In [20]:
```
import re
import nltk
import requests
from bs4 import BeautifulSoup
import numpy as np
from nltk.book import *
import matplotlib.pyplot as plt
import tabulate as tab

html = requests.get('http://www.fedearroz.com.co/new/noticias.php').text
soup = BeautifulSoup(html)
type(soup)

soup
links = soup.find_all('a',onclick=True)
links
```

```
<a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3178', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3179', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3180', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3181', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3174', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3176', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3188', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3189', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3190', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
 <a ,="" href="#" onclick="MM_openBrWindow('../noticias/noticiasd2.php?id=3191', 'Noticia', 'scrollbars=yes, resi
zable=yes, width=700, height=550')" style="text-decoration:none; color:#009;">Ver noticia...</a>,
```

In [4]:
```
noticias = re.findall('id=+[0-9]+',str(links))
enlace = 'http://www.fedearroz.com.co/noticias/noticiasd2.php?'
datos = []
for i in range(len(noticias)):
    datos.append(enlace + noticias[i])
    print(datos[i])
print(len(datos))
```

```
id=3146)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3144 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3144)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3139 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3139)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3133 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3133)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3134 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3134)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3135 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3135)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3136 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3136)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3137 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3137)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3138 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3138)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3143 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
id=3143)
http://www.fedearroz.com.co/noticias/noticiasd2.php?id=3140 (http://www.fedearroz.com.co/noticias/noticiasd2.php?
```

In [3]:
```
#html = requests.get(datos[25]).text
#soup = BeautifulSoup(html)
#type(soup)
#fechas=[]
#titulos= soup.find_all('h1')
#fechas.append(soup.find_all('b'))
#contenido =  soup.find_all('p',{"class": "body"})
#notiteca.append(soup)
#print(titulos)
#print(fechas)
#print(contenido)
```

In [21]:
```
date  = []
title = []
body  = []

#for i in range(2942):
 #   html = requests.get(datos[i]).text
  #  soup = BeautifulSoup(html)
   # type(soup)
    #date.append(soup.find_all('b'))
```

In [5]:
```
#print(date[50][0])
#n = len(date)
#print (n)
#fecha =[]
#for i in range(n):
 #   if i != 32 or i != 50:
 #       temporal = str(date[i][0])
 #       temporal = re.sub('<b>|</b>', '',temporal)
 #       print(i,": ",temporal)
```

In [22]:
```
html2 = requests.get('http://www.fedearroz.com.co/new/noticias.php').text
soup2 = BeautifulSoup(html2)
type(soup2)
soup2
fecha = soup2.find_all('span',{"class": "fecha"})
fechas=[]
#print(fecha)
```

In [23]:
```
n= len(fecha)
fechas_unidas=''
años =[]
for i in range(n):

    temporal = str(fecha[i])
    temporal = re.sub('<span|class="fecha">|</span>', '',temporal)
    año = re.findall('[0-9]+', temporal)
    años.append(año)
    fechas_unidas = fechas_unidas+temporal
    fechas.append(temporal)
```

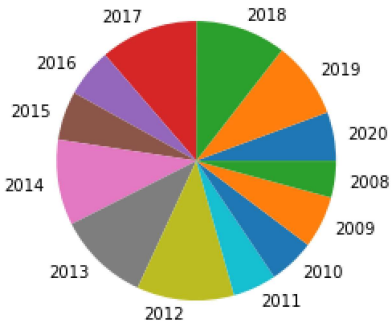
In [24]:
```
notiXaño = []
añolabel = []
actual = 2020
contador = 0

n= len(años)
#print (n)
for i in range(n):
    if(actual!=2007):
        if años[i][1] == str(actual):
            contador = contador+1
        else:
            notiXaño.append(contador)
            añolabel.append(actual)

            actual = actual-1
            contador = 0
    else:
        i=i+n
```

In [25]:
```
tabla=[]
tabla.append(añolabel)
tabla.append(notiXaño)
```

Años mas relevantes para el gremio Arrocero

In [26]:
```
print(tab.tabulate(tabla, headers='firstrow',tablefmt='fancy_grid'))
plt.pie(notiXaño, labels=añolabel)
plt.show()
```

2020	2019	2018	2017	2016	2015	2014	2013	2012	2011	2010	2009	2008
164	259	305	330	163	165	288	310	328	147	155	176	123



