RiordanLaguerre

March 5, 2020

In [24]:
```
from sympy import *
from IPython.display import *
init_printing()
var('a:z')
var('A:Z');
```

In [25]:
```
V=z/(1-z)
Z=solve(V-v,z)[0]
Z
```

Out[25]:

$$\frac{v}{v+1}$$

In [26]:
```
N=9
p=[]
f=series(exp(x*Z),v,0,N)
for i in range(N):
    p.append(factorial(i)*f.coeff(v,i))
p
```

Out[26]:

$$\left[1, \quad x, \quad x^2-2x, \quad x^3-6x^2+6x, \quad x^4-12x^3+36x^2-24x, \quad x^5-20x^4+120x^3-240x^2+120x, \quad x^6-30x^5+\right.$$

In [27]:
```
N=9
q=[]
f=series(exp(y*V),z,0,N)
for i in range(N):
    q.append(factorial(i)*f.coeff(z,i))
q
```

Out[27]:

$$\left[1, \quad y, \quad y^2+2y, \quad y^3+6y^2+6y, \quad y^4+12y^3+36y^2+24y, \quad y^5+20y^4+120y^3+240y^2+120y, \quad y^6+30y^5+3\right.$$

In [28]:
```
PCF=Matrix(N,N,lambda n,k: p[n].coeff(x,k))
QCF=Matrix(N,N,lambda n,k: q[n].coeff(y,k))
PCF,QCF,simplify(PCF*QCF)
```

Out[28]:

$$\left(\begin{bmatrix}1&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0\\0&-2&1&0&0&0&0&0&0\\0&6&-6&1&0&0&0&0&0\\0&-24&36&-12&1&0&0&0&0\\0&120&-240&120&-20&1&0&0&0\\0&-720&1800&-1200&300&-30&1&0&0\\0&5040&-15120&12600&-4200&630&-42&1&0\\0&-40320&141120&-141120&58800&-11760&1176&-56&1\end{bmatrix}, \begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&2&1&0&0\\0&6&6&1&0\\0&24&36&12&1\\0&120&240&120&20\\0&720&1800&1200&300\\0&5040&15120&12600&4200\\0&40320&141120&141120&58800\end{bmatrix}$$

In [29]:
```
qa=[]
for n in range(N):
    qa.append(bool(sum(p[n-k]*p[k].subs(x,y)*binomial(n,k) for k in range(n+1)).expan
qa
```

Out[29]: [True, True, True, True, True, True, True, True, True]

In [30]:
```
W=[]
WW=[]
for n in range(N):
    W.append(zeros(N,N))
    WW.append(zeros(N,N))
    for k in range(floor(N/2)):
        for l in range(floor(N/2)):
            W[n][k,l]=sum(binomial(n,j)*PCF[n-j,k]*PCF[j,l] for j in range(n+1))
            WW[n][k,l]=PCF[n,k+l]*binomial(k+l,l)
[(W[a]-WW[a]).is_zero for a in range(N)]
```

Out[30]: [True, True, True, True, True, True, True, True, True]

In [36]:
```
for i in range(N):
    display([p[i],q[i]])
```

$$[1, \quad 1]$$

$$[x, \quad y]$$

$$\left[x^2 - 2x, \quad y^2 + 2y\right]$$

$$\left[x^3 - 6x^2 + 6x, \quad y^3 + 6y^2 + 6y\right]$$

$$\left[x^4 - 12x^3 + 36x^2 - 24x, \quad y^4 + 12y^3 + 36y^2 + 24y \right]$$

$$\left[x^5 - 20x^4 + 120x^3 - 240x^2 + 120x, \quad y^5 + 20y^4 + 120y^3 + 240y^2 + 120y \right]$$

$$\left[x^6 - 30x^5 + 300x^4 - 1200x^3 + 1800x^2 - 720x, \quad y^6 + 30y^5 + 300y^4 + 1200y^3 + 1800y^2 + 720y \right]$$

$$\left[x^7 - 42x^6 + 630x^5 - 4200x^4 + 12600x^3 - 15120x^2 + 5040x, \quad y^7 + 42y^6 + 630y^5 + 4200y^4 + 12600y^3 + 15120y^2 + \right.$$

$$\left[x^8 - 56x^7 + 1176x^6 - 11760x^5 + 58800x^4 - 141120x^3 + 141120x^2 - 40320x, \quad y^8 + 56y^7 + 1176y^6 + 11760y^5 + 58\right.$$

In [32]:
```
yy=[]
for m in range(N):
    g=0
    for i in range(m+1):
       g=g+p[m].coeff(x,i)*q[i]
    yy.append(simplify(g))

display(yy)

xx=[]
for m in range(N):
    g=0
    for i in range(m+1):
       g=g+q[m].coeff(y,i)*p[i]
    xx.append(simplify(g))

display(xx)
```

$$\left[1, \quad y, \quad y^2, \quad y^3, \quad y^4, \quad y^5, \quad y^6, \quad y^7, \quad y^8 \right]$$

$$\left[1, \quad x, \quad x^2, \quad x^3, \quad x^4, \quad x^5, \quad x^6, \quad x^7, \quad x^8 \right]$$

In [33]:
```
display([(sum(binomial(n,k)*gamma(n)/gamma(k)*x**k*(-1)**(n-k) for k in range(1,n+1))
[(sum(binomial(n,k)*gamma(n)/gamma(k)*y**k for k in range(1,n+1))-q[n]) for n in rang
```

$$[0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0]$$

Out[33]:

$$[0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0]$$

Laguerre polynomials:

$$q_n(y) = n!\, L_n^{(-1)}(-y) = \sum_{k=1}^{n} \binom{n}{k} \frac{\Gamma(n)}{\Gamma(k)}\, y^k$$

Alternating sign polynomials going from q to p:

$$p_n(x) = (-1)^n\, q_n(-x)$$

In [34]:
```
L=p
%store L
```

```
Stored 'L' (list)
```

In [37]:
```
y=""
for n in range(N):
    y=y+"p_{"+latex(n)+ "} = "+latex(p[n])+"\\\\ "
display(Math(y))
%store y > P.tex

y=""
for n in range(N):
    y=y+"q_{"+latex(n)+ "} = "+latex(q[n])+"\\\\ "
display(Math(y))
%store y > Q.tex
```

$p_0 = 1 p_1 = x p_2 = x^2 - 2x p_3 = x^3 - 6x^2 + 6x p_4 = x^4 - 12x^3 + 36x^2 - 24x p_5 = x^5 - 20x^4 + 120x^3 - 240x^2 + 120x p_6$

```
Writing 'y' (str) to file 'P.tex'.
```

$q_0 = 1 q_1 = y q_2 = y^2 + 2y q_3 = y^3 + 6y^2 + 6y q_4 = y^4 + 12y^3 + 36y^2 + 24y q_5 = y^5 + 20y^4 + 120y^3 + 240y^2 + 120y q_6 =$

```
Writing 'y' (str) to file 'Q.tex'.
```

In [ ]: