

# Microsoft Malware detection

## 1. Business/Real-world Problem

### 1.1. What is Malware?

The term malware is a contraction of malicious software. Put simply, malware is any piece of software that was written with the intent of doing harm to data, devices or to people.

Source: <https://www.avg.com/en/signal/what-is-malware>

### 1.2. Problem Statement

In the past few years, the malware industry has grown very rapidly that, the syndicates invest heavily in technologies to evade traditional protection, forcing the anti-malware groups/communities to build more robust softwares to detect and terminate these attacks. The major part of protecting a computer system from a malware attack is to **identify whether a given piece of file/software is a malware**.

### 1.3 Source/Useful Links

Microsoft has been very active in building anti-malware products over the years and it runs its anti-malware utilities over 150 million computers around the world. This generates tens of millions of daily data points to be analyzed as potential malware. In order to be effective in analyzing and classifying such large amounts of data, we need to be able to group them into groups and identify their respective families.

This dataset provided by Microsoft contains about 9 classes of malware. ,

Source: <https://www.kaggle.com/c/malware-classification>

### 1.4. Real-world/Business objectives and constraints.

1. Minimize multi-class error.
2. Multi-class probability estimates.
3. Malware detection should not take hours and block the user's computer. It should finish in a few seconds or a minute.

## 2. Machine Learning Problem

### 2.1. Data

#### 2.1.1. Data Overview

- Source : <https://www.kaggle.com/c/malware-classification/data>
- For every malware, we have two files
  1. .asm file (read more: <https://www.reviversoft.com/file-extensions/asm>)
  2. .bytes file (the raw data contains the hexadecimal representation of the file's binary content, without the PE header)
- Total train dataset consists of 200GB data out of which 50GB of data is .bytes files and 150GB of data is .asm files:
- Lots of Data for a single-box/computer.
- There are total 10,868 .bytes files and 10,868 asm files total 21,736 files
- There are 9 types of malwares (9 classes) in our given data
- Types of Malware:
  1. Ramnit

1. Xanadu
2. Lollipop
3. Kelihos\_ver3
4. Vundo
5. Simda
6. Tracur
7. Kelihos\_ver1
8. Obfuscator.ACY
9. Gatak

## 2.1.2. Example Data Point

### .asm file

```

.text:00401000          assume es:nothing, ss:nothing, ds:_data,
s:nothing, gs:nothing
.text:00401000 56          push    esi
                            lea     eax, [esp+8]
.text:00401001 8D 44 24 08          push    eax
                            mov     esi, ecx
                            call    ???
0exception@std@@QAE@ABQBD@Z ; std::exception::exception(char const * const &)
.text:0040100D C7 06 08 BB 42 00          mov     dword ptr [esi], offset c
f_42BB08
.text:00401013 8B C6          mov     eax, esi
.text:00401015 5E          pop    esi
                            retn   4
.text:00401016 C2 04 00          ; -----
                            ; -----
.text:00401019 CC CC CC CC CC CC CC CC          align 10h
.text:00401020 C7 01 08 BB 42 00          mov     dword ptr [ecx], offset c
f_42BB08
.text:00401026 E9 26 1C 00 00          jmp    sub_402C51
.text:00401026          ; -----
                            ; -----
.text:0040102B CC CC CC CC CC CC          align 10h
.text:00401030 56          push    esi
                            mov     esi, ecx
                            mov     dword ptr [esi], offset c
f_42BB08
.text:00401039 E8 13 1C 00 00          call   sub_402C51
.text:0040103E F6 44 24 08 01          test   byte ptr [esp+8], 1
                            jz    short loc_40104E
.text:00401043 74 09          push    esi
                            call   ??3@YAXPAX@Z      ; operator
.delete(void *)
.text:0040104B 83 C4 04          add    esp, 4
.text:0040104E          loc_40104E:           ; CODE XREF:
                            ; -----
.text:00401043 j
.text:0040104E 8B C6          mov     eax, esi
.text:00401050 5E          pop    esi
                            retn   4
.text:00401051 C2 04 00          ; -----
                            ; -----

```



### .bytes file

```

00401000 00 00 80 40 40 28 00 1C 02 42 00 C4 00 20 04 20
00401010 00 00 20 09 2A 02 00 00 00 00 8E 10 41 0A 21 01
00401020 40 00 02 01 00 90 21 00 32 40 00 1C 01 40 C8 18
00401030 40 82 02 63 20 00 00 09 10 01 02 21 00 82 00 04
00401040 82 20 08 83 00 08 00 00 00 02 00 60 80 10 80
00401050 18 00 00 20 A9 00 00 00 04 04 78 01 02 70 90

```

```
00401060 00 02 00 08 20 12 00 00 00 40 10 00 80 00 40 19
00401070 00 00 00 11 20 80 04 80 10 00 20 00 00 25 00
00401080 00 00 01 00 00 04 00 10 02 C1 80 80 00 20 20 00
00401090 08 A0 01 01 44 28 00 00 08 10 20 00 02 08 00 00
004010A0 00 40 00 00 00 34 40 40 00 04 00 08 80 08 00 08
004010B0 10 00 40 00 68 02 40 04 E1 00 28 14 00 08 20 0A
004010C0 06 01 02 00 40 00 00 00 00 00 00 20 00 02 00 04
004010D0 80 18 90 00 00 10 A0 00 45 09 00 10 04 40 44 82
004010E0 90 00 26 10 00 00 04 00 82 00 00 00 20 40 00 00
004010F0 B4 00 00 40 00 02 20 25 08 00 00 00 00 00 00 00 00
00401100 08 00 00 50 00 08 40 50 00 02 06 22 08 85 30 00
00401110 00 80 00 80 60 00 09 00 04 20 00 00 00 00 00 00 00
00401120 00 82 40 02 00 11 46 01 4A 01 8C 01 E6 00 86 10
00401130 4C 01 22 00 64 00 AE 01 EA 01 2A 11 E8 10 26 11
00401140 4E 11 8E 11 C2 00 6C 00 0C 11 60 01 CA 00 62 10
00401150 6C 01 A0 11 CE 10 2C 11 4E 10 8C 00 CE 01 AE 01
00401160 6C 10 6C 11 A2 01 AE 00 46 11 EE 10 22 00 A8 00
00401170 EC 01 08 11 A2 01 AE 10 6C 00 6E 00 AC 11 8C 00
00401180 EC 01 2A 10 2A 01 AE 00 40 00 C8 10 48 01 4E 11
00401190 0E 00 EC 11 24 10 4A 10 04 01 C8 11 E6 01 C2 00
```

## 2.2. Mapping the real-world problem to an ML problem

### 2.2.1. Type of Machine Learning Problem

There are nine different classes of malware that we need to classify a given a data point => Multi class classification problem

### 2.2.2. Performance Metric

Source: <https://www.kaggle.com/c/malware-classification#evaluation>

Metric(s):

- Multi class log-loss
- Confusion matrix

### 2.2.3. Machine Learing Objectives and Constraints

Objective: Predict the probability of each data-point belonging to each of the nine classes.

Constraints:

- Class probabilities are needed.
- Penalize the errors in class probabilites => Metric is Log-loss.
- Some Latency constraints.

## 2.3. Train and Test Dataset

Split the dataset randomly into three parts train, cross validation and test with 64%, 16%, 20% of data respectively

## 2.4. Useful blogs, videos and reference papers

<http://blog.kaggle.com/2015/05/26/microsoft-malware-winners-interview-1st-place-no-to-overfitting/>  
<https://arxiv.org/pdf/1511.04317.pdf>

First place solution in Kaggle competition: <https://www.youtube.com/watch?v=VLQTRILGz5Y>

<https://github.com/dchad/malware-detection>  
<http://vizsec.org/files/2011/Nataraj.pdf>  
[https://www.dropbox.com/sh/gfqzv0ckgs4l1bf/AAB6EelnEjvvuQg2nu\\_pIB6ua?dl=0](https://www.dropbox.com/sh/gfqzv0ckgs4l1bf/AAB6EelnEjvvuQg2nu_pIB6ua?dl=0)  
" Cross validation is more trustworthy than domain knowledge."

In [1]:

```
# !pip --version
# import matplotlib
# matplotlib.__version__
```

### 3. Exploratory Data Analysis

In [2]:

```
import warnings
warnings.filterwarnings("ignore")
import shutil
import os
import pandas as pd
import matplotlib
# matplotlib.use('nbAgg')
%matplotlib notebook
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import pickle
from sklearn.manifold import TSNE
from sklearn import preprocessing
import pandas as pd
from multiprocessing import Process # this is used for multithreading
import multiprocessing
import codecs # this is used for file operations
import random as r
from xgboost import XGBClassifier
from sklearn.model_selection import RandomizedSearchCV
from sklearn.tree import DecisionTreeClassifier
from sklearn.calibration import CalibratedClassifierCV
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import log_loss
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.feature_extraction.text import CountVectorizer
from nltk import word_tokenize
from nltk.util import ngrams
import h5py
import copy
```

In [23]:

```
#separating byte files and asm files

source = 'train'
destination_1 = 'byteFiles'
destination_2 = 'asmFiles'

# we will check if the folder 'byteFiles' exists if it not there we will create a folder with the
# same name
if not os.path.isdir(destination_1):
    os.makedirs(destination_1)
if not os.path.isdir(destination_2):
    os.makedirs(destination_2)

# os.makedirs('abc')
# shutil.move('a/abc.txt', 'abc/abc.txt')

# if we have folder called 'train' (train folder contains both .asm files and .bytes files) we wil
# l rename it 'asmFiles'
# for every file that we have in our 'asmFiles' directory we check if it is ending with .bytes, if
# yes we will move it to
```

```

# so by the end of this snippet we will separate all the .byte files and .asm files
if os.path.isdir(source):
    data_files = os.listdir(source)
    for file in data_files:
        print(file)
        if (file.endswith("bytes")):
            shutil.move(source+'/'+file,destination_1)
        if (file.endswith("asm")):
            shutil.move(source+'/'+file,destination_2)

```

### 3.1. Distribution of malware classes in whole data set

In [3]:

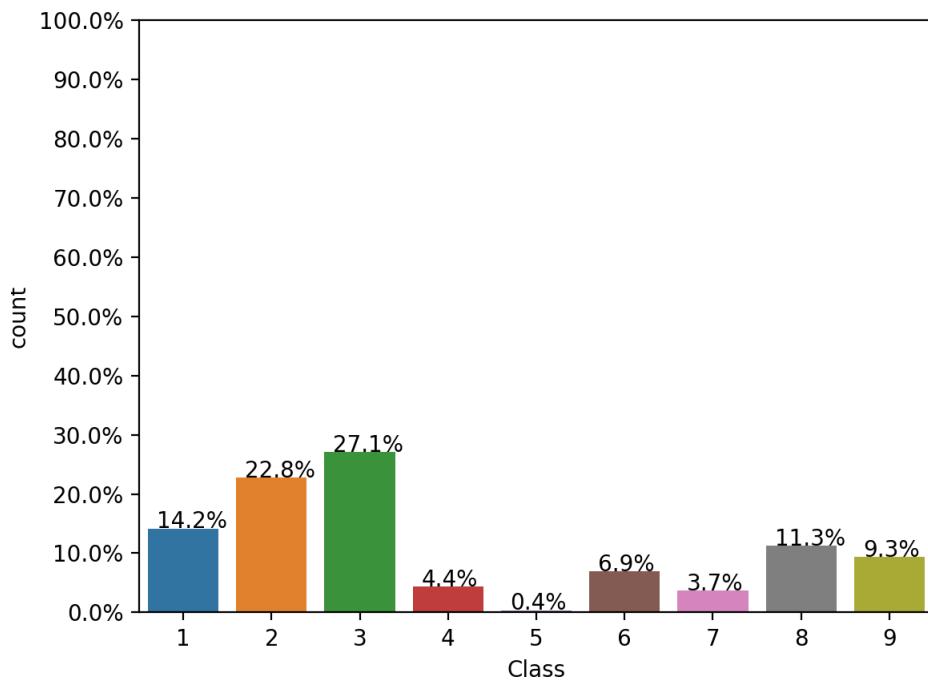
```

# close previous plot to reden new one
plt.close()
Y=pd.read_csv("trainLabels.csv")
total = len(Y)*1.
ax=sns.countplot(x="Class", data=Y)
for p in ax.patches:
    ax.annotate('{:.1f}%'.format(100*p.get_height()/total), (p.get_x()+0.1, p.get_height()+5))

#put 11 ticks (therefore 10 steps), from 0 to the total number of rows in the dataframe
ax.yaxis.set_ticks(np.linspace(0, total, 11))

#adjust the ticklabel to the desired format, without changing the position of the ticks.
ax.set_yticklabels(map('{:.1f}%'.format, 100*ax.yaxis.get_majorticklocs()/total))
plt.show()

```



### 3.2. Feature extraction

#### 3.2.1 File size of byte files as a feature

In [18]:

```

# file sizes of byte files
files=os.listdir('byteFiles')

```

```

filenames=Y['Id'].tolist()
class_y=Y['Class'].tolist()
class_bytes=[]
sizebytes=[]
fnames=[]
for file in files:
    # print(os.stat('byteFiles/0A32eTdBKayjCWhZqDOQ.txt'))
    # os.stat_result(st_mode=33206, st_ino=1125899906874507, st_dev=3561571700, st_nlink=1,
st_uid=0, st_gid=0,
    # st_size=3680109, st_atime=1519638522, st_mtime=1519638522, st_ctime=1519638522)
    # read more about os.stat: here https://www.tutorialspoint.com/python/os_stat.htm
    statinfo=os.stat('byteFiles/'+file)
    # split the file name at '.' and take the first part of it i.e the file name
    file=file.split('.')[0]
    if any(file == filename for filename in filenames):
        i=filenames.index(file)
        class_bytes.append(class_y[i])
        # converting into Mb's
        sizebytes.append(statinfo.st_size/(1024.0*1024.0))
        fnames.append(file)
data_size_byte=pd.DataFrame({'ID':fnames,'size':sizebytes,'Class':class_bytes})
print (data_size_byte.head())

```

	ID	size	Class
0	FGCV8wbAuQp1HiSd7kvm	3.972656	7
1	EmVYF0ZrIBWCGL5168XH	0.644531	1
2	CywHbj2cUaVYgp9xhIDk	0.222656	6
3	eF5UtAWbwBrYJ1RVpNIk	0.363281	8
4	5scq0VdirPRF8DxUKbh1	0.632812	9

In [19]:

```
# write size data in csv file
data_size_byte.to_csv('data_size_byte.csv')
```

In [3]:

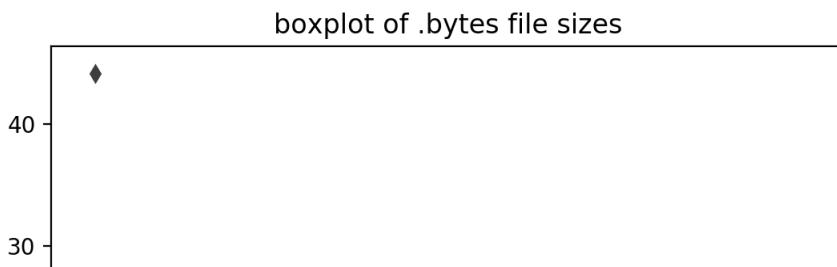
```
data_size_byte = pd.read_csv('data_size_byte.csv')
print (data_size_byte.head())
```

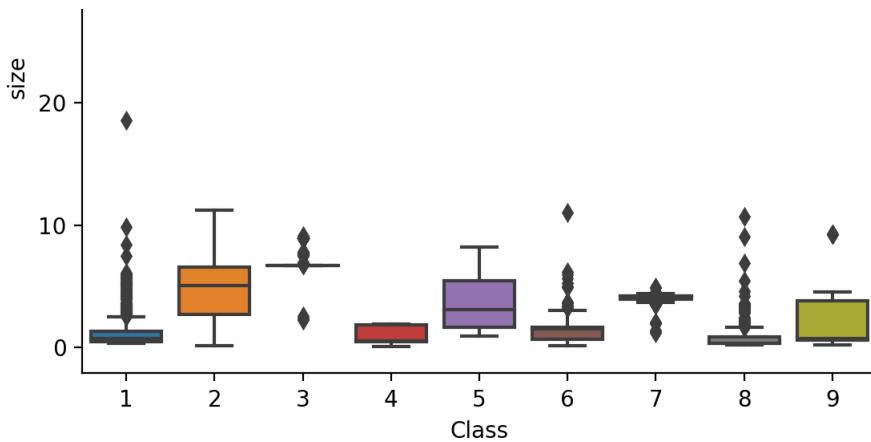
	Unnamed: 0	ID	size	Class
0	0	FGCV8wbAuQp1HiSd7kvm	3.972656	7
1	1	EmVYF0ZrIBWCGL5168XH	0.644531	1
2	2	CywHbj2cUaVYgp9xhIDk	0.222656	6
3	3	eF5UtAWbwBrYJ1RVpNIk	0.363281	8
4	4	5scq0VdirPRF8DxUKbh1	0.632812	9

### 3.2.2 box plots of file size (.byte files) feature

In [19]:

```
plt.close()
#boxplot of byte files
ax = sns.boxplot(x="Class", y="size", data=data_size_byte)
# print(ax)
plt.title("boxplot of .bytes file sizes")
plt.show()
```





### 3.2.3 feature extraction from byte files

In [20]:

```
# time taking task, take approx. 30-45 mins
#removal of address from byte files
# contents of .byte files
#
#00401000 56 8D 44 24 08 50 8B F1 E8 1C 1B 00 00 C7 06 08
#
#we remove the starting address 00401000

files = os.listdir('byteFiles')
filenames=[]
array=[]
for file in files:
    if(file.endswith("bytes")):
        file=file.split('.')[0]
        text_file = open('byteFiles/'+file+'.txt', 'w+')
        with open('byteFiles/'+file+'.bytes','r') as fp:
            lines=""
            for line in fp:
                a=line.rstrip().split(" ")[1:]
                b=' '.join(a)
                b=b+"\n"
                text_file.write(b)
            fp.close()
            os.remove('byteFiles/'+file+'.bytes')
        text_file.close()

files = os.listdir('byteFiles')
filenames2=[]
feature_matrix = np.zeros((len(files),257), dtype=int)
k=0

#program to convert into bag of words of bytefiles
#this is custom-built bag of words this is unigram bag of words
byte_feature_file=open('result.csv','w+')
byte_feature_file.write("ID,0,1,2,3,4,5,6,7,8,9,0a,0b,0c,0d,0e,0f,10,11,12,13,14,15,16,17,18,19,1a,1c,1d,1e,1f,20,21,22,23,24,25,26,27,28,29,2a,2b,2c,2d,2e,2f,30,31,32,33,34,35,36,37,38,39,3a,3b,3c,e,3f,40,41,42,43,44,45,46,47,48,49,4a,4b,4c,4d,4e,4f,50,51,52,53,54,55,56,57,58,59,5a,5b,5c,5d,5e,f,61,62,63,64,65,66,67,68,69,6a,6b,6c,6d,6e,6f,70,71,72,73,74,75,76,77,78,79,7a,7b,7c,7d,7e,7f,80,81,83,84,85,86,87,88,89,8a,8b,8c,8d,8e,8f,90,91,92,93,94,95,96,97,98,99,9a,9b,9c,9d,9e,9f,a0,a1,a2,a3,5,a6,a7,a8,a9,aa,ab,ac,ad,ae,af,b0,b1,b2,b3,b4,b5,b6,b7,b8,b9,ba,bb,bc,bd,be,bf,c0,c1,c2,c3,c4,c5,c,c8,c9,ca,cc,cd,ce,cf,d0,d1,d2,d3,d4,d5,d6,d7,d8,d9,da,db,dc,dd,de,df,e0,e1,e2,e3,e4,e5,e6,e7,e8,ea,eb,ec,ed,ee,ef,f0,f1,f2,f3,f4,f5,f6,f7,f8,f9,fa,fb,fc,fd,fe,ff,??")
byte_feature_file.write("\n")
for file in files:
    filenames2.append(file)
    byte_feature_file.write(file+",")
    if(file.endswith("txt")):
        with open('byteFiles/'+file,"r") as byte_file:
            for lines in byte_file:
                line=lines.rstrip().split(" ")
                for hex_code in line:
                    if hex_code=='??':
                        byte_feature_file.write("0,")
                    else:
                        byte_feature_file.write(str(hex_code)+",")
                byte_feature_file.write("\n")
byte_feature_file.close()
```

```

        feature_matrix[k][256] += 1
    else:
        feature_matrix[k][int(hex_code, 16)] += 1
byte_file.close()
for i, row in enumerate(feature_matrix[k]):
    if i != len(feature_matrix[k]) - 1:
        byte_feature_file.write(str(row) + ",")
    else:
        byte_feature_file.write(str(row))
byte_feature_file.write("\n")

k += 1

```

```
byte_feature_file.close()
```

In [4]:

```

byte_features = pd.read_csv("result.csv")
byte_features['ID'] = byte_features['ID'].str.split('.').str[0]
byte_features.head(2)

```

Out[4]:

	ID	0	1	2	3	4	5	6	7	8	...	f7	f8	f9	fa	fb	fc	fd	fe	ff
0	FGCV8wbAuQp1HiSd7kvm	5784	58	24	32	26	22	7	3	111	...	20	10	9	3	8	7	9	19	613
1	EmVYF0ZrlBWCGGL5168XH	24597	1258	747	907	1447	566	589	719	1117	...	890	823	675	621	683	946	906	815	5910

2 rows × 258 columns

In [5]:

```
data_size_byte.head(2)
```

Out[5]:

	Unnamed: 0	ID	size	Class
0	0	FGCV8wbAuQp1HiSd7kvm	3.972656	7
1	1	EmVYF0ZrlBWCGGL5168XH	0.644531	1

In [6]:

```

byte_features_with_size = byte_features.merge(data_size_byte, on='ID')
byte_features_with_size.to_csv("result_with_size.csv")
byte_features_with_size.head(2)

```

Out[6]:

	ID	0	1	2	3	4	5	6	7	8	...	fa	fb	fc	fd	fe	ff	??	Unnam
0	FGCV8wbAuQp1HiSd7kvm	5784	58	24	32	26	22	7	3	111	...	3	8	7	9	19	613	1377460	
1	EmVYF0ZrlBWCGGL5168XH	24597	1258	747	907	1447	566	589	719	1117	...	621	683	946	906	815	5910	4776	

2 rows × 261 columns

In [7]:

```

# https://stackoverflow.com/a/29651514
def normalize(df):
    result1 = df.copy()
    for feature_name in df.columns:
        if (str(feature_name) != str('ID') and str(feature_name) != str('Class')):
            max_value = df[feature_name].max()
            min_value = df[feature_name].min()
            result1[feature_name] = (df[feature_name] - min_value) / (max_value - min_value)
    return result1
result = normalize(byte_features_with_size)

```

In [8]:

```
result.head(2)
```

Out[8]:

	ID	0	1	2	3	4	5	6	7	8	...	fa
0	FGCV8wbAuQp1HiSd7kvm	0.002525	0.000082	0.000013	0.000017	0.000016	0.000012	0.000004	0.000003	0.000099	...	0.000012
1	EmVYF0ZrlBWCGl5168XH	0.010740	0.001771	0.000416	0.000489	0.000886	0.000320	0.000332	0.000662	0.000994	...	0.002535

2 rows × 261 columns

In [9]:

```
data_y = result['Class']
result.head()
```

Out[9]:

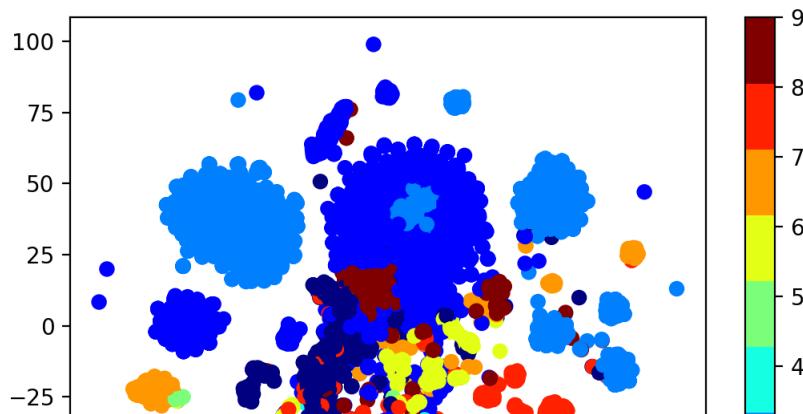
	ID	0	1	2	3	4	5	6	7	8	...	fa
0	FGCV8wbAuQp1HiSd7kvm	0.002525	0.000082	0.000013	0.000017	0.000016	0.000012	0.000004	0.000003	0.000099	...	0.000012
1	EmVYF0ZrlBWCGl5168XH	0.010740	0.001771	0.000416	0.000489	0.000886	0.000320	0.000332	0.000662	0.000994	...	0.002535
2	CywHbj2cUaVYgp9xhlDk	0.005374	0.000624	0.000130	0.000249	0.000129	0.000100	0.000105	0.000168	0.000174	...	0.000408
3	eF5UtAWbwBrYJIRVpNIk	0.008818	0.000957	0.000176	0.000247	0.000174	0.000207	0.000123	0.000221	0.000222	...	0.000976
4	5scq0VdirPRF8DxUKbh1	0.037465	0.000991	0.000251	0.000246	0.000315	0.000366	0.000252	0.000369	0.000447	...	0.001527

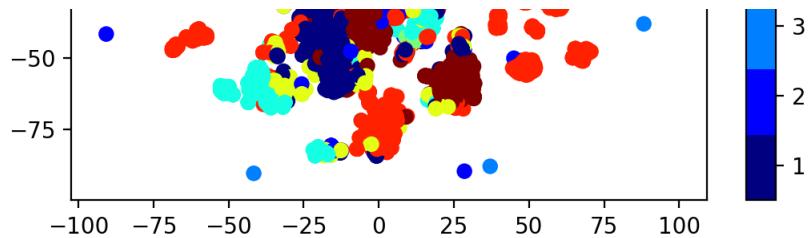
5 rows × 261 columns

### 3.2.4 Multivariate Analysis

In [14]:

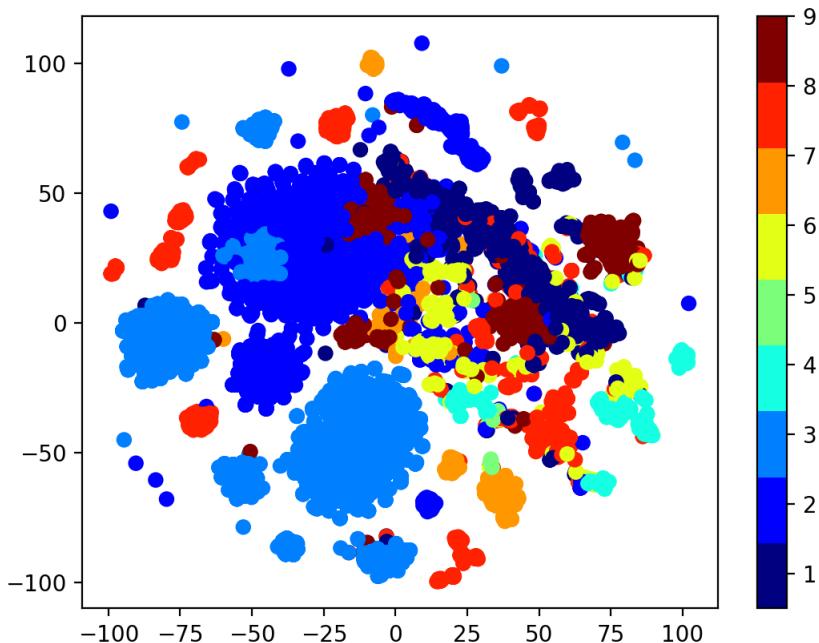
```
plt.close()
#multivariate analysis on byte files
#this is with perplexity 50
xtsne=TSNE(perplexity=50)
results=xtsne.fit_transform(result.drop(['ID','Class'], axis=1))
vis_x = results[:, 0]
vis_y = results[:, 1]
plt.scatter(vis_x, vis_y, c=data_y, cmap=plt.cm.get_cmap("jet", 9))
plt.colorbar(ticks=range(10))
plt.clim(0.5, 9)
plt.show()
```





In [15]:

```
plt.close()
#this is with perplexity 30
xtsne=TSNE(perplexity=30)
results=xtsne.fit_transform(result.drop(['ID','Class'], axis=1))
vis_x = results[:, 0]
vis_y = results[:, 1]
plt.scatter(vis_x, vis_y, c=data_y, cmap=plt.cm.get_cmap("jet", 9))
plt.colorbar(ticks=range(10))
plt.clim(0.5, 9)
plt.show()
```



## Train Test split

In [10]:

```
data_y = result['Class']
# split the data into test and train by maintaining same distribution of output variable 'y_true'
[stratify=y_true]
X_train, X_test, y_train, y_test = train_test_split(result.drop(['ID','Class'], axis=1), data_y,stratify=data_y,test_size=0.20)
# split the train data into train and cross validation by maintaining same distribution of output variable 'y_train' [stratify=y_train]
X_train, X_cv, y_train, y_cv = train_test_split(X_train, y_train,stratify=y_train,test_size=0.20)
```

In [11]:

```
print('Number of data points in train data:', X_train.shape[0])
print('Number of data points in test data:', X_test.shape[0])
print('Number of data points in cross validation data:', X_cv.shape[0])
```

```
Number of data points in train data: 6955
Number of data points in test data: 2174
Number of data points in cross validation data: 1739
```

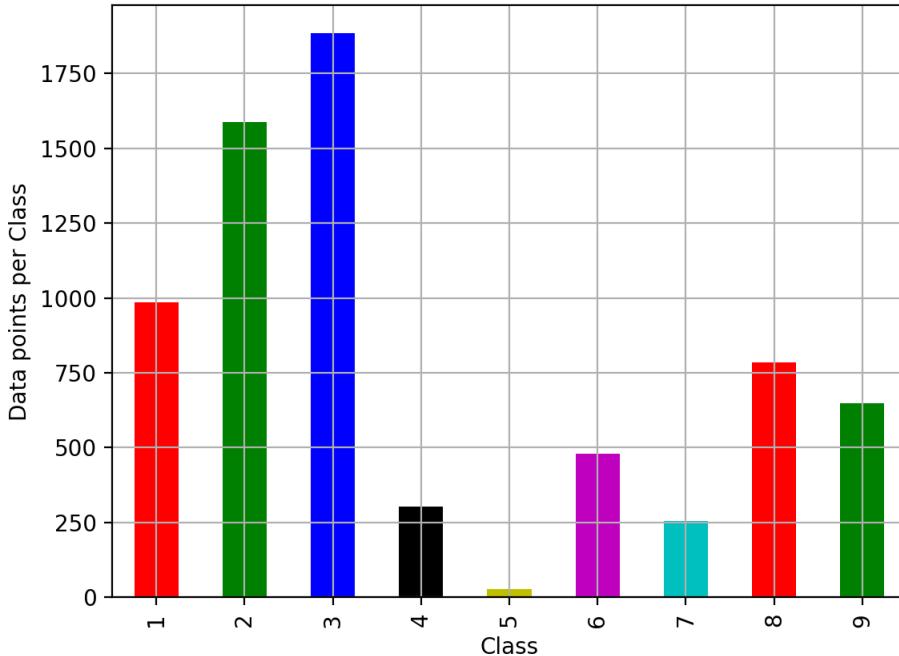
In [20]:

```
plt.close()
# it returns a dict, keys as class labels and values as the number of data points in that class
train_class_distribution = y_train.value_counts().sort_index()#.sortlevel()
test_class_distribution = y_test.value_counts().sort_index()#.sortlevel()
cv_class_distribution = y_cv.value_counts().sort_index()#.sortlevel()

# my_colors = 'rgbkymc'
my_colors = ['r', 'g', 'b', 'k', 'y', 'm', 'c'] # red, green, blue, black, etc.
train_class_distribution.plot(kind='bar', color=my_colors)
plt.xlabel('Class')
plt.ylabel('Data points per Class')
plt.title('Distribution of yi in train data')
plt.grid()
plt.show()

# ref: argsort https://docs.scipy.org/doc/numpy/reference/generated/numpy.argsort.html
# -(train_class_distribution.values): the minus sign will give us in decreasing order
sorted_yi = np.argsort(-train_class_distribution.values)
for i in sorted_yi:
    print('Number of data points in class', i+1, ':', train_class_distribution.values[i], '(', np.round((train_class_distribution.values[i]/y_train.shape[0]*100), 3), '%)')
```

Distribution of yi in train data



```
Number of data points in class 3 : 1883 ( 27.074 %)
Number of data points in class 2 : 1586 ( 22.804 %)
Number of data points in class 1 : 986 ( 14.177 %)
Number of data points in class 8 : 786 ( 11.301 %)
Number of data points in class 9 : 648 ( 9.317 %)
Number of data points in class 6 : 481 ( 6.916 %)
Number of data points in class 4 : 304 ( 4.371 %)
Number of data points in class 7 : 254 ( 3.652 %)
Number of data points in class 5 : 27 ( 0.388 %)
```

In [19]:

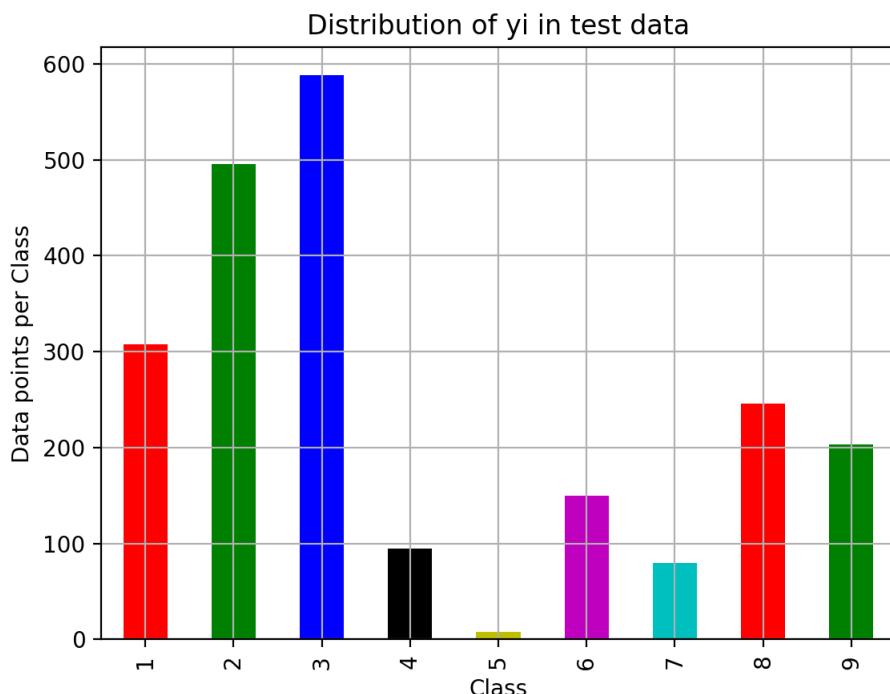
```
plt.close()
# my_colors = 'rgbkymc'
```

```

my_colors = ['r', 'g', 'b', 'k', 'y', 'm', 'c'] # red, green, blue, black, etc.
test_class_distribution.plot(kind='bar', color=my_colors)
plt.xlabel('Class')
plt.ylabel('Data points per Class')
plt.title('Distribution of yi in test data')
plt.grid()
plt.show()

print('*'*80)
# ref: argsort https://docs.scipy.org/doc/numpy/reference/generated/numpy.argsort.html
# -(train_class_distribution.values): the minus sign will give us in decreasing order
sorted_yi = np.argsort(-test_class_distribution.values)
for i in sorted_yi:
    print('Number of data points in class', i+1, ':', test_class_distribution.values[i], '(', np.round((test_class_distribution.values[i]/y_test.shape[0]*100), 3), '%)')

```




---

```

Number of data points in class 3 : 588 ( 27.047 %)
Number of data points in class 2 : 496 ( 22.815 %)
Number of data points in class 1 : 308 ( 14.167 %)
Number of data points in class 8 : 246 ( 11.316 %)
Number of data points in class 9 : 203 ( 9.338 %)
Number of data points in class 6 : 150 ( 6.9 %)
Number of data points in class 4 : 95 ( 4.37 %)
Number of data points in class 7 : 80 ( 3.68 %)
Number of data points in class 5 : 8 ( 0.368 %)

```

In [21]:

```

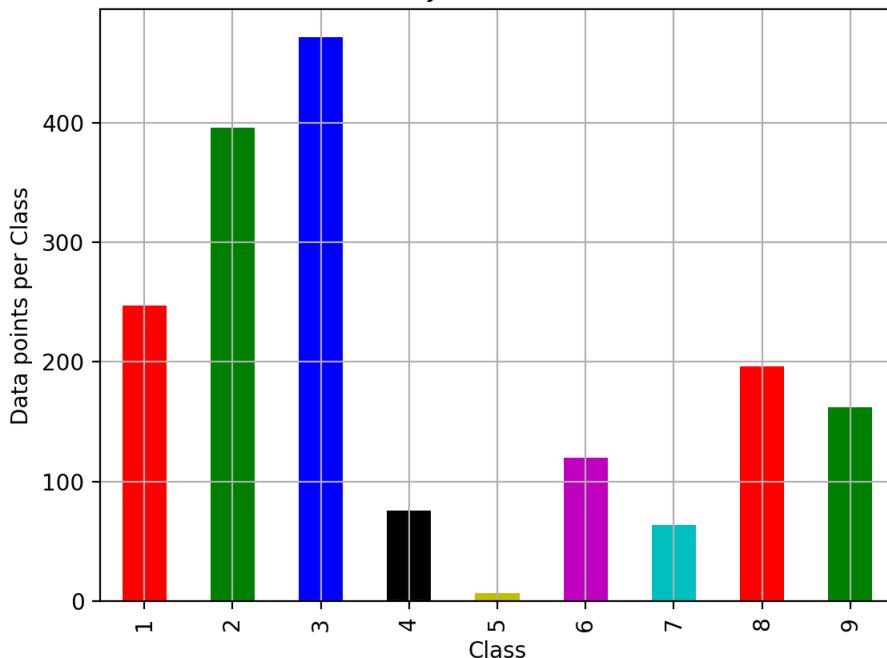
plt.close()
# my_colors = 'rgbkymc'
my_colors = ['r', 'g', 'b', 'k', 'y', 'm', 'c'] # red, green, blue, black, etc.
cv_class_distribution.plot(kind='bar', color=my_colors)
plt.xlabel('Class')
plt.ylabel('Data points per Class')
plt.title('Distribution of yi in cross validation data')
plt.grid()
plt.show()

print('*'*80)
# ref: argsort https://docs.scipy.org/doc/numpy/reference/generated/numpy.argsort.html
# -(train_class_distribution.values): the minus sign will give us in decreasing order
sorted_yi = np.argsort(-train_class_distribution.values)
for i in sorted_yi:
    print('Number of data points in class', i+1, ':', cv_class_distribution.values[i], '(', np.round((cv_class_distribution.values[i]/y_cv.shape[0]*100), 3), '%)')

```

```
((cv_class_distribution.values[i]/y_cv.shape[0]*100), 3), '%'))
```

Distribution of yi in cross validation data



```
Number of data points in class 3 : 471 ( 27.085 %)
Number of data points in class 2 : 396 ( 22.772 %)
Number of data points in class 1 : 247 ( 14.204 %)
Number of data points in class 8 : 196 ( 11.271 %)
Number of data points in class 9 : 162 ( 9.316 %)
Number of data points in class 6 : 120 ( 6.901 %)
Number of data points in class 4 : 76 ( 4.37 %)
Number of data points in class 7 : 64 ( 3.68 %)
Number of data points in class 5 : 7 ( 0.403 %)
```