In [27]:
```
titulos =[]
uni_titulos=''

texto =[]
textosunidos =''

for i in range(2942):
    html = requests.get(datos[i]).text
    soup = BeautifulSoup(html)
    type(soup)
    title.append(soup.find_all('h1'))

    temporal = str(title[i])
    temporal = re.sub('<h1>|</h1>', '',temporal)
    temporal = temporal.replace('[','')
    temporal = temporal.replace(']','')
    uni_titulos=uni_titulos+temporal
    titulos.append(temporal)

    body.append(soup.find_all('p',{"class": "body"}))

    temporal2 = str(body[i])
    temporal2 = re.sub('\xa0\xa0|<p|class="body">|\n|</p>|<br/>|align="right"|vspace="5"/>|align="default"|</u>','',t
    temporal2 = temporal2.replace('[','')
    temporal2 = temporal2.replace(']','')
    textosunidos = textosunidos+ temporal2
    texto.append(temporal2)
```

In [ ]:
```
#print(titulos)
```

In [ ]:

In [ ]:

1 DEPARTAMENTOS EN TENDENCIA

In [7]:
```
textosunidostxt = str(textosunidos)
departamentos = ['Bogotá','Amazonas','Antioquía','Arauca','Atlántico','Bolívar',
'Boyacá','Caldas','Caquetá','Casanare','Cauca','Cesar','Chocó','Córdoba','Cundinamarca',
'Guainía','Guaviare','Huila','La','Magdalena','Meta','Nariño','Norte','Putumayo','Quindío',
'Risaralda','San','Santander','Sucre','Tolima','Valle','Vaupés','Vichada']
grafica_dp =[None]*len(departamentos)
for i in range(len(departamentos)):
    print(departamentos[i],': ',textosunidostxt.count(departamentos[i]))
    grafica_dp[i]=[departamentos[i],textosunidostxt.count(departamentos[i])]
```

```
Bogotá :  26
Amazonas :  1
Antioquía :  3
Arauca :  44
Atlántico :  3
Bolívar :  20
Boyacá :  0
Caldas :  21
Caquetá :  1
Casanare :  211
Cauca :  180
Cesar :  155
Chocó :  4
Córdoba :  102
Cundinamarca :  46
Guainía :  0
Guaviare :  9
Huila :  164
La :  2146
Magdalena :  114
Meta :  208
Nariño :  0
Norte :  121
Putumayo :  23
Quindío :  2
Risaralda :  21
San :  803
Santander :  186
Sucre :  166
Tolima :  331
Valle :  191
Vaupés :  0
Vichada :  10
```

2 Presidentes Nombrados

In [8]:
```
tot = textosunidostxt.count('Iván Duque') + textosunidostxt.count('Uribe')+textosunidostxt.count('Santos')

print('Duque',textosunidostxt.count('Iván Duque')*100/tot)
print('Duque',uni_titulos.count('Iván Duque'))

print('Uribe',textosunidostxt.count('Uribe'))
print('Uribe',uni_titulos.count('Uribe:'))

print('Juan manuel Santos',textosunidostxt.count('Santos'))
print('Juan manuel Santos',uni_titulos.count('Santos'))
```

```
Duque 1.935483870967742
Duque 1
Uribe 49
Uribe 1
Juan manuel Santos 103
Juan manuel Santos 5
```

3 PLAGA

In [9]:
```
print('Plaga',textosunidostxt.count('plagas'))
print('Plaga',textosunidostxt.count('Oryzophagus'))
print('Oebalus',textosunidostxt.count('Oebalus'))
print('Spodoptera',textosunidostxt.count('Spodoptera'))
print('Insectos',textosunidostxt.count('insectos'))
print('Helminthosporium',textosunidostxt.count('Helminthosporium'))
print('Rhizoctonia',textosunidostxt.count('Rhizoctonia'))
```

```
Plaga 138
Plaga 0
Oebalus 4
Spodoptera 22
Insectos 194
Helminthosporium 4
Rhizoctonia 17
```

