
In[6]:= DConv[x1, h1, n]
```

$$\text{Out[6]=} \begin{cases} \frac{2^{9-n}}{7} & n > 6 \\ \frac{1}{7}\, 2^{-9+2n} & \text{True} \end{cases}$$

entonces

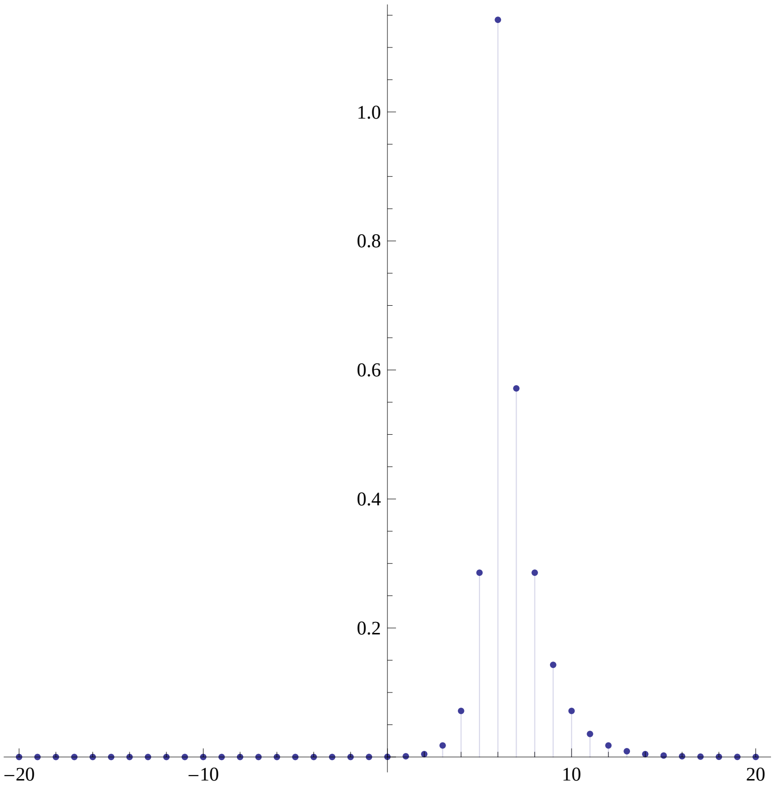
$$y_1(n) = \begin{cases} \frac{2^{9-n}}{7} & n > 6 \\ \frac{1}{7}\, 2^{-9+2n} & n \le 6 \end{cases} \tag{0.6}$$

y definimos

In[7]:=
```
y1[n_] := Piecewise[{{2^(9 - n) / 7, n > 6}}, 2^(-9 + 2 * n) / 7]
```

In[8]:=
```
DiscretePlot[y1[n], {n, -20, 20}, AspectRatio → 1, PlotRange → Full]
```

Out[8]=



Ejecicio 3.2

Hacer la convolucion de las fuinciones

$$x_2(n) = u(-n) - u(n-2) \tag{0.7}$$

$$h_2(n) = u(n-1) - u(n-4) \tag{0.8}$$

Definimos entonces

In[9]:=
```
x2[n_] := u[-n] - u[-n - 2]
```

In[10]:=
```
h2[n_] := u[n - 1] - u[n - 4]
```

La convolución la calculamos con la funcion `DConv` según

In[11]:=
```
DConv[x2, h2, n]
```

Out[11]=
```
1  n == 0 || n == 3
2  1 ≤ n < 3
0  True
```

entonces

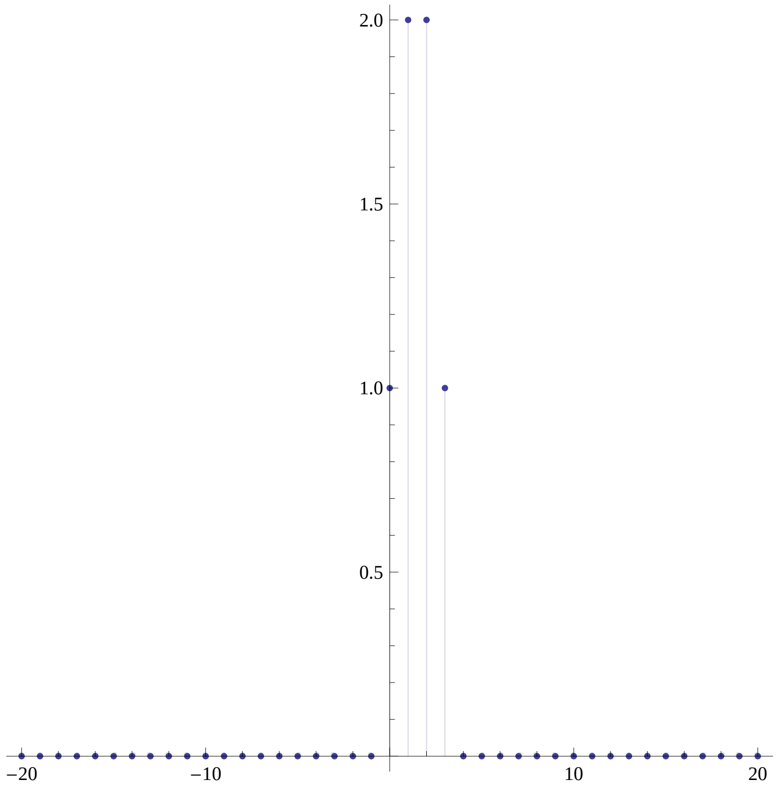
$$y_1(n) = \begin{cases} 1 & n = 1 \vee n = 3 \\ 2 & 1 \le n < 3 \\ 0 & n < 1 \vee 3 < n \end{cases} \tag{0.9}$$

y definimos