In [10]:

```
def plot_confusion_matrix(test_y, predict_y):
    C = confusion_matrix(test_y, predict_y)
    print("Number of misclassified points ", (len(test_y)-np.trace(C))/len(test_y)*100)
    # C = 9,9 matrix, each cell (i,j) represents number of points of class i are predicted class j

    A = (((C.T)/(C.sum(axis=1))).T)
    # divid each element of the confusion matrix with the sum of elements in that column

    # C = [[1, 2],
    #       [3, 4]]
    # C.T = [[1, 3],
    #         [2, 4]]
    # C.sum(axis = 1) axis=0 corresonds to columns and axis=1 corresponds to rows in two
    # diamensional array
    # C.sum(axix =1) = [[3, 7]]
    # ((C.T)/(C.sum(axis=1))) = [[1/3, 3/7]
    #                             [2/3, 4/7]]

    # ((C.T)/(C.sum(axis=1))).T = [[1/3, 2/3]
    #                               [3/7, 4/7]]
    # sum of row elements = 1

    B =(C/C.sum(axis=0))
    # divid each element of the confusion matrix with the sum of elements in that row
    # C = [[1, 2],
    #       [3, 4]]
    # C.sum(axis = 0) axis=0 corresonds to columns and axis=1 corresponds to rows in two
    # diamensional array
    # C.sum(axix =0) = [[4, 6]]
    # (C/C.sum(axis=0)) = [[1/4, 2/6],
```

```

# [3/4, 4/6]

labels = [1,2,3,4,5,6,7,8,9]
cmap=sns.light_palette("green")
# representing A in heatmap format
print("-"*50, "Confusion matrix", "*"-50)
plt.figure(figsize=(10,5))
sns.heatmap(C, annot=True, cmap=cmap, fmt=".3f", xticklabels=labels, yticklabels=labels)
plt.xlabel('Predicted Class')
plt.ylabel('Original Class')
plt.show()

print("-"*50, "Precision matrix", "*"-50)
plt.figure(figsize=(10,5))
sns.heatmap(B, annot=True, cmap=cmap, fmt=".3f", xticklabels=labels, yticklabels=labels)
plt.xlabel('Predicted Class')
plt.ylabel('Original Class')
plt.show()
print("Sum of columns in precision matrix",B.sum(axis=0))

# representing B in heatmap format
print("-"*50, "Recall matrix", "*"-50)
plt.figure(figsize=(10,5))
sns.heatmap(A, annot=True, cmap=cmap, fmt=".3f", xticklabels=labels, yticklabels=labels)
plt.xlabel('Predicted Class')
plt.ylabel('Original Class')
plt.show()
print("Sum of rows in precision matrix",A.sum(axis=1))

```

## 4. Machine Learning Models

### 4.1. Machine Learning Models on bytes files

#### 4.1.1. Random Model

In [22]:

```

# we need to generate 9 numbers and the sum of numbers should be 1
# one solution is to generate 9 numbers and divide each of the numbers by their sum
# ref: https://stackoverflow.com/a/18662466/4084039

test_data_len = X_test.shape[0]
cv_data_len = X_cv.shape[0]

# we create a output array that has exactly same size as the CV data
cv_predicted_y = np.zeros((cv_data_len,9))
for i in range(cv_data_len):
    rand_probs = np.random.rand(1,9)
    cv_predicted_y[i] = ((rand_probs/sum(sum(rand_probs)))[0])
print("Log loss on Cross Validation Data using Random Model",log_loss(y_cv,cv_predicted_y, eps=1e-15))

# Test-Set error.
#we create a output array that has exactly same as the test data
test_predicted_y = np.zeros((test_data_len,9))
for i in range(test_data_len):
    rand_probs = np.random.rand(1,9)
    test_predicted_y[i] = ((rand_probs/sum(sum(rand_probs)))[0])
print("Log loss on Test Data using Random Model",log_loss(y_test,test_predicted_y, eps=1e-15))

predicted_y = np.argmax(test_predicted_y, axis=1)
plot_confusion_matrix(y_test, predicted_y+1)

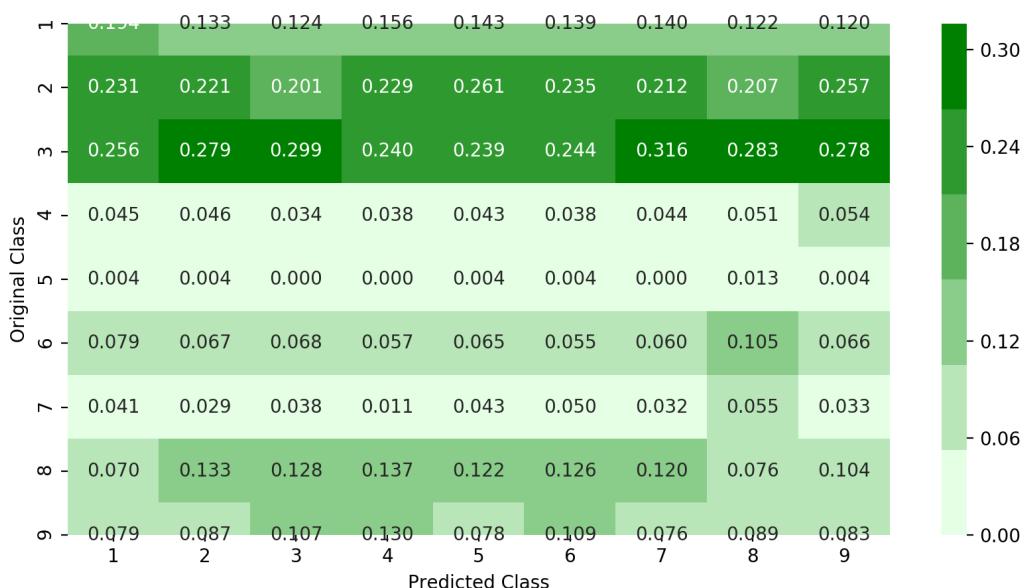
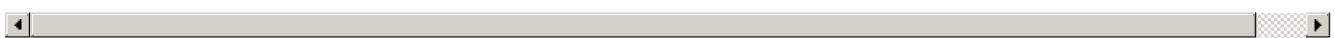
```

```

Log loss on Cross Validation Data using Random Model 2.4765227422803386
Log loss on Test Data using Random Model 2.5086309644540004
Number of misclassified points 88.9604415823367
----- Confusion matrix -----
-----
```

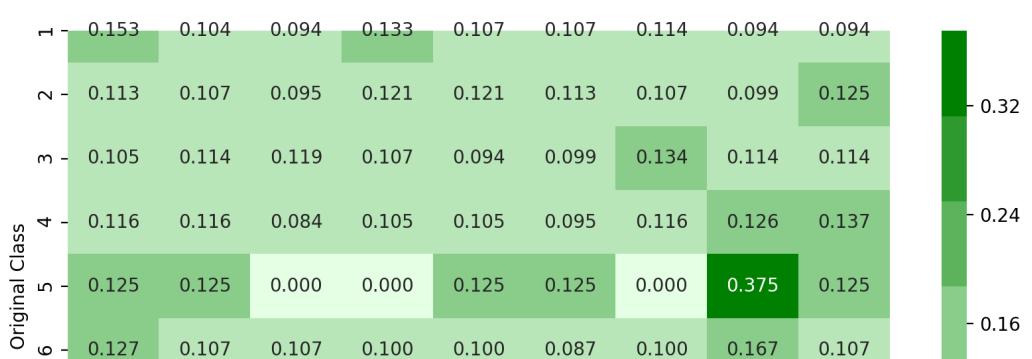


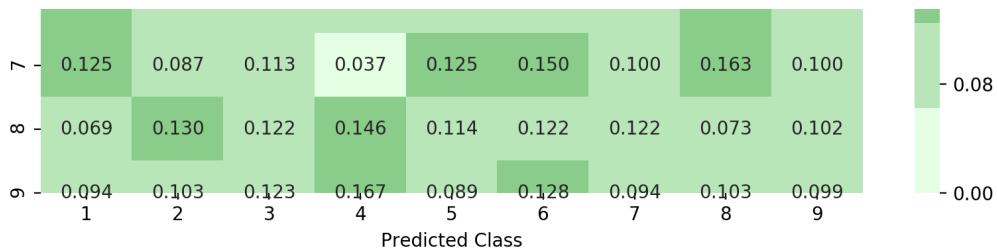
Precision matrix



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

Recall matrix





```
Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

#### 4.1.2. K Nearest Neighbour Classification

In [30]:

```
%%time
plt.close()
# find more about KNeighborsClassifier() here http://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html
# -----
# default parameter
# KNeighborsClassifier(n_neighbors=5, weights='uniform', algorithm='auto', leaf_size=30, p=2,
# metric='minkowski', metric_params=None, n_jobs=1, **kwargs)

# methods of
# fit(X, y) : Fit the model using X as training data and y as target values
# predict(X):Predict the class labels for the provided data
# predict_proba(X):Return probability estimates for the test data X.
#-----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-nearest-neighbors-geometric-intuition-with-a-toy-example-1/
#-----


# find more about CalibratedClassifierCV here at http://scikit-learn.org/stable/modules/generated/sklearn.calibration.CalibratedClassifierCV.html
# -----
# default parameters
# sklearn.calibration.CalibratedClassifierCV(base_estimator=None, method='sigmoid', cv=3)
#
# some of the methods of CalibratedClassifierCV()
# fit(X, y[, sample_weight]) Fit the calibrated model
# get_params([deep]) Get parameters for this estimator.
# predict(X) Predict the target of new samples.
# predict_proba(X) Posterior probabilities of classification
#-----
# video link:
#-----


alpha = [x for x in range(1, 15, 2)]
cv_log_error_array=[]
for i in alpha:
    k_cfl=KNeighborsClassifier(n_neighbors=i)
    k_cfl.fit(X_train,y_train)
    sig_clf = CalibratedClassifierCV(k_cfl, method="sigmoid")
    sig_clf.fit(X_train, y_train)
    predict_y = sig_clf.predict_proba(X_cv)
    cv_log_error_array.append(log_loss(y_cv, predict_y, labels=k_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for k = ',alpha[i],'is',cv_log_error_array[i])

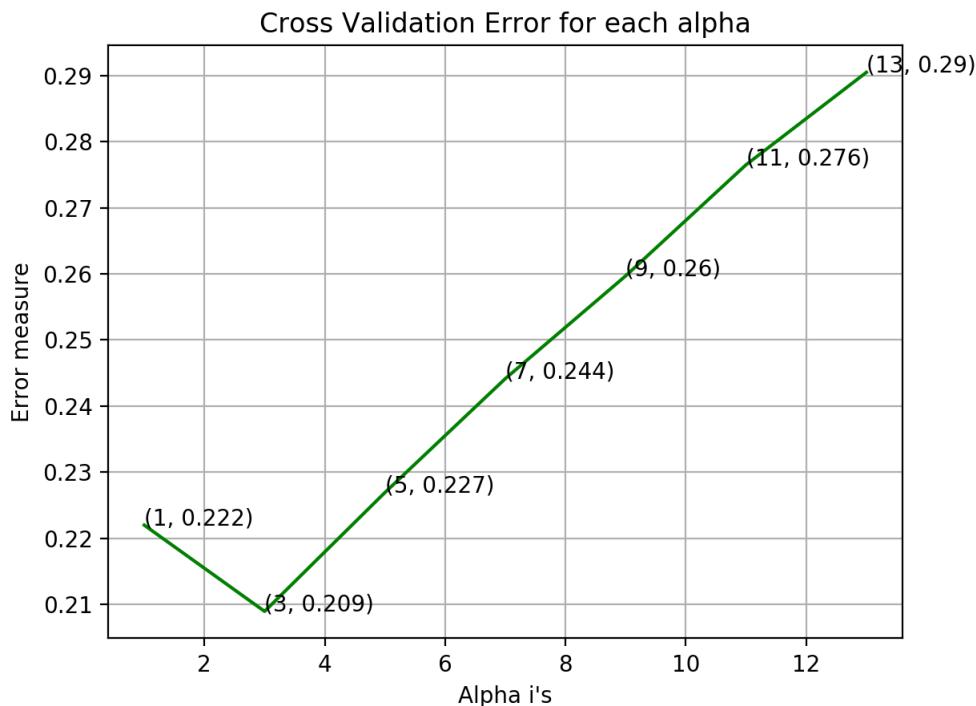
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```

```

log_loss for k = 1 is 0.22197129792560755
log_loss for k = 3 is 0.2089185330714807
log_loss for k = 5 is 0.22690358877552813
log_loss for k = 7 is 0.24413578025615393
log_loss for k = 9 is 0.25968782214603464
log_loss for k = 11 is 0.2764987380541104
log_loss for k = 13 is 0.2904920581051572

```



```

CPU times: user 57.8 s, sys: 0 ns, total: 57.8 s
Wall time: 57.8 s

```

In [32]:

```

# %%time
plt.close()
k_cfl=KNeighborsClassifier(n_neighbors=alpha[best_alpha])
k_cfl.fit(X_train,y_train)
sig_clf = CalibratedClassifierCV(k_cfl, method="sigmoid")
sig_clf.fit(X_train, y_train)

predict_y = sig_clf.predict_proba(X_train)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train, predict_y))
predict_y = sig_clf.predict_proba(X_cv)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv, predict_y))
predict_y = sig_clf.predict_proba(X_test)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test, predict_y))
plot_confusion_matrix(y_test, sig_clf.predict(X_test))

```

```

For values of best alpha = 3 The train log loss is: 0.13309130472155042
For values of best alpha = 3 The cross validation log loss is: 0.2089185330714807
For values of best alpha = 3 The test log loss is: 0.20041775276335702
Number of misclassified points 5.105795768169273
----- Confusion matrix -----
-----
```

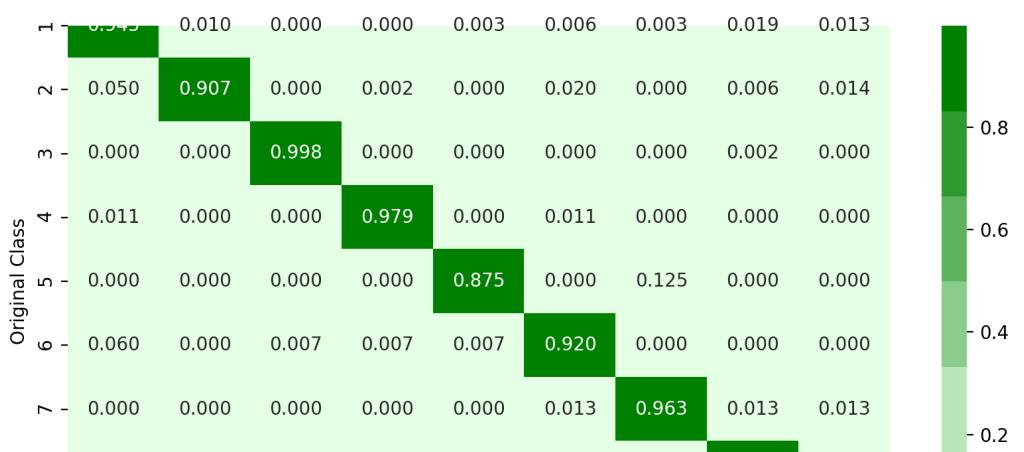


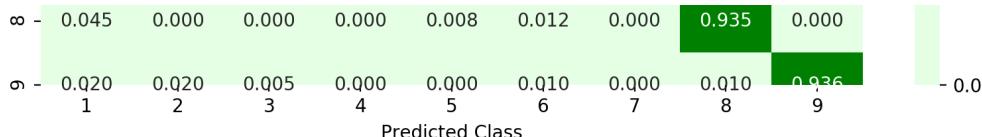
Precision matrix



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

Recall matrix





```
Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

#### 4.1.3. Logistic Regression

In [39]:

```
%%time
plt.close()
# read more about SGDClassifier() at http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html
# -----
# default parameters
# SGDClassifier(loss='hinge', penalty='l2', alpha=0.0001, l1_ratio=0.15, fit_intercept=True, max_iter=None, tol=None,
# shuffle=True, verbose=0, epsilon=0.1, n_jobs=1, random_state=None, learning_rate='optimal', eta0=0.0, power_t=0.5,
# class_weight=None, warm_start=False, average=False, n_iter=None)

# some of methods
# fit(X, y[, coef_init, intercept_init, ...]) Fit linear model with Stochastic Gradient Descent.
# predict(X) Predict class labels for samples in X.

#-----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-intuition-1/
#-----

alpha = [10 ** x for x in range(-5, 4)]
cv_log_error_array=[]
for i in alpha:
    logisticR=LogisticRegression(penalty='l2',C=i,class_weight='balanced')
    logisticR.fit(X_train,y_train)
    sig_clf = CalibratedClassifierCV(logisticR, method="sigmoid")
    sig_clf.fit(X_train, y_train)
    predict_y = sig_clf.predict_proba(X_cv)
    cv_log_error_array.append(log_loss(y_cv, predict_y, labels=logisticR.classes_, eps=1e-15))

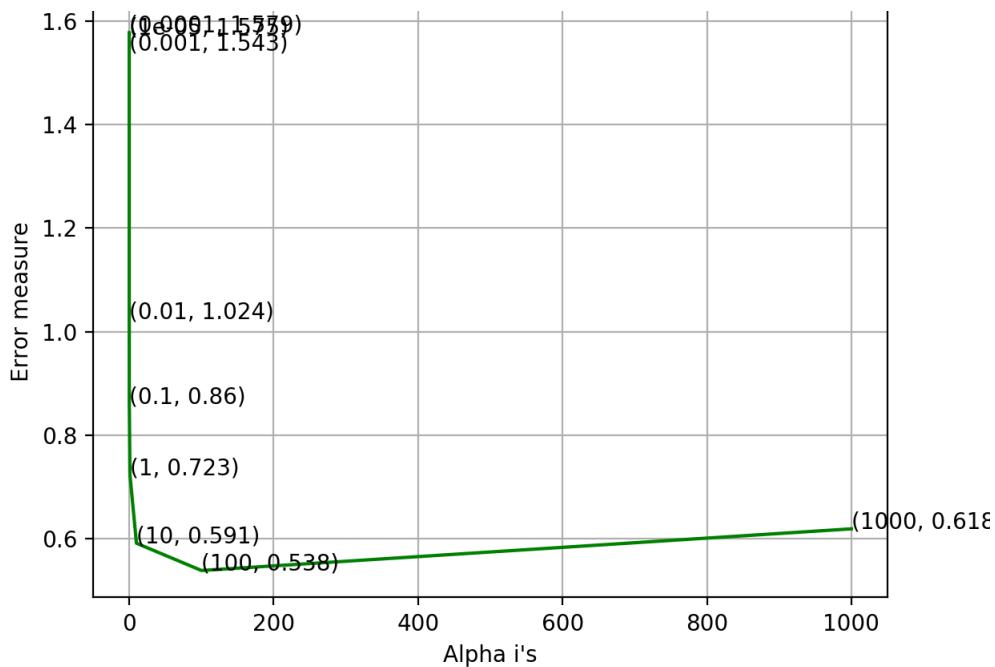
for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c = 1e-05 is 1.5750350606589754
log_loss for c = 0.0001 is 1.5790516337094596
log_loss for c = 0.001 is 1.5431918470950965
log_loss for c = 0.01 is 1.0242786142597624
log_loss for c = 0.1 is 0.8602633724481396
log_loss for c = 1 is 0.7232469080407065
log_loss for c = 10 is 0.5905983721266695
log_loss for c = 100 is 0.5378658285030614
log_loss for c = 1000 is 0.6184257388607202
```

Cross Validation Error for each alpha



CPU times: user 1min 32s, sys: 8.72 s, total: 1min 41s  
Wall time: 1min 23s

In [40]:

```
%%time
plt.close()
logisticR=LogisticRegression(penalty='l2',C=alpha[best_alpha],class_weight='balanced')
logisticR.fit(X_train,y_train)
sig_clf = CalibratedClassifierCV(logisticR, method="sigmoid")
sig_clf.fit(X_train, y_train)
pred_y=sig_clf.predict(X_test)