4 Sanciones por libre competencia

In [10]:
```
print('Arroz Roa',textosunidostxt.count(' roa '))
print('Cartel',textosunidostxt.count('Cartel'))
print('Sanción',textosunidostxt.count('sanción'))
print('Flor huila',textosunidostxt.count('Flor huila'))
print('Consejo de estado',textosunidostxt.count('Consejo de estado'))
print('UNIÓN DE ARROCEROS',textosunidostxt.count('UNIÓN DE ARROCEROS'))
```

```
Arroz Roa 0
Cartel 0
Sanción 0
Flor huila 0
Consejo de estado 0
UNIÓN DE ARROCEROS 0
```

Contrabando

In [11]:
```
print('Peru',textosunidostxt.count('Peru'))
print('Uruguay',textosunidostxt.count('Uruguay'))
print('Estados unidos',textosunidostxt.count('Estados Unidos'))
print('Venezuela',textosunidostxt.count('Venezuela'))
print('Ecuador',textosunidostxt.count('Ecuador'))
```

```
Peru 3
Uruguay 15
Estados unidos 182
Venezuela 79
Ecuador 83
```

In [*]:
```
#print(textosunidostxt)
```

In [ ]:
```
```

In [13]:
```
textosunidostxt=re.sub('hspace="5"|border="1"|<img|align="right"|vspace="5"/>|align="default"|</u>|align="left"','',t
listaPalabras = textosunidostxt.split()
```

In [*]:
```
#print(listaPalabras)
frecuenciaPalab = []
for w in listaPalabras:
    frecuenciaPalab.append(listaPalabras.count(w))

#print("Lista\n" + str(listaPalabras) + "\n")
#print("Frecuencias\n" + str(frecuenciaPalab) + "\n")
print("Pares\n" + str(list(zip(listaPalabras, frecuenciaPalab))))
```

In [15]:

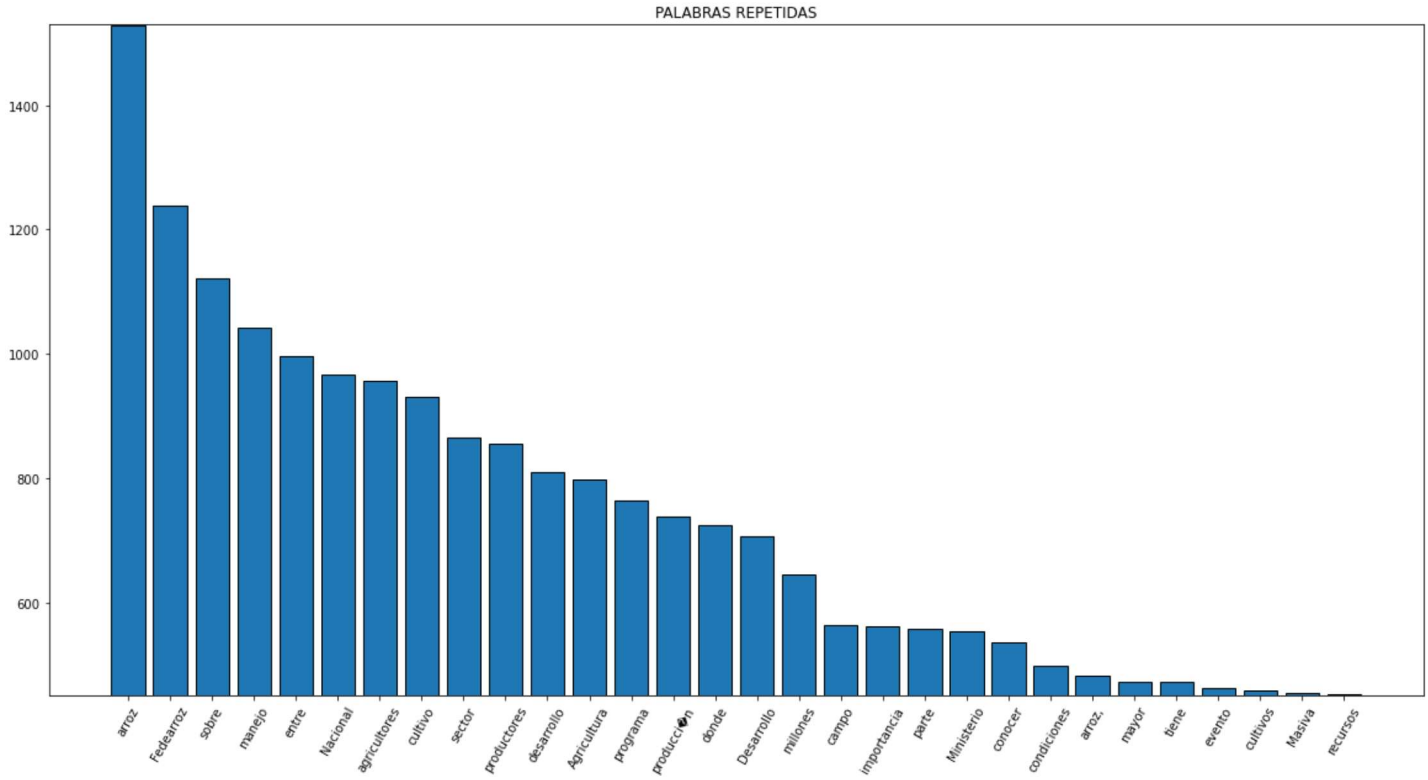
```
palabras = [w for w in listaPalabras if len(w) > 4]

#print (sorted(palabras))

print("Lista de palabras mayores a 4 letras")

frecuencia = FreqDist(palabras)
print(frecuencia)
print("")
print("LAS QUE MAS SE REPITEN")

grafica=frecuencia.most_common(30)
grafica
```

```
Lista de palabras mayores a 4 letras
<FreqDist with 51968 samples and 274130 outcomes>

LAS QUE MAS SE REPITEN
```