```
In[12]:= y2[n_] :=
  Piecewise[{{1, n == 0 || n == 3}, {2, Inequality[1, LessEqual, n, Less, 3]}}, 0]
```

```
In[13]:= DiscretePlot[y2[n], {n, -20, 20}, AspectRatio → 1, PlotRange → Full]
```

Out[13]=



Ejecicio 3.3

Hacer la convolucion de las fuinciones

$$x_3(n) = u(n) \tag{0.10}$$

$$h_3(n) = \left(\frac{1}{2}\right)^{-n} u(-n) \tag{0.11}$$

Definimos entonces

```
In[14]:= x3[n_] := u[n]
```

```
In[15]:= h3[n_] := (1/2)^-n u[-n]
```

La convolución la calculamos con la funcion `DConv` según

```
In[16]:= DConv[x3, h3, n]
```

Out[16]= $\begin{cases} 2 & n > 0 \\ 2^{1+n} & \text{True} \end{cases}$

entonces

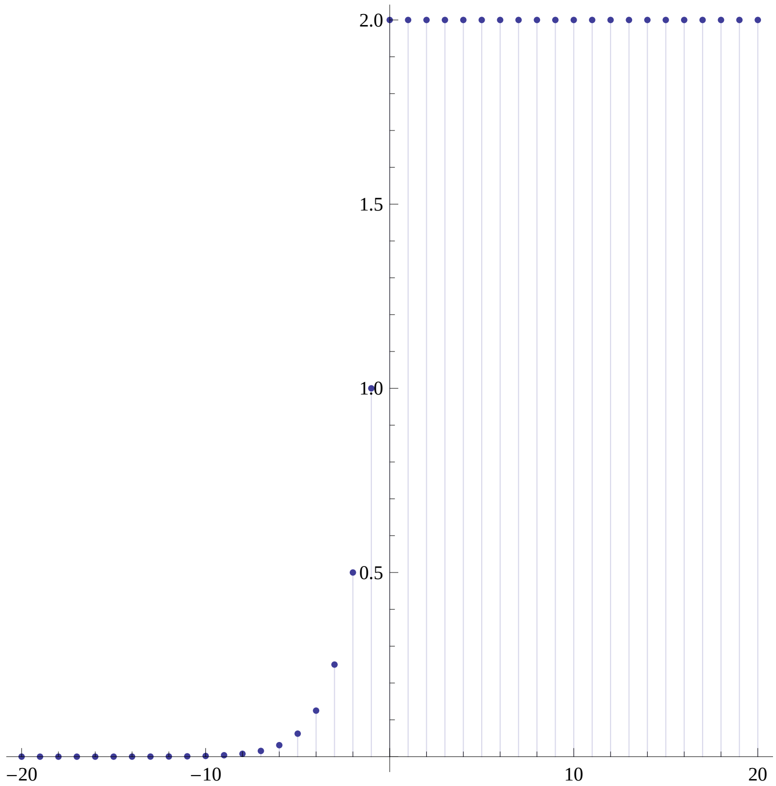
$$y_1(n) = \begin{cases} 2 & n > 0 \\ 2^{1+n} & n \le 0 \end{cases} \tag{0.12}$$

y definimos