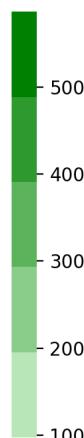
predict_y = sig_clf.predict_proba(X_train)
print ('log loss for train data',log_loss(y_train, predict_y, labels=logisticR.classes_, eps=1e-15))
predict_y = sig_clf.predict_proba(X_cv)
print ('log loss for cv data',log_loss(y_cv, predict_y, labels=logisticR.classes_, eps=1e-15))
predict_y = sig_clf.predict_proba(X_test)
print ('log loss for test data',log_loss(y_test, predict_y, labels=logisticR.classes_, eps=1e-15))
plot_confusion_matrix(y_test, sig_clf.predict(X_test))
```

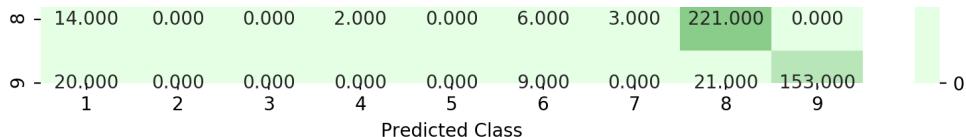
log loss for train data 0.5161568454325155  
log loss for cv data 0.5378658285030614  
log loss for test data 0.5287814318994476  
Number of misclassified points 12.051517939282428

----- Confusion matrix -----

[]

Original Class	0	1	2	3	4	5	6	7	8
0	245.000	7.000	0.000	0.000	0.000	0.000	1.000	55.000	0.000
1	23.000	447.000	6.000	2.000	0.000	10.000	0.000	5.000	3.000
2	0.000	0.000	587.000	0.000	0.000	1.000	0.000	0.000	0.000
3	1.000	0.000	0.000	94.000	0.000	0.000	0.000	0.000	0.000
4	1.000	0.000	0.000	2.000	0.000	0.000	5.000	0.000	0.000
5	8.000	0.000	3.000	21.000	0.000	90.000	0.000	22.000	6.000
6	1.000	0.000	0.000	0.000	0.000	1.000	75.000	0.000	3.000
7	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000





----- Precision matrix -----



Sum of columns in precision matrix [ 1. 1. 1. 1. nan 1. 1. 1. 1.]  
----- Recall matrix -----



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]  
CPU times: user 16.3 s, sys: 1.12 s, total: 17.4 s  
Wall time: 15 s

#### 4.1.4. Random Forest Classifier

In [41]:

```
%%time
plt.close()
# -----
# default parameters
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,
# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
# min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
# verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forest-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
train_log_error_array=[]
from sklearn.ensemble import RandomForestClassifier
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train,y_train)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train, y_train)
    predict_y = sig_clf.predict_proba(X_cv)
    cv_log_error_array.append(log_loss(y_cv, predict_y, labels=r_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

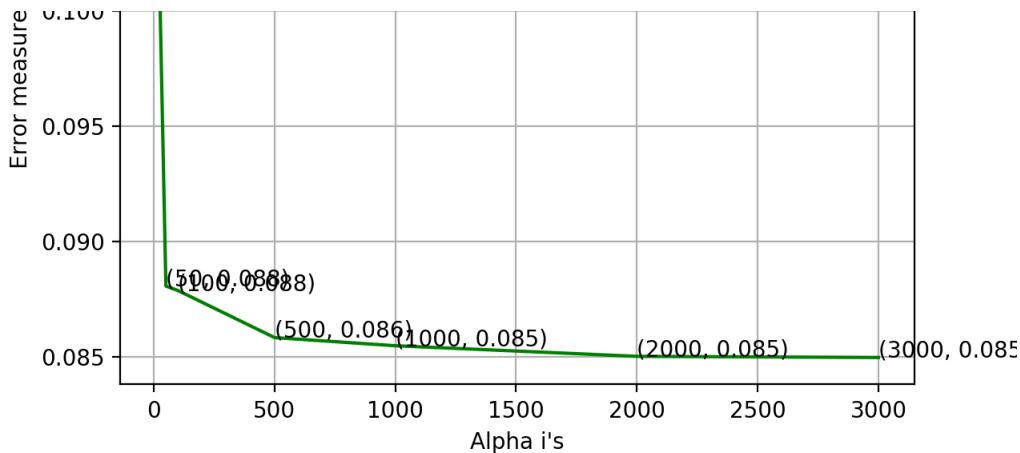
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c =  10 is  0.10832075575112872
log_loss for c =  50 is  0.08807820249822448
log_loss for c =  100 is  0.08788476126650518
log_loss for c =  500 is  0.08584374512867923
log_loss for c =  1000 is  0.08549101437528983
log_loss for c =  2000 is  0.08503123176112597
log_loss for c =  3000 is  0.08498094708786469
```

Cross Validation Error for each alpha





```
CPU times: user 29min 31s, sys: 18.4 s, total: 29min 49s
Wall time: 4min 17s
```

In [42]:

```
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train,y_train)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train, y_train)

predict_y = sig_clf.predict_proba(X_train)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train,
predict_y))
predict_y = sig_clf.predict_proba(X_cv)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv, predict_y))
predict_y = sig_clf.predict_proba(X_test)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test, p
redict_y))
plot_confusion_matrix(y_test, sig_clf.predict(X_test))
```

```
For values of best alpha = 3000 The train log loss is: 0.028706691980921458
For values of best alpha = 3000 The cross validation log loss is: 0.08498094708786469
For values of best alpha = 3000 The test log loss is: 0.06130664008293945
Number of misclassified points 1.3339466421343145
```





Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

#### 4.1.5. XgBoost Classification

In [22]:

```
%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
```

```

# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link1: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/regression-using-decision-trees-2/
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----


alpha=[10,50,100,500,1000,2000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClassifier(n_estimators=i,nthread=-1)
    x_cfl.fit(X_train,y_train)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train, y_train)
    predict_y = sig_clf.predict_proba(X_cv)
    cv_log_error_array.append(log_loss(y_cv, predict_y, labels=x_cfl.classes_, eps=1e-15))

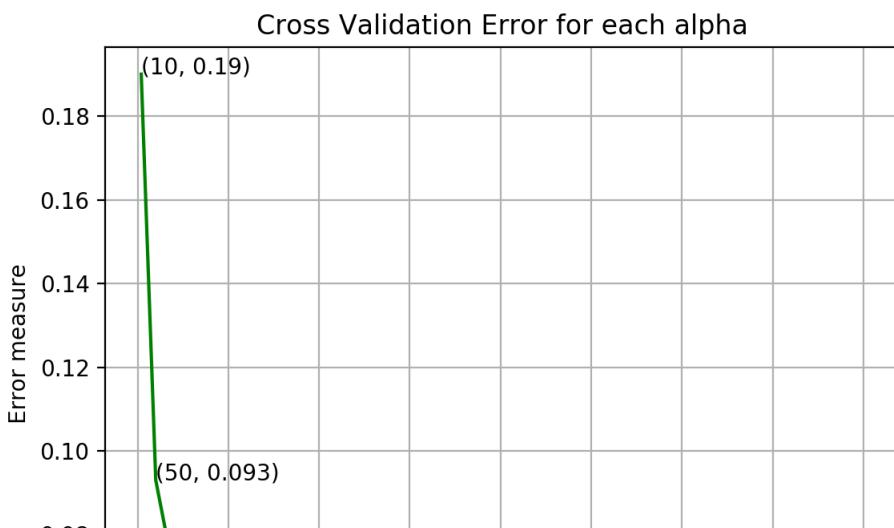
for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

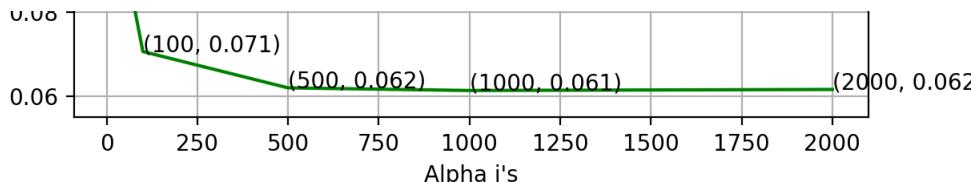
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c =  10 is  0.19005861435310809
log_loss for c =  50 is  0.09300883717009732
log_loss for c =  100 is  0.0706585262688187
log_loss for c =  500 is  0.06197621824496104
log_loss for c =  1000 is  0.0613439706923832
log_loss for c =  2000 is  0.06159227398299267

```





CPU times: user 1h 52min 8s, sys: 3min 11s, total: 1h 55min 19s  
 Wall time: 14min 27s

In [23]:

```
%time
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train,y_train)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train, y_train)

predict_y = sig_clf.predict_proba(X_train)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train,
, predict_y))
predict_y = sig_clf.predict_proba(X_cv)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv, predict_y))
predict_y = sig_clf.predict_proba(X_test)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test, p
redict_y))

For values of best alpha = 1000 The train log loss is: 0.023647153426002675
For values of best alpha = 1000 The cross validation log loss is: 0.0613439706923832
For values of best alpha = 1000 The test log loss is: 0.07842603813981379
CPU times: user 31min 41s, sys: 56.6 s, total: 32min 37s
Wall time: 4min 5s
```

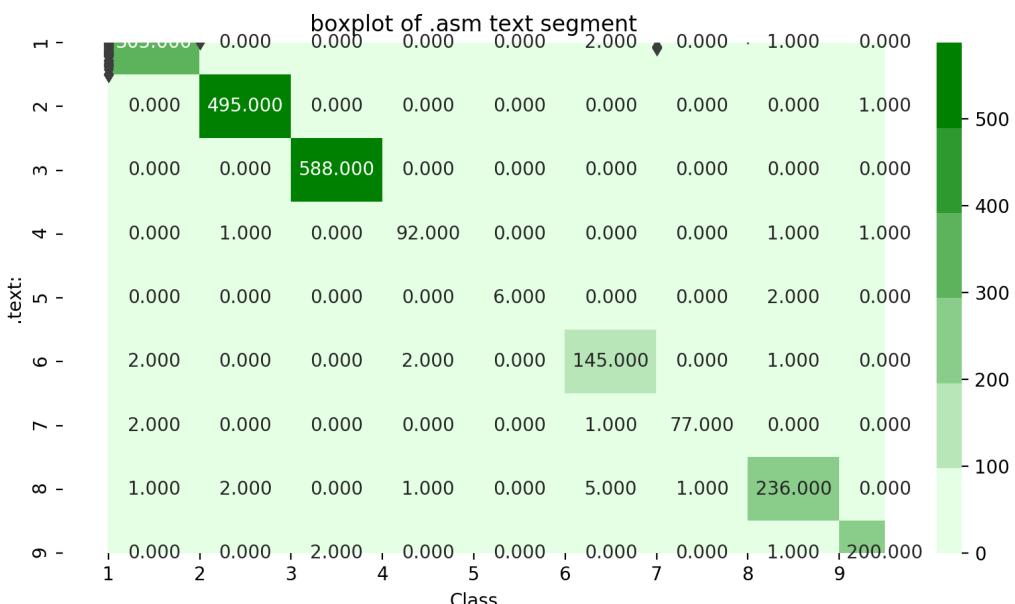
In [26]:

```
plt.close()
plot_confusion_matrix(y_test, sig_clf.predict(X_test))
```

Number of misclassified points 1.3799448022079117

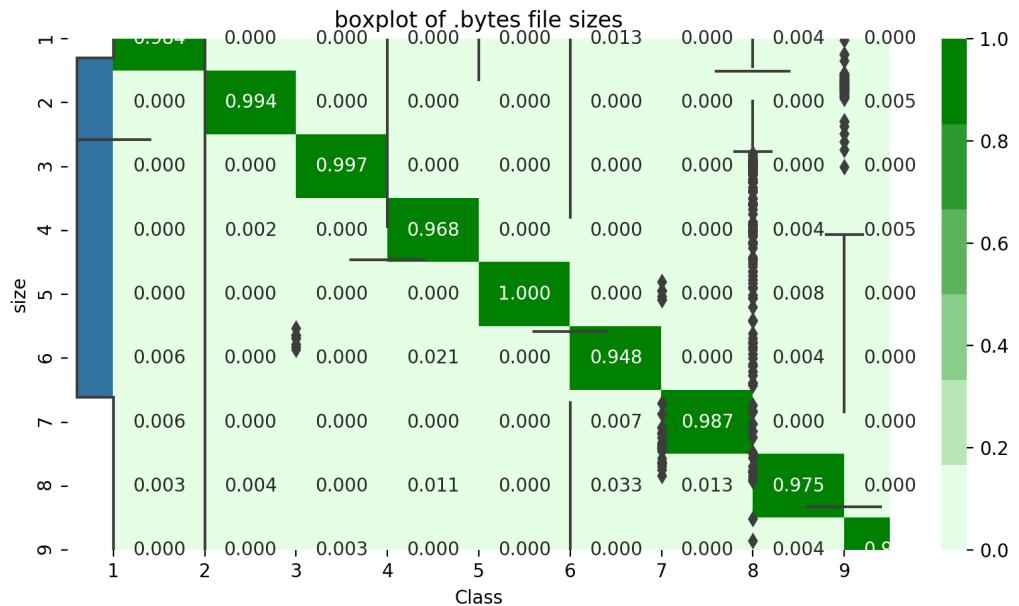
----- Confusion matrix -----

-----



----- Precision matrix -----

-----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

#### 4.1.5. XgBoost Classification with best hyper parameters using RandomSearch

In [27]:

```
# https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/
x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
```

```

        'subsample':[0.1,0.3,0.5,1]
    }
random_cfl1=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1,)
random_cfl1.fit(X_train,y_train)

```

Fitting 3 folds for each of 10 candidates, totalling 30 fits

```

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:  26.1s
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:  1.0min
[Parallel(n_jobs=-1)]: Done  19 out of  30 | elapsed:  3.6min remaining:  2.1min
[Parallel(n_jobs=-1)]: Done  23 out of  30 | elapsed:  6.4min remaining:  1.9min
[Parallel(n_jobs=-1)]: Done  27 out of  30 | elapsed: 10.5min remaining:  1.2min
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed: 11.5min finished

```

Out[27]:

```

RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                   estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                           colsample_bylevel=1,
                                           colsample_bynode=1,
                                           colsample_bytree=1, gamma=0,
                                           learning_rate=0.1, max_delta_step=0,
                                           max_depth=3, min_child_weight=1,
                                           missing=None, n_estimators=100,
                                           n_jobs=1, nthread=None,
                                           objective='binary:logistic',
                                           random_state=0, reg_alpha=0.001,
                                           seed=None, silent=None, subsample=1,
                                           verbosity=1),
                   iid='warn', n_iter=10, n_jobs=-1,
                   param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                   pre_dispatch='2*n_jobs', random_state=None, refit=True,
                   return_train_score=False, scoring=None, verbose=10)

```

In [28]:

```
print (random_cfl1.best_params_)
```

```
{'subsample': 1, 'n_estimators': 2000, 'max_depth': 3, 'learning_rate': 0.15, 'colsample_bytree': 0.3}
```

In [48]:

```
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# # max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# # scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
```

```

# ----

x_cfl=XGBClassifier(n_estimators=2000, learning_rate=0.15, colsample_bytree=0.3, max_depth=3, subsample=1)
x_cfl.fit(X_train,y_train)
c_cfl=CalibratedClassifierCV(x_cfl,method='sigmoid')
c_cfl.fit(X_train,y_train)

predict_y = c_cfl.predict_proba(X_train)
print ('train loss',log_loss(y_train, predict_y))
predict_y = c_cfl.predict_proba(X_cv)
print ('cv loss',log_loss(y_cv, predict_y))
predict_y = c_cfl.predict_proba(X_test)
print ('test loss',log_loss(y_test, predict_y))

train loss 0.023248932222341635
cv loss 0.0697311485102128
test loss 0.08183939406086423

```

## 4.2 Modeling with .asm files

There are 10868 files of asm  
 All the files make up about 150 GB  
 The asm files contains :  
 1. Address  
 2. Segments  
 3. Opcodes  
 4. Registers  
 5. function calls  
 6. APIs  
 With the help of parallel processing we extracted all the features. In parallel we can use all the cores that are present in our computer.

Here we extracted 52 features from all the asm files which are important.

We read the top solutions and handpicked the features from those papers/videos/blogs.  
 Refer:<https://www.kaggle.com/c/malware-classification/discussion>

### 4.2.1 Feature extraction from asm files

- To extract the unigram features from the .asm files we need to process ~150GB of data
- **Note: Below two cells will take lot of time (over 48 hours to complete)**
- We will provide you the output file of these two cells, which you can directly use it

In [14]:

```

#intially create five folders
#first
#second
#third
#fourth
#fifth
#this code tells us about random split of files into five folders
folder_1 ='first'
folder_2 ='second'
folder_3 ='third'
folder_4 ='fourth'
folder_5 ='fifth'
folder_6 = 'output'
for i in [folder_1,folder_2,folder_3,folder_4,folder_5,folder_6]:
    if not os.path.isdir(i):
        os.makedirs(i)

source='train/'

```

```

files = os.listdir('train')
ID=df['Id'].tolist()
data=range(0,10868)
r.shuffle(data)
count=0
for i in range(0,10868):
    if i % 5==0:
        shutil.move(source+files[data[i]],'first')
    elif i%5==1:
        shutil.move(source+files[data[i]],'second')
    elif i%5 ==2:
        shutil.move(source+files[data[i]],'third')
    elif i%5 ==3:
        shutil.move(source+files[data[i]],'fourth')
    elif i%5==4:
        shutil.move(source+files[data[i]],'fifth')

```

In [15]:

```

#http://flint.cs.yale.edu/cs421/papers/x86-asm/asm.html

def firstprocess():
    #The prefixes tells about the segments that are present in the asm files
    #There are 450 segments(approx) present in all asm files.
    #this prefixes are best segments that gives us best values.
    #https://en.wikipedia.org/wiki/Data_segment

    prefixes = ['HEADER','.text','.Pav','.idata','.data','.bss','.rdata','.edata','.rsrc',
'.tls','.reloc','.BSS','.CODE']
    #this are opcodes that are used to get best results
    #https://en.wikipedia.org/wiki/X86_instruction_listings

    opcodes = ['jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc', 'dec',
'add', 'imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror', 'rol', 'jnb','jz','rtn','lea','movzx']
    #best keywords that are taken from different blogs
    keywords = ['.dll','std::',':dword']
    #Below taken registers are general purpose registers and special registers
    #All the registers which are taken are best
    registers=['edx','esi','eax','ebx','ecx','edi','ebp','esp','eip']
    file1=open("output\asmsmallfile.txt","w+")
    files = os.listdir('first')
    for f in files:
        #filling the values with zeros into the arrays
        prefixescount=np.zeros(len(prefixes),dtype=int)
        opcodescount=np.zeros(len(opcodes),dtype=int)
        keywordcount=np.zeros(len(keywords),dtype=int)
        registerscount=np.zeros(len(registers),dtype=int)
        features=[]
        f2=f.split('.')[0]
        file1.write(f2+",")
        opcodefile.write(f2+" ")
        # https://docs.python.org/3/library/codecs.html#codecs.ignore_errors
        # https://docs.python.org/3/library/codecs.html#codecs.Codec.encode
        with codecs.open('first/'+f,encoding='cp1252',errors ='replace') as fli:
            for lines in fli:
                # https://www.tutorialspoint.com/python3/string.rstrip.htm
                line=lines.rstrip().split()
                l=line[0]
                #counting the prefixes in each and every line
                for i in range(len(prefixes)):
                    if prefixes[i] in line[0]:
                        prefixescount[i]+=1
                line=line[1:]
                #counting the opcodes in each and every line
                for i in range(len(opcodes)):
                    if any(opcodes[i]==li for li in line):
                        features.append(opcodes[i])
                        opcodescount[i]+=1
                #counting registers in the line
                for i in range(len(registers)):
                    for li in line:
                        # we will use registers only in 'text' and 'CODE' segments
                        if registers[i] in li and ('text' in l or 'CODE' in l):
                            registerscount[i]+=1
                #counting keywords in the line

```

```

        for i in range(len(keywords)):
            for li in line:
                if keywords[i] in li:
                    keywordcount[i]+=1
    #pushing the values into the file after reading whole file
    for prefix in prefixescount:
        file1.write(str(prefix)+",")
    for opcode in opcodescount:
        file1.write(str(opcode)+",")
    for register in registerscount:
        file1.write(str(register)+",")
    for key in keywordcount:
        file1.write(str(key)+",")
    file1.write("\n")
file1.close()