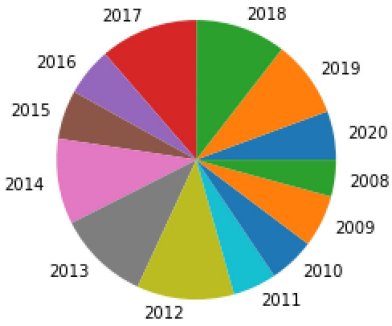
Out[15]:

```
[('arroz', 1529),
 ('Fedearroz', 1238),
 ('sobre', 1122),
 ('manejo', 1042),
 ('entre', 997),
 ('Nacional', 966),
 ('agricultores', 956),
 ('cultivo', 930),
 ('sector', 866),
 ('productores', 855),
 ('desarrollo', 810),
 ('Agricultura', 797),
 ('programa', 764),
 ('producci�n', 739),
 ('donde', 725),
 ('Desarrollo', 706),
 ('millones', 645),
 ('campo', 564),
 ('importancia', 562),
 ('parte', 558),
 ('Ministerio', 553),
 ('conocer', 535),
 ('condiciones', 498),
 ('arroz,', 482),
 ('mayor', 473),
 ('tiene', 473),
 ('evento', 463),
 ('cultivos', 458),
 ('Masiva', 455),
 ('recursos', 452)]
```

In [16]:
```
#grafica

x=[]
y= []

for i in range(len(grafica)):
    x.append(grafica[i][1])
    y.append(grafica[i][0])

s=len(x)
plt.figure(figsize=(20,10))
plt.bar(range(s), x, edgecolor='black')

plt.xticks(range(s), y, rotation=60)
plt.title("PALABRAS REPETIDAS")
plt.ylim(min(x)-1, max(x)+1)
plt.show()
```