```
In[17]:= y3[n_] := Piecewise[{{2, n > 0}}, 2^(1 + n)]
```

In[18]:= `DiscretePlot[y3[n], {n, -20, 20}, AspectRatio → 1, PlotRange → Full]`

Out[18]=



Ejecicio 3.4

Hacer la convolucion de las fuinciones

$$x_4(n) = e^{-a\,t}\, u(t) \tag{0.13}$$

$$h_4(n) = e^{-a\,t}\, u(t) \tag{0.14}$$

Definimos entonces

In[19]:= `x4[t_] := Exp[-a t] u[t]`

In[20]:= `h4[t_] := Exp[-a t] u[t]`

En este caso las funciones son continuas por lo cual la convolución la calculamos con la funcion `CConv` según

In[21]:= `CConv[x4, h4, t]`

Out[21]= $\begin{cases} e^{-a\,t}\, t & t > 0 \\ 0 & \text{True} \end{cases}$

entonces

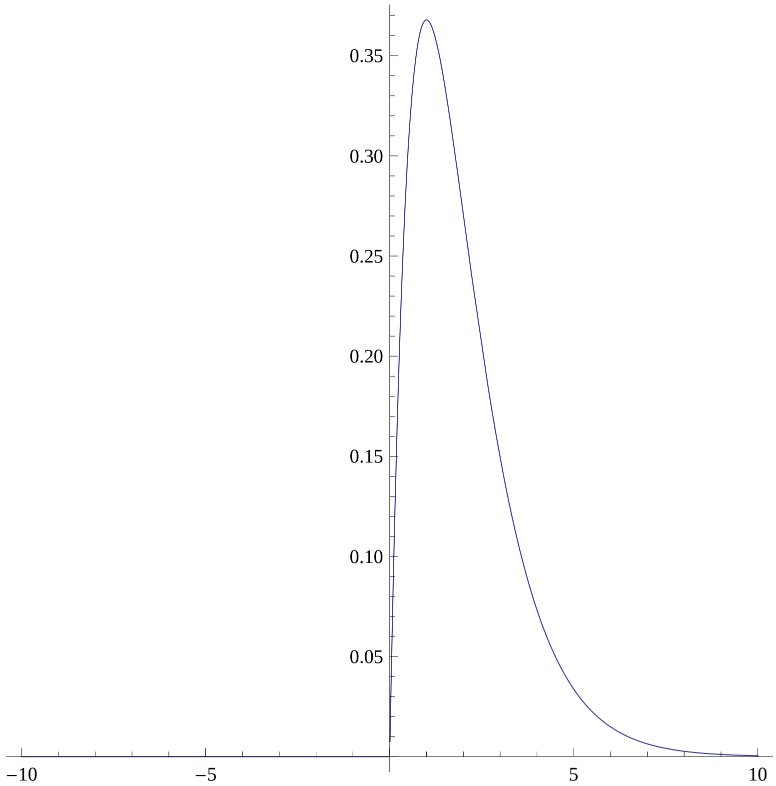
$$y_1(n) = \begin{cases} e^{-a\,t}\, t & t > 0 \\ 0 & t \leq 0 \end{cases} \tag{0.15}$$

y definimos

In[22]:= `y4[t_, a_] := Piecewise[{{t / E^(a * t), t > 0}}, 0]`

In[23]:=
```
Plot[y4[n, 1], {n, -10, 10}, AspectRatio → 1, PlotRange → Full]
```

Out[23]=



Para distintos valores de a hacemos

In[24]:=
```
Manipulate[Plot[y4[n, a], {n, -10, 10},
  AspectRatio -> 1, PlotRange -> Full], {a, -10, 10, 1}]
```

Out[24]=