#same as above
def secondprocess():
    prefixes = ['HEADER','.text','.Pav','.idata','.data','.bss','.rdata','.edata','.rsrc',
'.tls','.reloc','.BSS','.CODE']
    opcodes = ['jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc', 'dec',
'add','imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror', 'rol', 'jnb','jz','rtn','lea','movz
x']
    keywords = ['.dll','std::':':dword']
    registers=['edx','esi','eax','ebx','ecx','edi','ebp','esp','eip']
    file1=open("output\mediumasmfile.txt","w+")
    files = os.listdir('second')
    for f in files:
        prefixescount=np.zeros(len(prefixes),dtype=int)
        opcodescount=np.zeros(len(opcodes),dtype=int)
        keywordcount=np.zeros(len(keywords),dtype=int)
        registerscount=np.zeros(len(registers),dtype=int)
        features=[]
        f2=f.split('.')[0]
        file1.write(f2+",")
        opcodefile.write(f2+" ")
        with codecs.open('second/'+f,encoding='cp1252',errors ='replace') as fli:
            for lines in fli:
                line=lines.rstrip().split()
                l=line[0]
                for i in range(len(prefixes)):
                    if prefixes[i] in line[0]:
                        prefixescount[i]+=1
                line=line[1:]
                for i in range(len(opcodes)):
                    if any(opcodes[i]==li for li in line):
                        features.append(opcodes[i])
                        opcodescount[i]+=1
                for i in range(len(registers)):
                    for li in line:
                        if registers[i] in li and ('text' in l or 'CODE' in l):
                            registerscount[i]+=1
                for i in range(len(keywords)):
                    for li in line:
                        if keywords[i] in li:
                            keywordcount[i]+=1
        for prefix in prefixescount:
            file1.write(str(prefix)+",")
        for opcode in opcodescount:
            file1.write(str(opcode)+",")
        for register in registerscount:
            file1.write(str(register)+",")
        for key in keywordcount:
            file1.write(str(key)+",")
        file1.write("\n")
file1.close()

# same as smallprocess() functions
def thirdprocess():
    prefixes = ['HEADER','.text','.Pav','.idata','.data','.bss','.rdata','.edata','.rsrc',
'.tls','.reloc','.BSS','.CODE']
    opcodes = ['jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc', 'dec',
'add','imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror', 'rol', 'jnb','jz','rtn','lea','movz
x']
    keywords = ['.dll','std::':':dword']
    registers=['edx','esi','eax','ebx','ecx','edi','ebp','esp','eip']

```

```

file1=open("output\largeasmfile.txt","w+")
files = os.listdir('thrid')
for f in files:
    prefixescount=np.zeros(len(prefixes),dtype=int)
    opcodescount=np.zeros(len(opcodes),dtype=int)
    keywordcount=np.zeros(len(keywords),dtype=int)
    registerscount=np.zeros(len(registers),dtype=int)
    features=[]
    f2=f.split('.')[0]
    file1.write(f2+",")
    opcodefile.write(f2+" ")
    with codecs.open('thrid/'+f,encoding='cp1252',errors ='replace') as fli:
        for lines in fli:
            line=lines.rstrip().split()
            l=line[0]
            for i in range(len(prefixes)):
                if prefixes[i] in line[0]:
                    prefixescount[i]+=1
            line=line[1:]
            for i in range(len(opcodes)):
                if any(opcodes[i]==li for li in line):
                    features.append(opcodes[i])
                    opcodescount[i]+=1
            for i in range(len(registers)):
                for li in line:
                    if registers[i] in li and ('text' in l or 'CODE' in l):
                        registerscount[i]+=1
            for i in range(len(keywords)):
                for li in line:
                    if keywords[i] in li:
                        keywordcount[i]+=1
        for prefix in prefixescount:
            file1.write(str(prefix)+",")
        for opcode in opcodescount:
            file1.write(str(opcode)+",")
        for register in registerscount:
            file1.write(str(register)+",")
        for key in keywordcount:
            file1.write(str(key)+",")
    file1.write("\n")
file1.close()

def fourthprocess():
    prefixes = ['HEADER','.text','.Pav','.idata','.data','.bss','.rdata','.edata','.rsrc',
'.tls','.reloc','.BSS','.CODE']
    opcodes = ['jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc', 'dec',
'add','imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror', 'rol', 'jnb','jz','rtn','lea','movz
x']
    keywords = ['.dll','std::':':dword']
    registers=['edx','esi','eax','ebx','ecx','edi','ebp','esp','eip']
    file1=open("output\hugeasmfile.txt","w+")
    files = os.listdir('fourth/')
    for f in files:
        prefixescount=np.zeros(len(prefixes),dtype=int)
        opcodescount=np.zeros(len(opcodes),dtype=int)
        keywordcount=np.zeros(len(keywords),dtype=int)
        registerscount=np.zeros(len(registers),dtype=int)
        features=[]
        f2=f.split('.')[0]
        file1.write(f2+",")
        opcodefile.write(f2+" ")
        with codecs.open('fourth/'+f,encoding='cp1252',errors ='replace') as fli:
            for lines in fli:
                line=lines.rstrip().split()
                l=line[0]
                for i in range(len(prefixes)):
                    if prefixes[i] in line[0]:
                        prefixescount[i]+=1
                line=line[1:]
                for i in range(len(opcodes)):
                    if any(opcodes[i]==li for li in line):
                        features.append(opcodes[i])
                        opcodescount[i]+=1
                for i in range(len(registers)):
                    for li in line:
                        if registers[i] in li and ('text' in l or 'CODE' in l):
                            registerscount[i]+=1
            for prefix in prefixescount:
                file1.write(str(prefix)+",")
            for opcode in opcodescount:
                file1.write(str(opcode)+",")
            for register in registerscount:
                file1.write(str(register)+",")
            for key in keywordcount:
                file1.write(str(key)+",")
    file1.write("\n")

```

```

        registerscount[i]+=1
    for i in range(len(keywords)):
        for li in line:
            if keywords[i] in li:
                keywordcount[i]+=1
    for prefix in prefixescount:
        file1.write(str(prefix)+",")
    for opcode in opcodescount:
        file1.write(str(opcode)+",")
    for register in registerscount:
        file1.write(str(register)+",")
    for key in keywordcount:
        file1.write(str(key)+",")
    file1.write("\n")
file1.close()

def fifthprocess():
    prefixes = ['HEADER','.text','.Pav','.idata','.data','.bss','.rdata','.edata','_.rsrc',
'.tls','.reloc','.BSS','.CODE']
    opcodes = ['jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc', 'dec',
'add','imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror', 'rol', 'jnb','jz','rtn','lea','movz
x']
    keywords = ['.dll','std::':dword]
    registers=['edx','esi','eax','ebx','ecx','edi','ebp','esp','eip']
    file1=open("output\trainasmfile.txt","w+")
    files = os.listdir('fifth/')
    for f in files:
        prefixescount=np.zeros(len(prefixes),dtype=int)
        opcodescount=np.zeros(len(opcodes),dtype=int)
        keywordcount=np.zeros(len(keywords),dtype=int)
        registerscount=np.zeros(len(registers),dtype=int)
        features=[]
        f2=f.split('.')[0]
        file1.write(f2+",")
        opcodefile.write(f2+" ")
        with codecs.open('fifth/'+f,encoding='cp1252',errors ='replace') as fli:
            for lines in fli:
                line=lines.rstrip().split()
                l=line[0]
                for i in range(len(prefixes)):
                    if prefixes[i] in line[0]:
                        prefixescount[i]+=1
                line=line[1:]
                for i in range(len(opcodes)):
                    if any(opcodes[i]==li for li in line):
                        features.append(opcodes[i])
                        opcodescount[i]+=1
                for i in range(len(registers)):
                    for li in line:
                        if registers[i] in li and ('text' in l or 'CODE' in l):
                            registerscount[i]+=1
                for i in range(len(keywords)):
                    for li in line:
                        if keywords[i] in li:
                            keywordcount[i]+=1
        for prefix in prefixescount:
            file1.write(str(prefix)+",")
        for opcode in opcodescount:
            file1.write(str(opcode)+",")
        for register in registerscount:
            file1.write(str(register)+",")
        for key in keywordcount:
            file1.write(str(key)+",")
        file1.write("\n")
file1.close()

def main():
    #the below code is used for multiprogramming
    #the number of process depends upon the number of cores present System
    #process is used to call multiprogramming
    manager=multiprocessing.Manager()
    p1=Process(target=firstprocess)
    p2=Process(target=secondprocess)
    p3=Process(target=thirdprocess)
    p4=Process(target=fourthprocess)

```

```

p5=Process(target=fifthprocess)
#p1.start() is used to start the thread execution
p1.start()
p2.start()
p3.start()
p4.start()
p5.start()
#After completion all the threads are joined
p1.join()
p2.join()
p3.join()
p4.join()
p5.join()

if __name__ == "__main__":
    main()

```

In [16]:

```

# asmoutputfile.csv(output generated from the above two cells) will contain all the extracted features from .asm files
# this file will be uploaded in the drive, you can directly use this
dfasm=pd.read_csv("asmoutputfile.csv")
Y.columns = ['ID', 'Class']
result_asm = pd.merge(dfasm, Y,on='ID', how='left')
result_asm.head()

```

Out[16]:

	ID	HEADER:	.text:	.Pav:	.idata:	.data:	.bss:	.rdata:	.edata:	.rsrc:	...	edx	esi	eax	ebx	ecx	edi	eb
0	01kcPWA9K2B0xQeS5Rju	19	744	0	127	57	0	323	0	3	...	18	66	15	43	83	0	1
1	1E93CpP60RHFNiT5Qfvn	17	838	0	103	49	0	0	0	3	...	18	29	48	82	12	0	1
2	3ekVow2ajZHbTnBcsDfX	17	427	0	50	43	0	145	0	3	...	13	42	10	67	14	0	1
3	3X2nY7iQaPBIWDrAZqJe	17	227	0	43	19	0	0	0	3	...	6	8	14	7	2	0	
4	46OZzdsSKDCFV8h7XWxf	17	402	0	59	170	0	0	0	3	...	12	9	18	29	5	0	1

5 rows × 53 columns

#### 4.2.1.1 Files sizes of each .asm file

In [17]:

```

#file sizes of byte files

files=os.listdir('asmFiles')
filenames=Y['ID'].tolist()
class_y=Y['Class'].tolist()
class_bytes=[]
sizebytes=[]
fnames=[]
for file in files:
    # print(os.stat('byteFiles/0A32eTdBKayjCWhZqDOQ.txt'))
    # os.stat_result(st_mode=33206, st_ino=1125899906874507, st_dev=3561571700, st_nlink=1,
    st_uid=0, st_gid=0,
    # st_size=3680109, st_atime=1519638522, st_mtime=1519638522, st_ctime=1519638522)
    # read more about os.stat: here https://www.tutorialspoint.com/python/os_stat.htm
    statinfo=os.stat('asmFiles/'+file)
    # split the file name at '.' and take the first part of it i.e the file name
    file=file.split('.')[0]
    if any(file == filename for filename in filenames):
        i=filenames.index(file)
        class_bytes.append(class_y[i])
        # converting into Mb's
        sizebytes.append(statinfo.st_size/(1024.0*1024.0))
        fnames.append(file)
asm_size_byte=pd.DataFrame({'ID':fnames,'size':sizebytes,'Class':class_bytes})
print (asm_size_byte.head())

```

ID	size	Class
01kcPWA9K2B0xQeS5Rju	744	1
1E93CpP60RHFNiT5Qfvn	838	1
3ekVow2ajZHbTnBcsDfX	427	1
3X2nY7iQaPBIWDrAZqJe	227	0
46OZzdsSKDCFV8h7XWxf	402	1

```

0 4sCh00eqidfLtS83JMVE 0.233131 3
1 89VdQqEkGB2Fzv4lXnbU 0.121401 3
2 3hfRptP5gUEqDH1CkMYX 0.360250 4
3 6GgdCXPL2Bali0f8ZfkA 0.122087 3
4 E4SvkGO5CuB8LehadFo6 0.122047 3

```

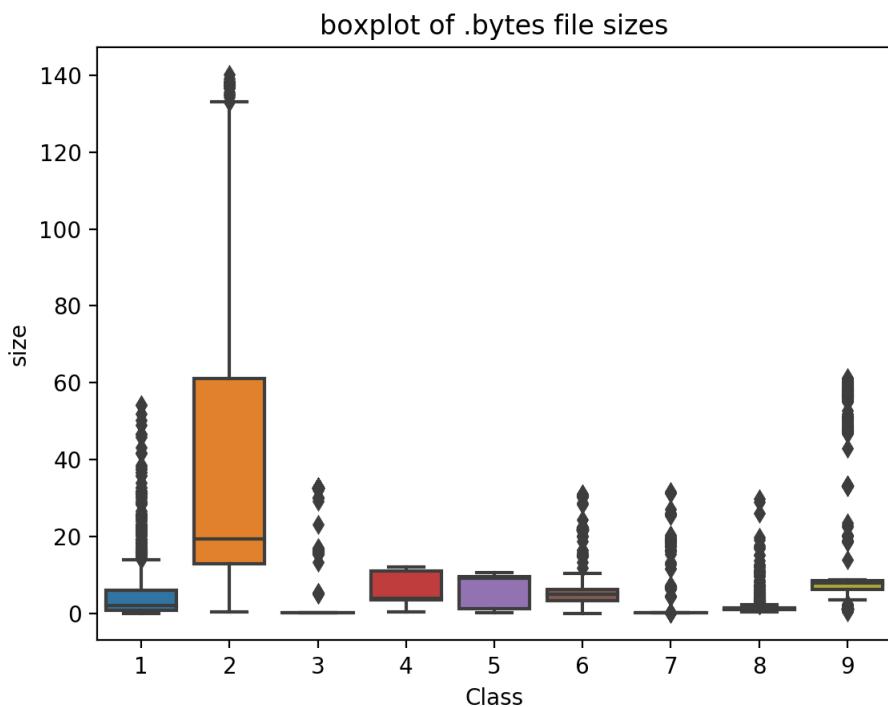
#### 4.2.1.2 Distribution of .asm file sizes

In [34]:

```

plt.close()
#boxplot of asm files
ax = sns.boxplot(x="Class", y="size", data=asm_size_byte)
plt.title("boxplot of .bytes file sizes")
plt.show()

```



In [18]:

```

# add the file size feature to previous extracted features
print(result_asm.shape)
print(asm_size_byte.shape)
result_asm = pd.merge(result_asm, asm_size_byte.drop(['Class'], axis=1), on='ID', how='left')
result_asm.head()

```

```
(10868, 53)
(10868, 3)
```

Out[18]:

	ID	HEADER:	.text:	.Pav:	.idata:	.data:	.bss:	.rdata:	.edata:	.rsrc:	...	esi	eax	ebx	ecx	edi	ebp	es	
0	01kcPWA9K2BOxQeS5Rju		19	744	0	127	57	0	323	0	3	...	66	15	43	83	0	17	4
1	1E93CpP60RHFNiT5Qfvn		17	838	0	103	49	0	0	0	3	...	29	48	82	12	0	14	
2	3ekVow2ajZHbTnBcsDfX		17	427	0	50	43	0	145	0	3	...	42	10	67	14	0	11	
3	3X2nY7iQaPB1WDrAZqJe		17	227	0	43	19	0	0	0	3	...	8	14	7	2	0	8	
4	46OZzdsSKDCFV8h7XWxf		17	402	0	59	170	0	0	0	3	...	9	18	29	5	0	11	

5 rows × 54 columns

In [19]:

```
# we normalize the data each column
result_asm = normalize(result_asm)
result_asm.head()
```

Out[19]:

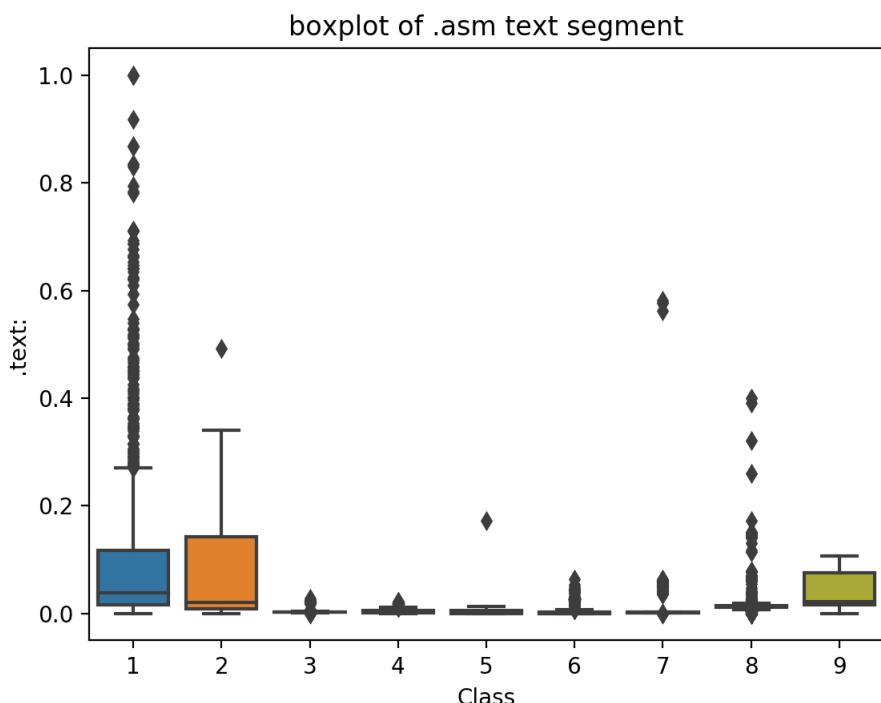
	ID	HEADER:	.text:	.Pav:	.idata:	.data:	.bss:	.rdata:	.edata:	.rsrc:	...	esi	eax
0	01kcPWA9K2BOxQeS5Rju	0.107345	0.001092	0.0	0.000761	0.000023	0.0	0.000084	0.0	0.000072	...	0.000746	0.000301
1	1E93CpP60RHFniT5Qfvn	0.096045	0.001230	0.0	0.000617	0.000019	0.0	0.000000	0.0	0.000072	...	0.000328	0.000965
2	3ekVow2ajZHbTnBcsDfX	0.096045	0.000627	0.0	0.000300	0.000017	0.0	0.000038	0.0	0.000072	...	0.000475	0.000201
3	3X2nY7iQaPBIDrAZqJe	0.096045	0.000333	0.0	0.000258	0.000008	0.0	0.000000	0.0	0.000072	...	0.000090	0.000281
4	46OZzdsSKDCFV8h7XWxf	0.096045	0.000590	0.0	0.000353	0.000068	0.0	0.000000	0.0	0.000072	...	0.000102	0.000362

5 rows × 54 columns

#### 4.2.2 Univariate analysis on asm file features

In [55]:

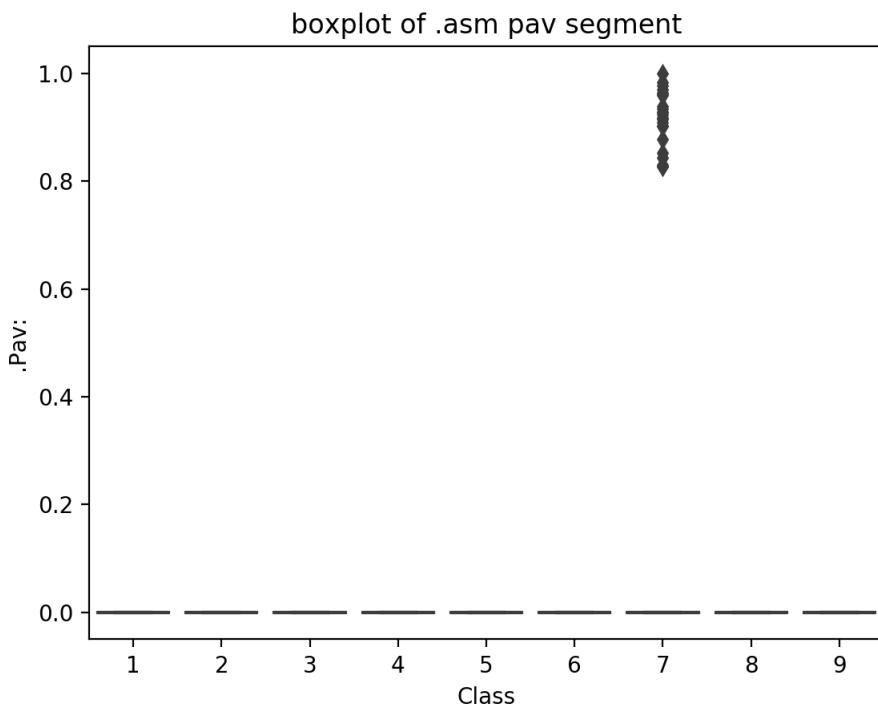
```
plt.close()
ax = sns.boxplot(x="Class", y=".text:", data=result_asm)
plt.title("boxplot of .asm text segment")
plt.show()
```



The plot is between Text and class  
Class 1,2 and 9 can be easily separated

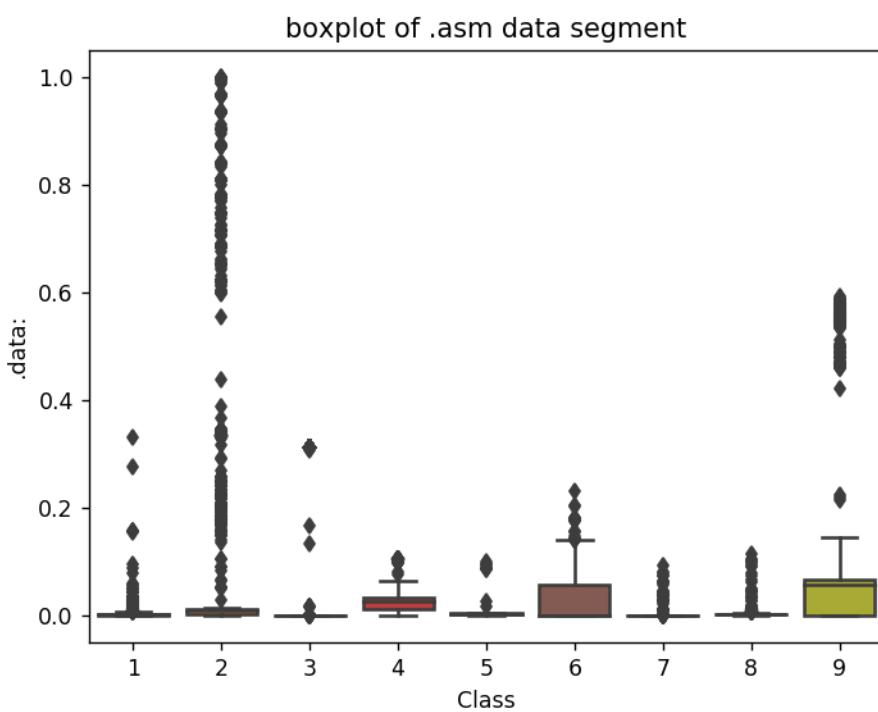
In [56]:

```
plt.close()
ax = sns.boxplot(x="Class", y=".Pav:", data=result_asm)
plt.title("boxplot of .asm pav segment")
plt.show()
```



In [0]:

```
plt.close()
ax = sns.boxplot(x="Class", y=".data:", data=result_asm)
plt.title("boxplot of .asm data segment")
plt.show()
```



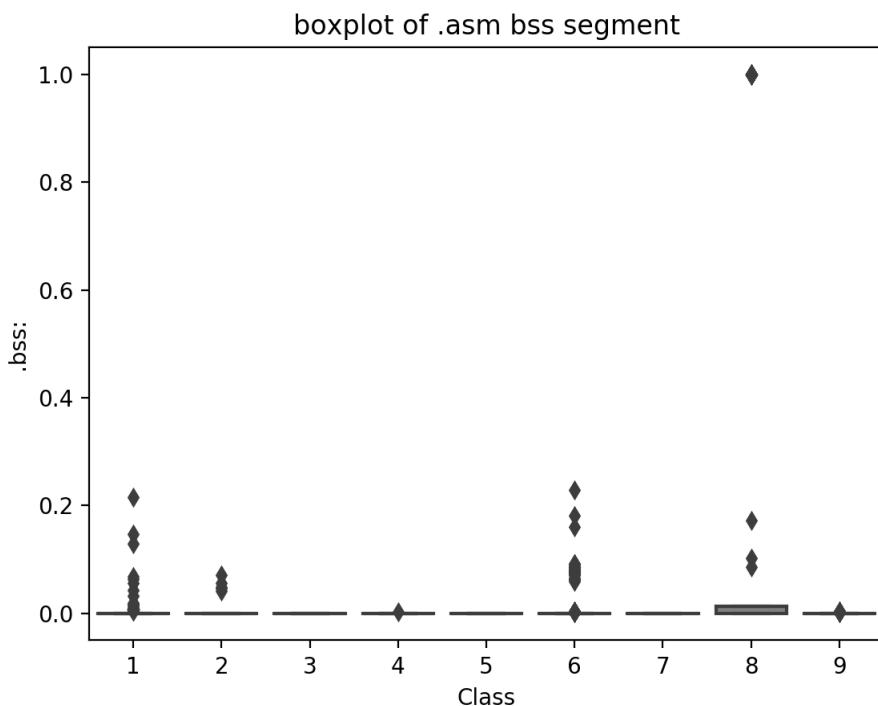
The plot is between data segment and class label  
class 6 and class 9 can be easily separated from given points

In [57]:

```

plt.close()
ax = sns.boxplot(x="Class", y=".bss:", data=result_asm)
plt.title("boxplot of .asm bss segment")
plt.show()

```



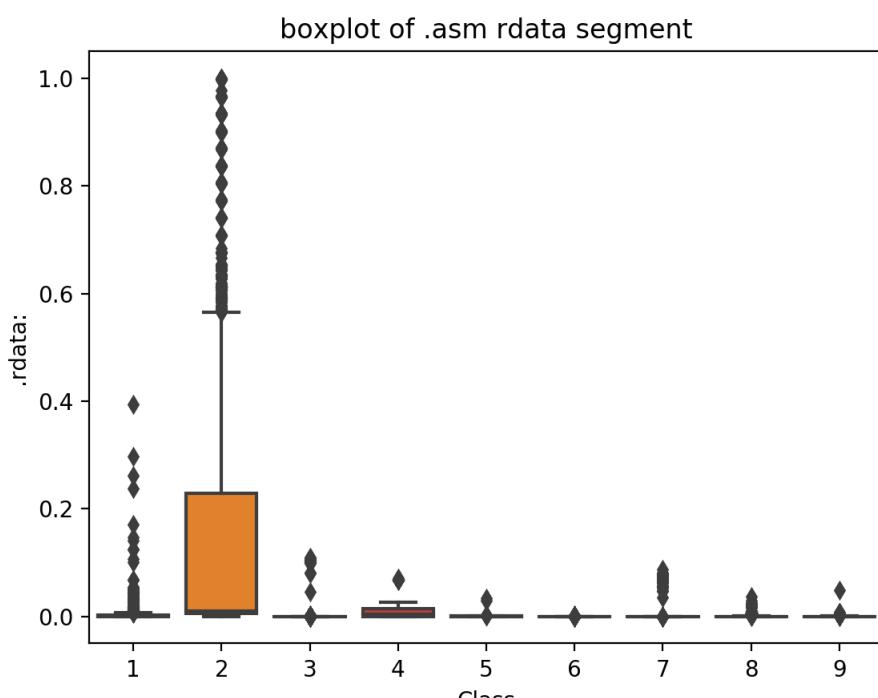
plot between bss segment and class label  
very less number of files are having bss segment

In [58]:

```

plt.close()
ax = sns.boxplot(x="Class", y=".rdata:", data=result_asm)
plt.title("boxplot of .asm rdata segment")
plt.show()

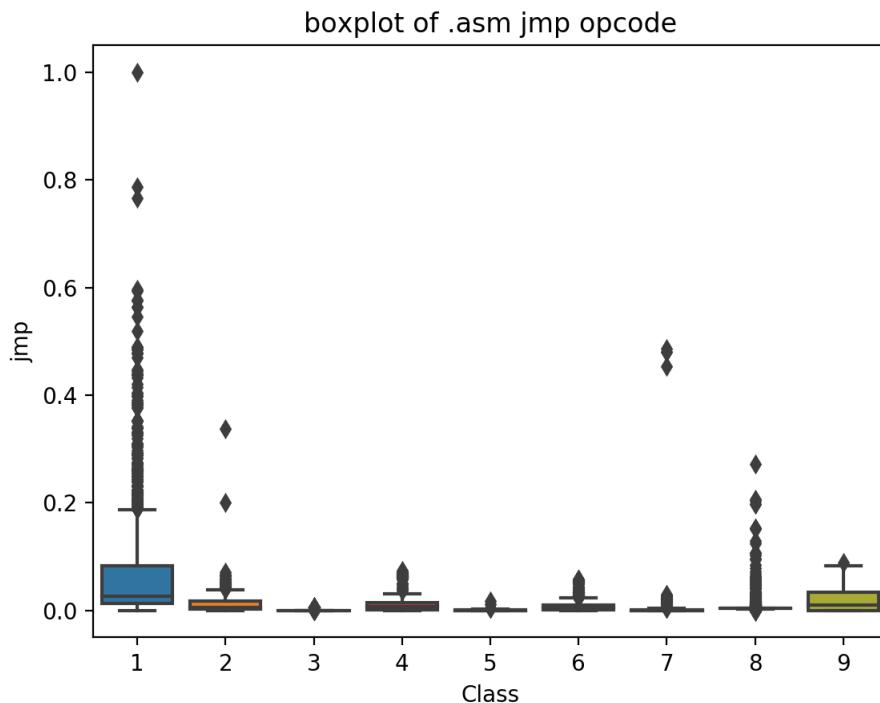
```



```
Plot between rdata segment and Class segment
Class 2 can be easily separated 75 percentile files are having 1M rdata lines
```

In [59]:

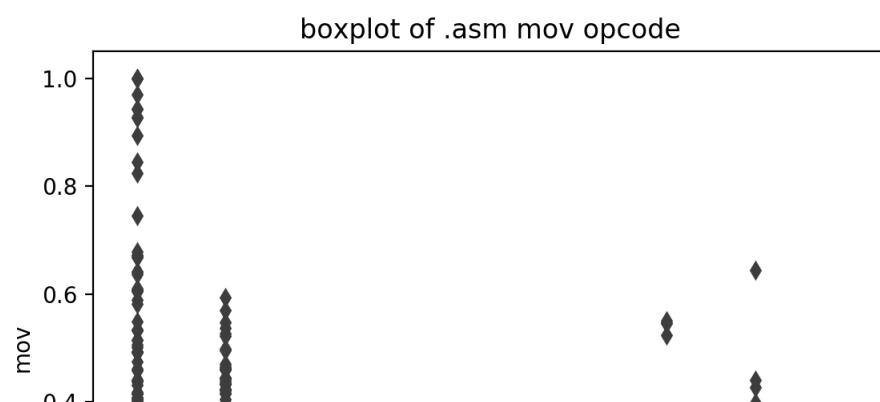
```
plt.close()
ax = sns.boxplot(x="Class", y="jmp", data=result_asm)
plt.title("boxplot of .asm jmp opcode")
plt.show()
```

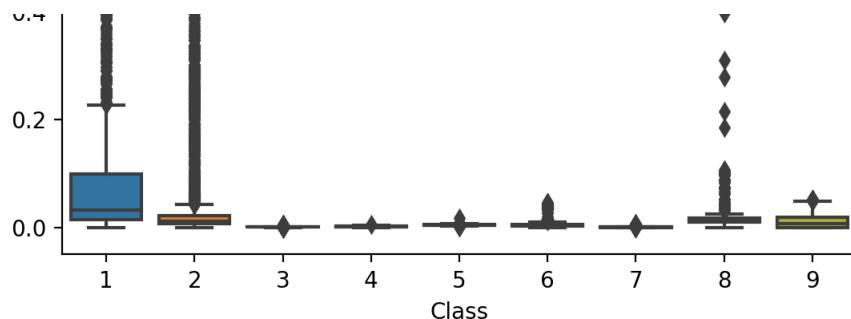


```
plot between jmp and Class label
Class 1 is having frequency of 2000 approx in 75 percentile of files
```

In [60]:

```
plt.close()
ax = sns.boxplot(x="Class", y="mov", data=result_asm)
plt.title("boxplot of .asm mov opcode")
plt.show()
```

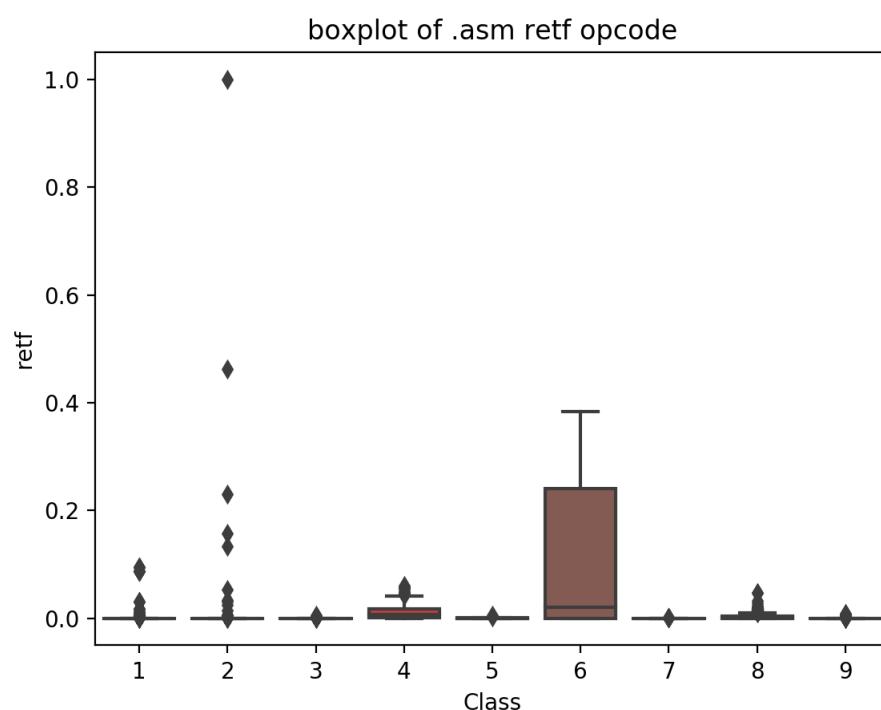




```
plot between Class label and mov opcode
Class 1 is having frequency of 2000 approx in 75 percentile of files
```

In [61]:

```
plt.close()
ax = sns.boxplot(x="Class", y="retf", data=result_asm)
plt.title("boxplot of .asm retf opcode")
plt.show()
```

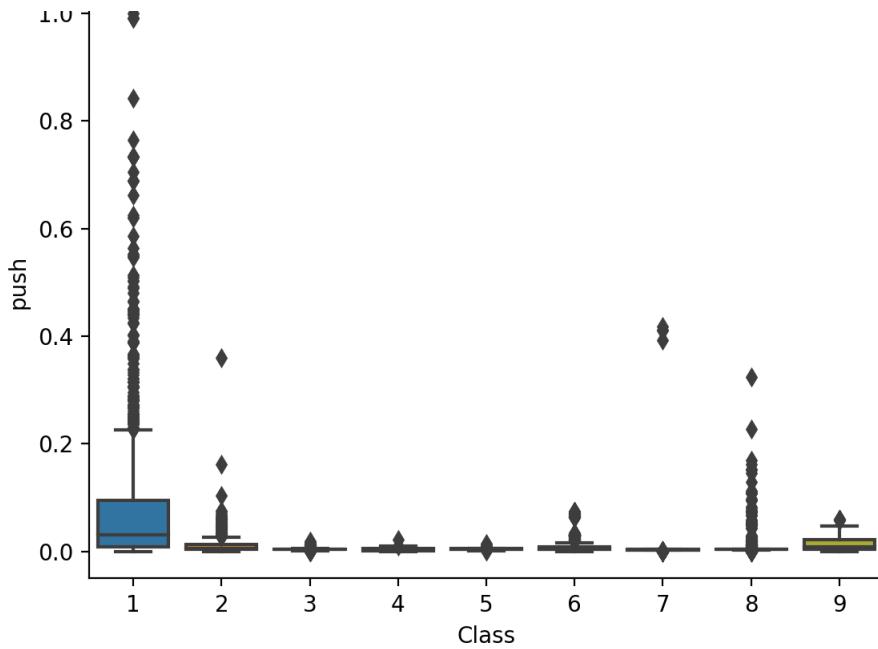


```
plot between Class label and retf
Class 6 can be easily separated with opcode retf
The frequency of retf is approx of 250.
```

In [62]:

```
plt.close()
ax = sns.boxplot(x="Class", y="push", data=result_asm)
plt.title("boxplot of .asm push opcode")
plt.show()
```

boxplot of .asm push opcode



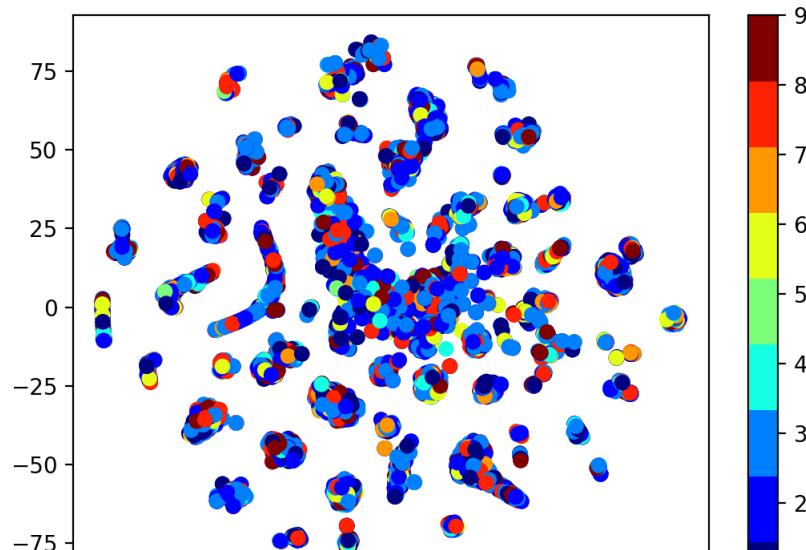
```
plot between push opcode and Class label
Class 1 is having 75 precentile files with push opcodes of frequency 1000
```

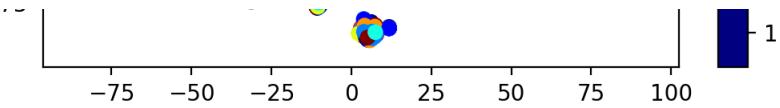
#### 4.2.2 Multivariate Analysis on .asm file features

In [47]:

```
plt.close()
# check out the course content for more explantion on tsne algorithm
# https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/t-distributed-stochastic-
-neighbourhood-embeddingt-sne-part-1/

#multivariate analysis on byte files
#this is with perplexity 50
xtsne=TSNE(perplexity=50)
results=xtsne.fit_transform(result_asm.drop(['ID','Class'], axis=1).fillna(0))
vis_x = results[:, 0]
vis_y = results[:, 1]
plt.scatter(vis_x, vis_y, c=data_y, cmap=plt.cm.get_cmap("jet", 9))
plt.colorbar(ticks=range(10))
plt.clim(0.5, 9)
plt.show()
```

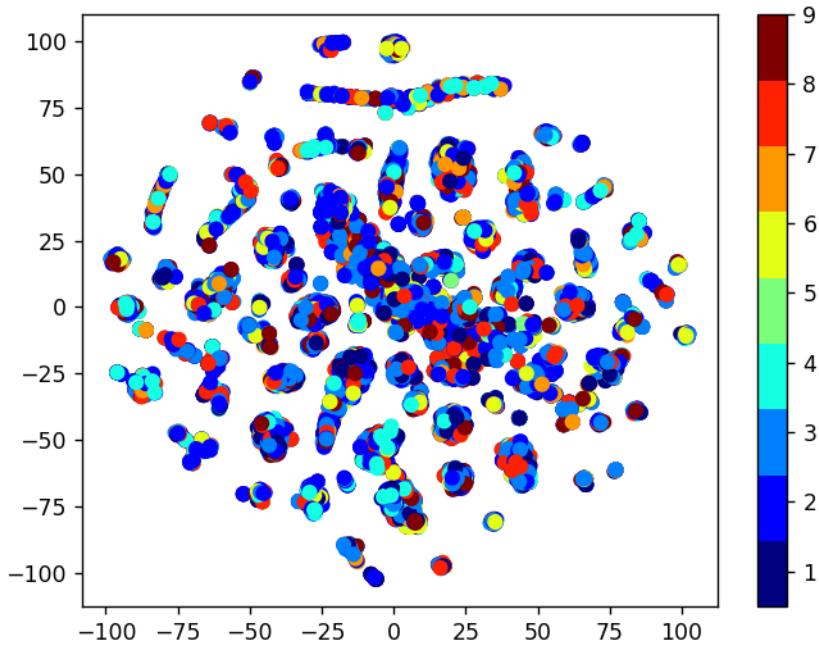




In [0]:

```
plt.close()
# by univariate analysis on the .asm file features we are getting very negligible information from
# 'rtn', '.BSS:' '.CODE' features, so here we are trying multivariate analysis after removing those
# features
# the plot looks very messy

xtsne=TSNE(perplexity=30)
results=xtsne.fit_transform(result_asm.drop(['ID','Class', 'rtn', '.BSS:', '.CODE','size'], axis=1))
vis_x = results[:, 0]
vis_y = results[:, 1]
plt.scatter(vis_x, vis_y, c=data_y, cmap=plt.cm.get_cmap("jet", 9))
plt.colorbar(ticks=range(10))
plt.clim(0.5, 9)
plt.show()
```



TSNE for asm data with perplexity 50

#### 4.2.3 Conclusion on EDA

- We have taken only 52 features from asm files (after reading through many blogs and research papers)
- The univariate analysis was done only on few important features.
- Take-aways
  - 1. Class 3 can be easily separated because of the frequency of segments, opcodes and keywords being less
  - 2. Each feature has its unique importance in separating the Class labels.

#### 4.3 Train and test split

```
In [20]:
```

```
asm_y = result_asm['Class']
asm_x = result_asm.drop(['ID', 'Class', '.BSS:', 'rtn', '.CODE'], axis=1)
```

```
In [21]:
```

```
X_train_asm, X_test_asm, y_train_asm, y_test_asm = train_test_split(asm_x,asm_y ,stratify=asm_y,test_size=0.20)
X_train_asm, X_cv_asm, y_train_asm, y_cv_asm = train_test_split(X_train_asm, y_train_asm,stratify=y_train_asm,test_size=0.20)
```

```
In [22]:
```

```
print( X_cv_asm.isnull().all())
```

```
HEADER:      False
.text:       False
.Pav:        False
.idata:      False
.data:       False
.bss:        False
.rdata:      False
.edata:      False
.rsrc:       False
.tls:        False
.reloc:      False
jmp:         False
mov:         False
retf:        False
push:        False
pop:         False
xor:         False
retn:        False
nop:         False
sub:         False
inc:         False
dec:         False
add:         False
imul:        False
xchg:        False
or:          False
shr:         False
cmp:         False
call:        False
shl:         False
ror:         False
rol:         False
jnb:         False
jz:          False
lea:          False
movzx:       False
.dll:        False
std:::       False
:dword:      False
edx:         False
esi:         False
eax:         False
ebx:         False
ecx:         False
edi:         False
ebp:         False
esp:         False
eip:         False
size:        False
dtype: bool
```

## 4.4. Machine Learning models on features of .asm files

### 4.4.1 K-Nearest Neighbors

In [66]:

```
%%time
plt.close()
# find more about KNeighborsClassifier() here http://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html
# -----
# default parameter
# KNeighborsClassifier(n_neighbors=5, weights='uniform', algorithm='auto', leaf_size=30, p=2,
# metric='minkowski', metric_params=None, n_jobs=1, **kwargs)

# methods of
# fit(X, y) : Fit the model using X as training data and y as target values
# predict(X) : Predict the class labels for the provided data
# predict_proba(X) : Return probability estimates for the test data X.
# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-nearest-neighbors-geometric-intuition-with-a-toy-example-1/
#-----

# find more about CalibratedClassifierCV here at http://scikit-learn.org/stable/modules/generated/sklearn.calibration.CalibratedClassifierCV.html
# -----
# default parameters
# sklearn.calibration.CalibratedClassifierCV(base_estimator=None, method='sigmoid', cv=3)
#
# some of the methods of CalibratedClassifierCV()
# fit(X, y[, sample_weight]) Fit the calibrated model
# get_params([deep]) Get parameters for this estimator.
# predict(X) Predict the target of new samples.
# predict_proba(X) Posterior probabilities of classification
# -----
# video link:
#-----

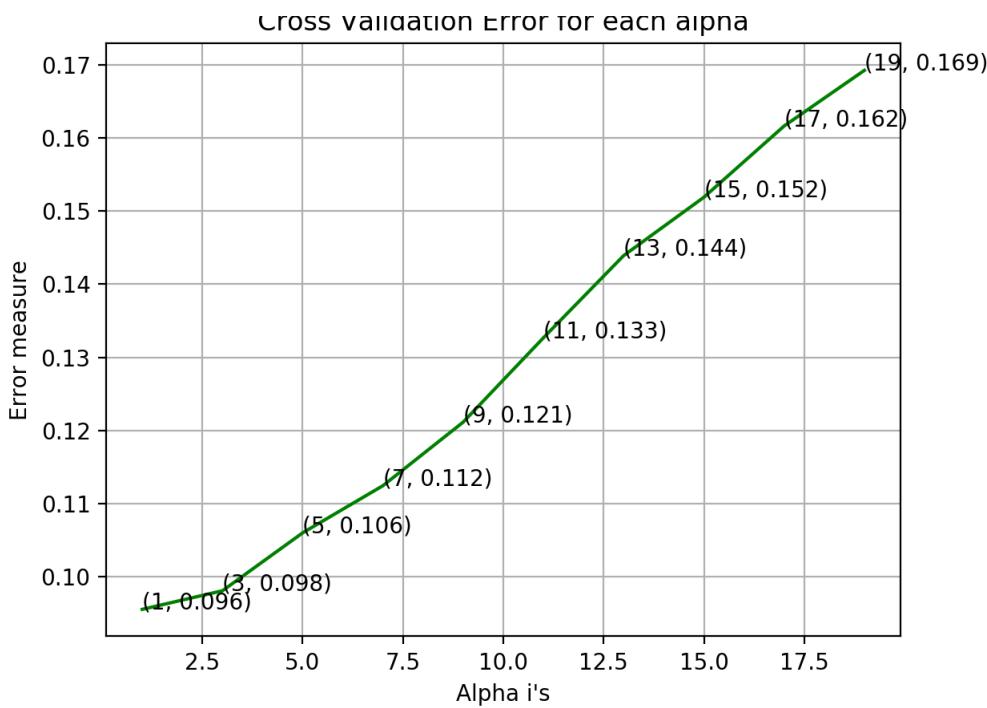

alpha = [x for x in range(1, 21,2)]
cv_log_error_array=[]
for i in alpha:
    k_cfl=KNeighborsClassifier(n_neighbors=i)
    k_cfl.fit(X_train_asm,y_train_asm)
    sig_clf = CalibratedClassifierCV(k_cfl, method="sigmoid")
    sig_clf.fit(X_train_asm, y_train_asm)
    predict_y = sig_clf.predict_proba(X_cv_asm)
    cv_log_error_array.append(log_loss(y_cv_asm, predict_y, labels=k_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for k = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for k = 1 is 0.09551157959211777
log_loss for k = 3 is 0.09805566398432622
log_loss for k = 5 is 0.1059879771204611
log_loss for k = 7 is 0.11243681772622365
log_loss for k = 9 is 0.12113271316966784
log_loss for k = 11 is 0.1326757552240621
log_loss for k = 13 is 0.1439229805056757
log_loss for k = 15 is 0.15190968928970974
log_loss for k = 17 is 0.16166115338106687
log_loss for k = 19 is 0.16927369275398452
```



```
CPU times: user 19.3 s, sys: 8.33 ms, total: 19.3 s
Wall time: 19.3 s
```

In [67]:

```
%%time
plt.close()
k_cfl=KNeighborsClassifier(n_neighbors=alpha[best_alpha])
k_cfl.fit(X_train_asm,y_train_asm)
sig_clf = CalibratedClassifierCV(k_cfl, method="sigmoid")
sig_clf.fit(X_train_asm, y_train_asm)
pred_y=sig_clf.predict(X_test_asm)

predict_y = sig_clf.predict_proba(X_train_asm)
print ('log loss for train data',log_loss(y_train_asm, predict_y))
predict_y = sig_clf.predict_proba(X_cv_asm)
print ('log loss for cv data',log_loss(y_cv_asm, predict_y))
predict_y = sig_clf.predict_proba(X_test_asm)
print ('log loss for test data',log_loss(y_test_asm, predict_y))
plot_confusion_matrix(y_test_asm,sig_clf.predict(X_test_asm))
```

```
log loss for train data 0.037475912101338785
```

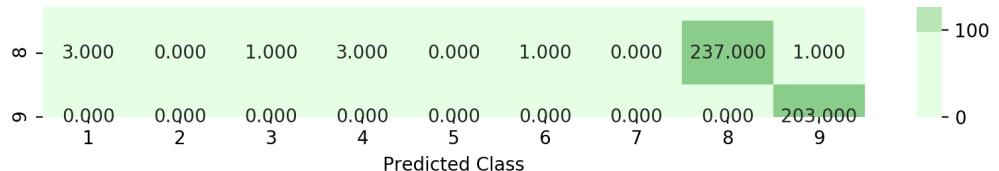
```
log loss for cv data 0.09551157959211777
```

```
log loss for test data 0.07985178727072437
```

```
Number of misclassified points 1.1039558417663293
```

----- Confusion matrix -----

Original Class	1	2	3	4	5	6	7	8	9
1	298.000	0.000	0.000	0.000	0.000	0.000	1.000	7.000	2.000
2	0.000	496.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	587.000	1.000	0.000	0.000	0.000	0.000	0.000
4	0.000	0.000	0.000	94.000	0.000	1.000	0.000	0.000	0.000
5	0.000	0.000	0.000	1.000	6.000	0.000	0.000	0.000	1.000
6	0.000	0.000	0.000	1.000	0.000	149.000	0.000	0.000	0.000
7	0.000	0.000	0.000	0.000	0.000	0.000	80.000	0.000	0.000

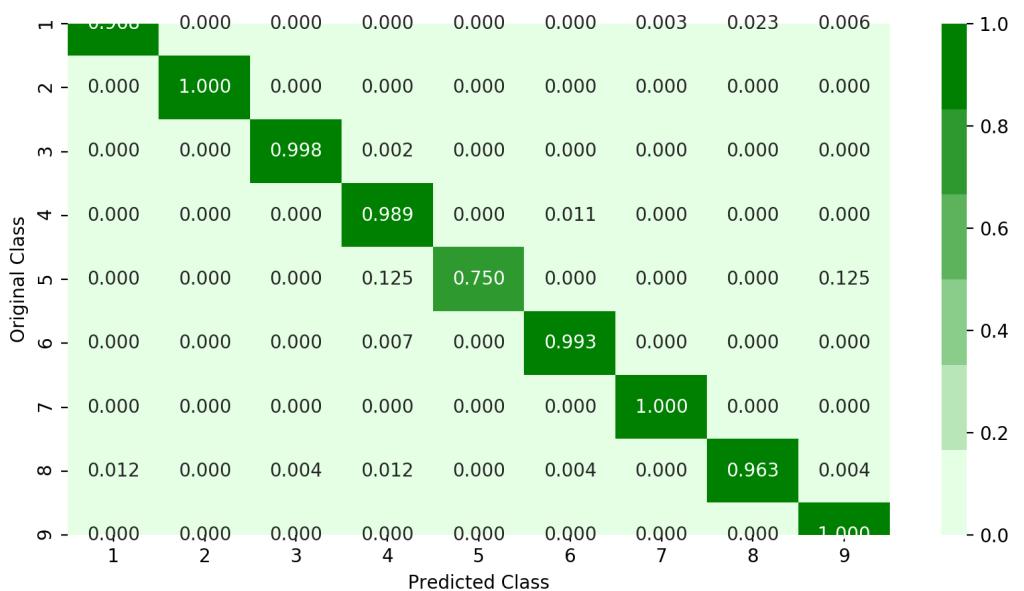


Precision matrix



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

Recall matrix



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

CPU times: user 5.24 s, sys: 28.6 ms, total: 5.27 s

Wall time: 5.24 s

#### 4.4.2 Logistic Regression

In [68]:

```
%%time
plt.close()
# read more about SGDClassifier() at http://scikit-
learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html
# -----
# default parameters
# SGDClassifier(loss='hinge', penalty='l2', alpha=0.0001, l1_ratio=0.15, fit_intercept=True, max_i
ter=None, tol=None,
# shuffle=True, verbose=0, epsilon=0.1, n_jobs=1, random_state=None, learning_rate='optimal', eta0
=0.0, power_t=0.5,
# class_weight=None, warm_start=False, average=False, n_iter=None)

# some of methods
# fit(X, y[, coef_init, intercept_init, ...]) Fit linear model with Stochastic Gradient Descent.
# predict(X) Predict class labels for samples in X.

#-----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-in
tuition-1/
#-----

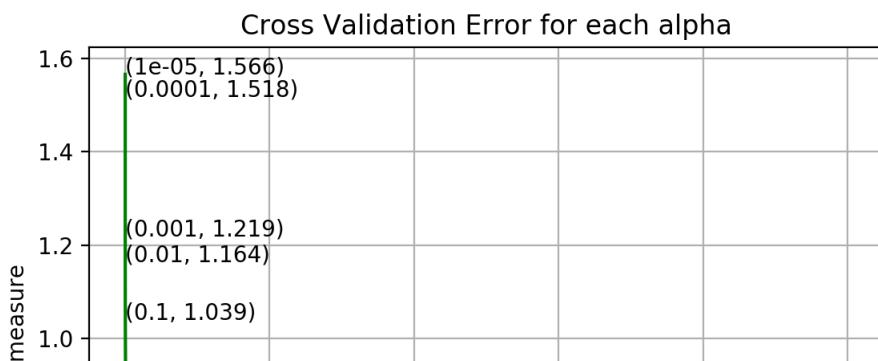

alpha = [10 ** x for x in range(-5, 4)]
cv_log_error_array=[]
for i in alpha:
    logisticR=LogisticRegression(penalty='l2',C=i,class_weight='balanced')
    logisticR.fit(X_train_asm,y_train_asm)
    sig_clf = CalibratedClassifierCV(logisticR, method="sigmoid")
    sig_clf.fit(X_train_asm, y_train_asm)
    predict_y = sig_clf.predict_proba(X_cv_asm)
    cv_log_error_array.append(log_loss(y_cv_asm, predict_y, labels=logisticR.classes_, eps=1e-15))

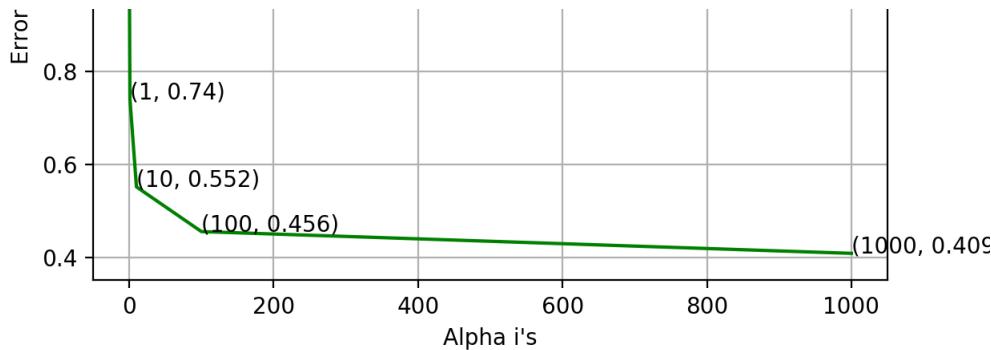
for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c =  1e-05 is 1.5661498686463826
log_loss for c =  0.0001 is 1.517675047020921
log_loss for c =  0.001 is 1.2191622564580826
log_loss for c =  0.01 is 1.1642755471095085
log_loss for c =  0.1 is 1.0393439618593712
log_loss for c =  1 is 0.7399622931531011
log_loss for c =  10 is 0.5521355759037754
log_loss for c =  100 is 0.4561736339097347
log_loss for c =  1000 is 0.40949221859900775
```





CPU times: user 17.2 s, sys: 4.15 ms, total: 17.2 s  
Wall time: 17.2 s

In [69]:

```
%time
plt.close()
logisticR=LogisticRegression(penalty='l2',C=alpha[best_alpha],class_weight='balanced')
logisticR.fit(X_train_asm,y_train_asm)
sig_clf = CalibratedClassifierCV(logisticR, method="sigmoid")
sig_clf.fit(X_train_asm, y_train_asm)

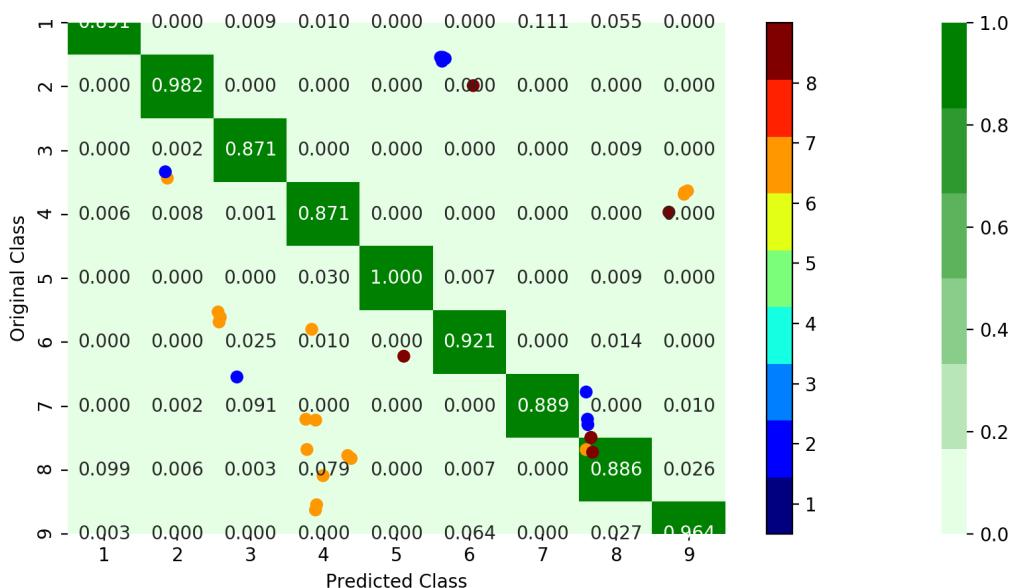
predict_y = sig_clf.predict_proba(X_train_asm)
print ('log loss for train data',(log_loss(y_train_asm, predict_y, labels=logisticR.classes_, eps=1e-15)))
predict_y = sig_clf.predict_proba(X_cv_asm)
print ('log loss for cv data',(log_loss(y_cv_asm, predict_y, labels=logisticR.classes_, eps=1e-15)))
predict_y = sig_clf.predict_proba(X_test_asm)
print ('log loss for test data',(log_loss(y_test_asm, predict_y, labels=logisticR.classes_, eps=1e-15)))
plot_confusion_matrix(y_test_asm,sig_clf.predict(X_test_asm))
```

log loss for train data 0.3951231421379966  
log loss for cv data 0.40949221859900775  
log loss for test data 0.39929985766626985  
Number of misclassified points 8.693652253909843

----- Confusion matrix -----



----- Precision matrix -----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

CPU times: user 7.11 s, sys: 16.8 ms, total: 7.13 s

Wall time: 7.1 s

#### 4.4.3 Random Forest Classifier

In [70]:

```
%%time
plt.close()
# -----
# default parameters
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,
# # min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
# # min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
```

```

verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forest-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train_asm,y_train_asm)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train_asm, y_train_asm)
    predict_y = sig_clf.predict_proba(X_cv_asm)
    cv_log_error_array.append(log_loss(y_cv_asm, predict_y, labels=r_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