In [28]:
```
plt.pie(notiXaño, labels=añolabel)
plt.show()
```



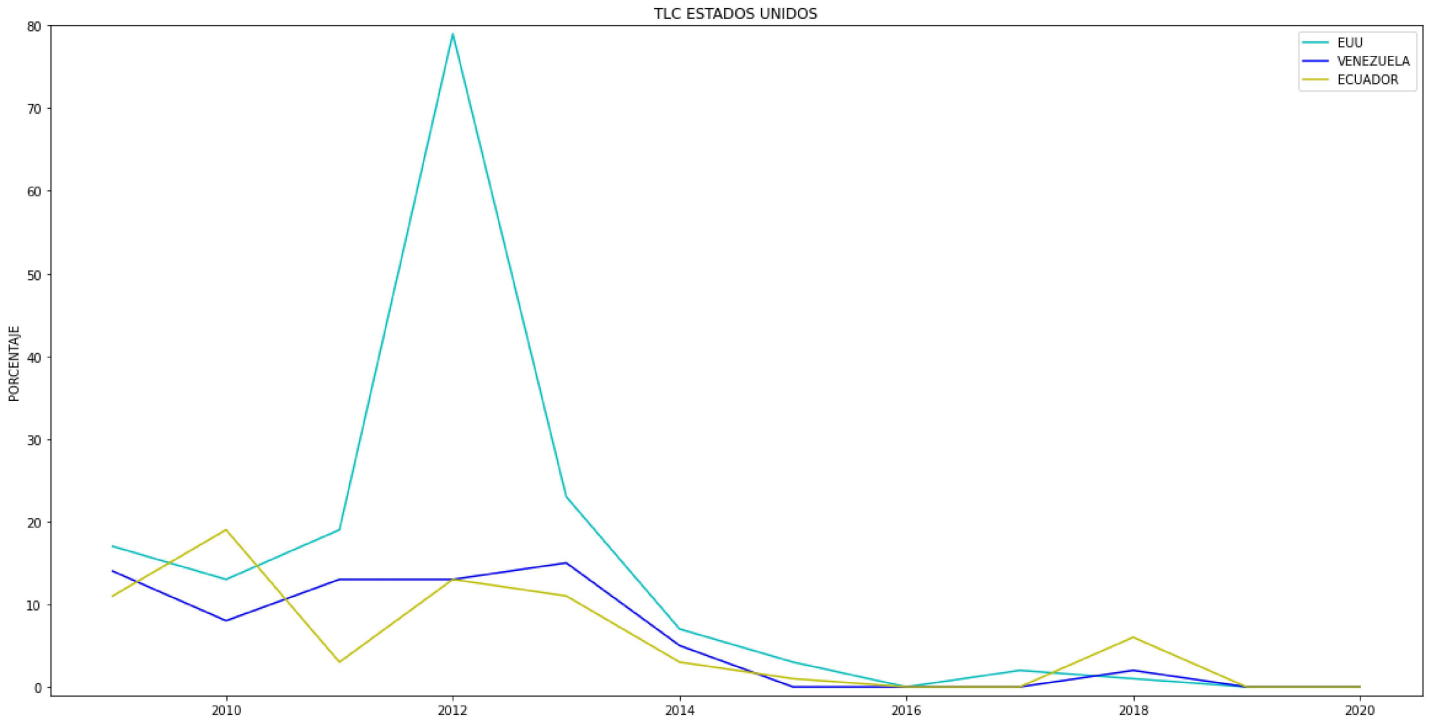
In [ ]:
```
#print(titulos[10])
#print(texto[10])

print(len(titulos))
print(len(texto))
print(len(años))
```

In [ ]:
```
texto[7]
```

In [29]:

```
Euu_ctlc =[]
Ven_ctlc =[]
Ecu_ctlc =[]
nE = 0
nV = 0
nEc = 0
actual = 2020
for i in range(2924):

    if(actual!=2007):

        añ = int(años[i][1])

        textotp = str(texto[i])
        #textotp = str(titulos[i])

        if  añ == actual :

                nE = nE +textotp.count('Estados Unidos')
                nV = nV +textotp.count('Venezuela')
                nEc = nEc +textotp.count('Ecuador')
        else:

            Euu_ctlc.append([actual,nE])
            Ven_ctlc.append([actual,nV])
            Ecu_ctlc.append([actual,nEc])
            nE = 0
            nV = 0
            nEc = 0
            actual = actual-1
            i=i-1

    else:
        print(actual,"notengo noticias")
        i=i*2020

         #print(año[i][1],": ",textotp.count('Estados Unidos'))
```

In [30]:
```
x=[]
y= []

x2=[]
x3=[]
for i in range(len(Euu_ctlc)):
    x.append(Euu_ctlc[11-i][1])
    x2.append(Ven_ctlc[11-i][1])
    x3.append(Ecu_ctlc[11-i][1])
    y.append(Euu_ctlc[11-i][0])

s=len(y)
plt.figure(figsize=(20,10))

plt.plot(y, x,'c')
plt.plot(y, x2,'b')
plt.plot(y, x3,'y')
plt.legend(['EUU','VENEZUELA','ECUADOR'], loc=1)
plt.title("TLC ESTADOS UNIDOS")
plt.ylabel("PORCENTAJE")
plt.ylim(min(x)-1, max(x)+1)

plt.show()
```



In [ ]:
```

```

/