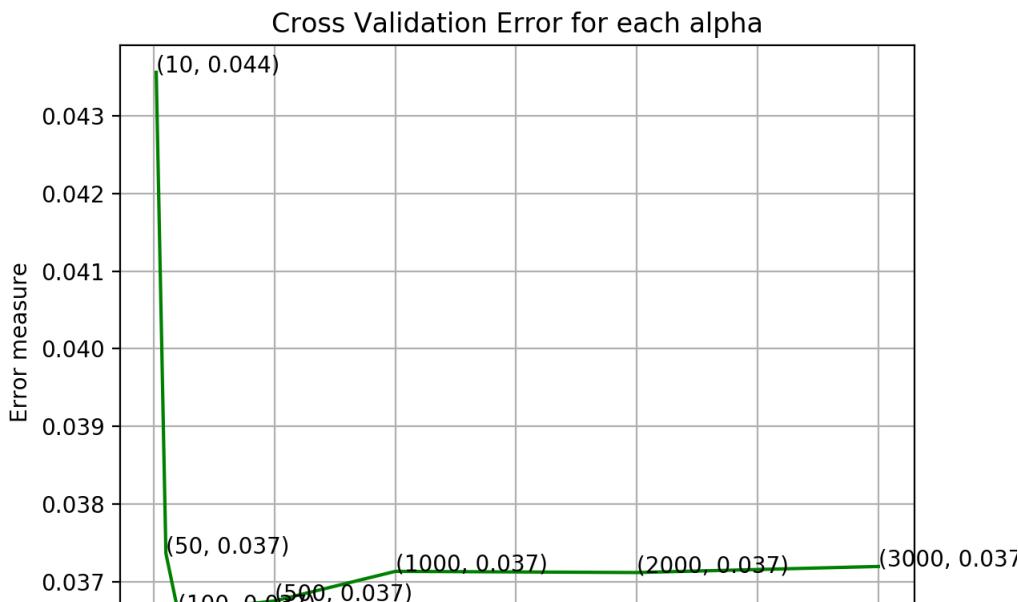
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

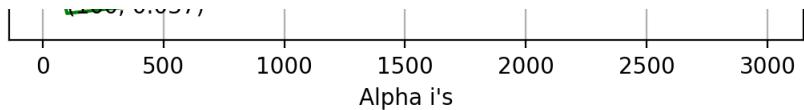
```

```

log_loss for c =  10 is  0.04356250797901248
log_loss for c =  50 is  0.037363030786582735
log_loss for c =  100 is  0.036614779171787376
log_loss for c =  500 is  0.0367484713872376
log_loss for c =  1000 is  0.03713029805086209
log_loss for c =  2000 is  0.03711513955536283
log_loss for c =  3000 is  0.03719323042006945

```





CPU times: user 7min 10s, sys: 29.6 s, total: 7min 40s  
 Wall time: 2min

In [71]:

```
%time
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train_asm,y_train_asm)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train_asm, y_train_asm)
predict_y = sig_clf.predict_proba(X_train_asm)
print ('log loss for train data',(log_loss(y_train_asm, predict_y, labels=sig_clf.classes_, eps=1e-15)))
predict_y = sig_clf.predict_proba(X_cv_asm)
print ('log loss for cv data',(log_loss(y_cv_asm, predict_y, labels=sig_clf.classes_, eps=1e-15)))
predict_y = sig_clf.predict_proba(X_test_asm)
print ('log loss for test data',(log_loss(y_test_asm, predict_y, labels=sig_clf.classes_, eps=1e-15)))
plot_confusion_matrix(y_test_asm,sig_clf.predict(X_test_asm))
```

log loss for train data 0.021489116411912302

log loss for cv data 0.036614779171787376

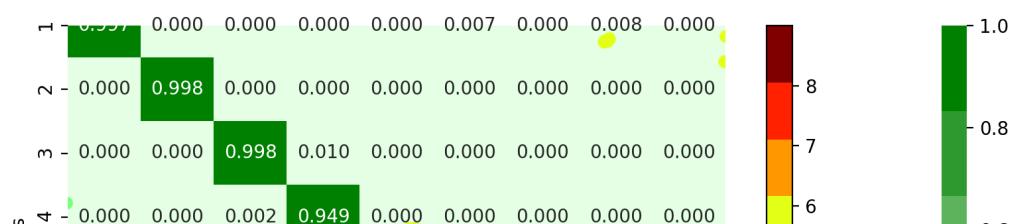
log loss for test data 0.03239270498531275

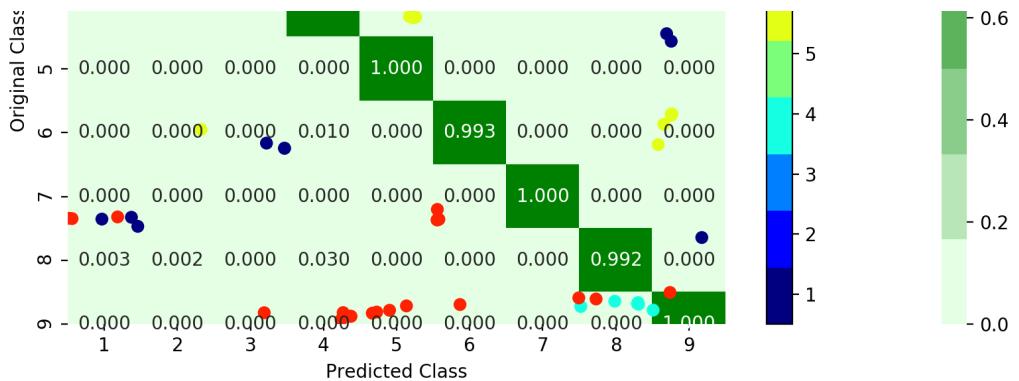
Number of misclassified points 0.5059797608095675

----- Confusion matrix -----



----- Precision matrix -----

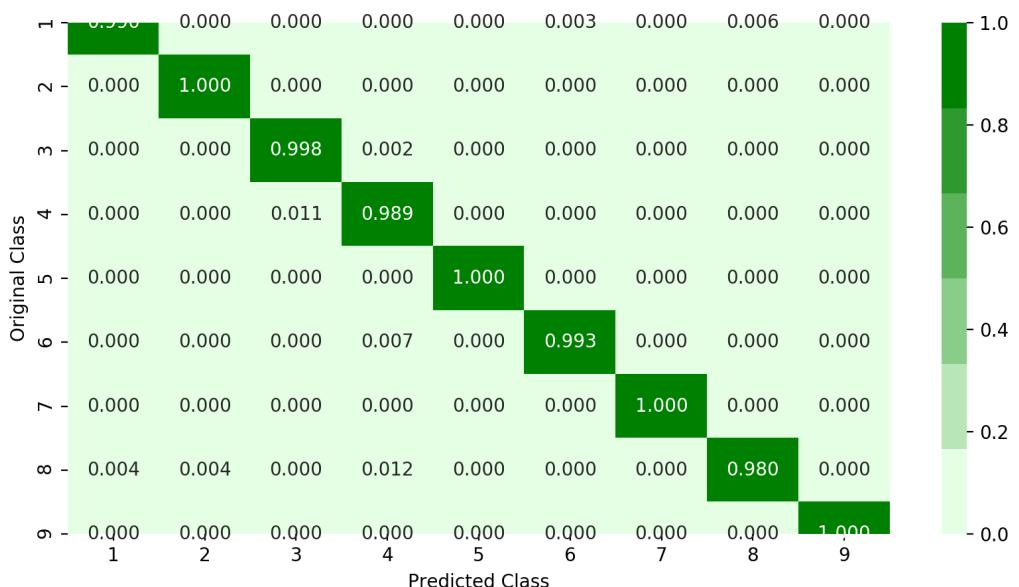




Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----

[]



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

CPU times: user 8.2 s, sys: 645 ms, total: 8.84 s

Wall time: 4.07 s

#### 4.4.4 XgBoost Classifier

In [72]:

```
%%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python\_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is no
```

```

t thread safe.

# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClassifier(n_estimators=i,nthread=-1)
    x_cfl.fit(X_train_asm,y_train_asm)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train_asm, y_train_asm)
    predict_y = sig_clf.predict_proba(X_cv_asm)
    cv_log_error_array.append(log_loss(y_cv_asm, predict_y, labels=x_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

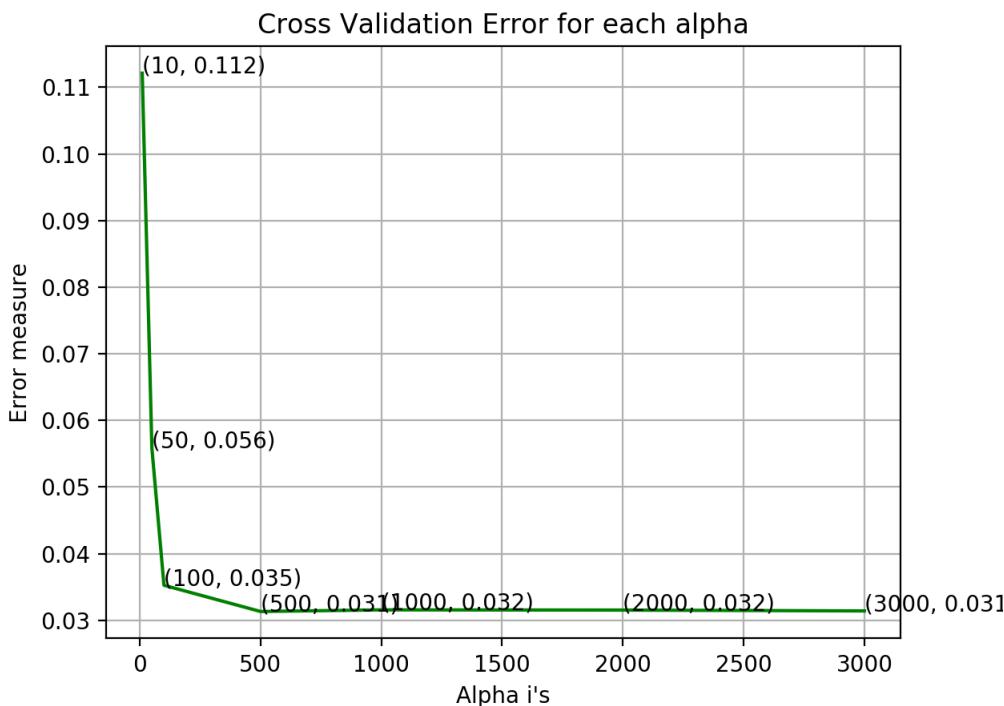
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

```

```

log_loss for c =  10 is  0.11212661518377143
log_loss for c =  50 is  0.05582972313723833
log_loss for c =  100 is  0.03525743616432047
log_loss for c =  500 is  0.031322696772212756
log_loss for c =  1000 is  0.031566884648293325
log_loss for c =  2000 is  0.03153641864350838
log_loss for c =  3000 is  0.03140732250886729

```



```

CPU times: user 17min 32s, sys: 0 ns, total: 17min 32s
Wall time: 17min 32s

```

In [73]:

```
%time
# plt.close()
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train_asm,y_train_asm)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_asm, y_train_asm)

predict_y = sig_clf.predict_proba(X_train_asm)

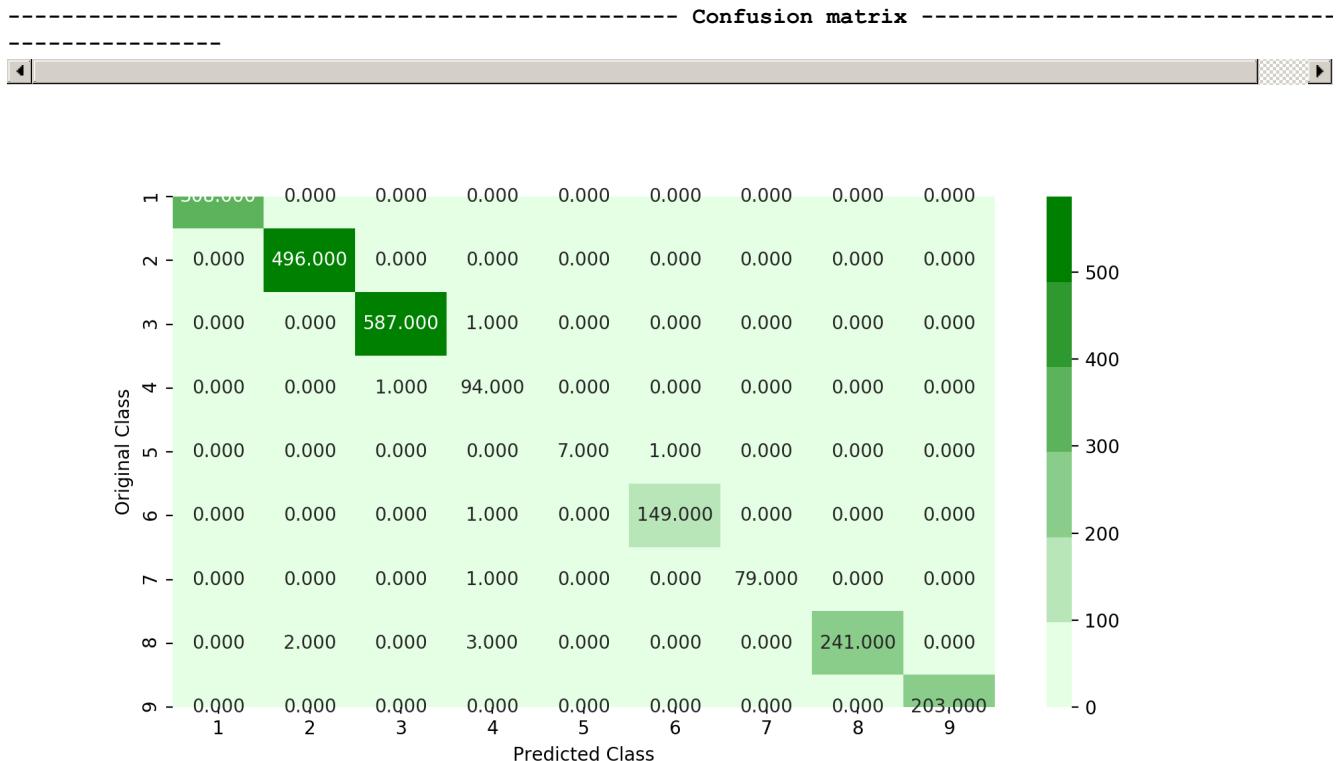
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_asm, predict_y))
predict_y = sig_clf.predict_proba(X_cv_asm)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_asm, predict_y))
predict_y = sig_clf.predict_proba(X_test_asm)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_asm, predict_y))
```

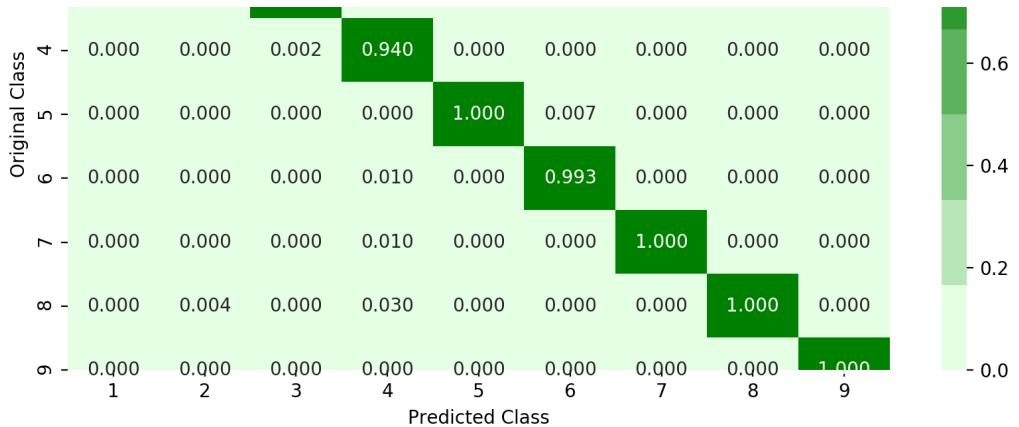
```
For values of best alpha = 500 The train log loss is: 0.02020393568415983
For values of best alpha = 500 The cross validation log loss is: 0.031322696772212756
For values of best alpha = 500 The test log loss is: 0.030377059563789385
CPU times: user 1min 48s, sys: 0 ns, total: 1min 48s
Wall time: 1min 48s
```

In [74]:

```
plt.close()
plot_confusion_matrix(y_test_asm,sig_clf.predict(X_test_asm))
```

Number of misclassified points 0.45998160073597055





Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----

[| | | | | | | | | |]



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

#### 4.4.5 Xgboost Classifier with best hyperparameters

In [75]:

```
%time
x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
    'subsample':[0.1,0.3,0.5,1]
}
random_cfl=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1,)
random_cfl.fit(X_train_asm,y_train_asm)
```

Fitting 3 folds for each of 10 candidates, totalling 30 fits

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:   12.4s
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:  1.1min
[Parallel(n_jobs=-1)]: Done  19 out of 30 | elapsed:  1.6min remaining:  56.6s
```

```
[Parallel(n_jobs=-1)]: Done 23 out of 30 | elapsed: 1.9min remaining: 34.1s
[Parallel(n_jobs=-1)]: Done 27 out of 30 | elapsed: 2.0min remaining: 13.4s
[Parallel(n_jobs=-1)]: Done 30 out of 30 | elapsed: 2.2min finished
```

CPU times: user 14.5 s, sys: 323 ms, total: 14.8 s  
Wall time: 2min 24s

Out[75]:

```
RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                    estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                            colsample_bylevel=1,
                                            colsample_bynode=1,
                                            colsample_bytree=1, gamma=0,
                                            learning_rate=0.1, max_delta_step=0,
                                            max_depth=3, min_child_weight=1,
                                            missing=None, n_estimators=100,
                                            n_jobs=1, nthread=None,
                                            objective='binary:logistic',
                                            random_state=0, reg_alpha=0.3,
                                            seed=None, silent=None, subsample=1,
                                            verbosity=1),
                    iid='warn', n_iter=10, n_jobs=-1,
                    param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                    pre_dispatch='2*n_jobs', random_state=None, refit=True,
                    return_train_score=False, scoring=None, verbose=10)
```

In [76]:

```
print (random_cfl.best_params_)
```

```
{'subsample': 1, 'n_estimators': 500, 'max_depth': 3, 'learning_rate': 0.15, 'colsample_bytree': 0.3}
```

In [77]:

```
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----

x_cfl=XGBClassifier(n_estimators=500,subsample=1,learning_rate=0.15,colsample_bytree=0.3,max_depth=3)
x_cfl.fit(X_train_asm,y_train_asm)
c_cfl=CalibratedClassifierCV(x_cfl,method='sigmoid')
c_cfl.fit(X_train_asm,y_train_asm)
```

```

predict_y = c_cfl.predict_proba(X_train_asm)
print ('train loss',log_loss(y_train_asm, predict_y))
predict_y = c_cfl.predict_proba(X_cv_asm)
print ('cv loss',log_loss(y_cv_asm, predict_y))
predict_y = c_cfl.predict_proba(X_test_asm)
print ('test loss',log_loss(y_test_asm, predict_y))

train loss 0.019288923540468395
cv loss 0.03159265585460439
test loss 0.02576461657919649

```

## 4.5. Machine Learning models on features of both .asm and .bytes files

### 4.5.1. Merging both asm and byte file features

In [23]:

```
result.head()
```

Out[23]:

	ID	0	1	2	3	4	5	6	7	8	...	fa
0	FGCV8wbAuQp1HiSd7kvm	0.002525	0.000082	0.000013	0.000017	0.000016	0.000012	0.000004	0.000003	0.000099	...	0.000012
1	EmVYF0ZrlBWCGL5168XH	0.010740	0.001771	0.000416	0.000489	0.000886	0.000320	0.000332	0.000662	0.000994	...	0.002535
2	CywHbj2cUaVYgp9xhlDk	0.005374	0.000624	0.000130	0.000249	0.000129	0.000100	0.000105	0.000168	0.000174	...	0.000408
3	eF5UtAWbwBrYJIRVpNik	0.008818	0.000957	0.000176	0.000247	0.000174	0.000207	0.000123	0.000221	0.000222	...	0.000976
4	5scq0VdirPRF8DxUKbh1	0.037465	0.000991	0.000251	0.000246	0.000315	0.000366	0.000252	0.000369	0.000447	...	0.001527

5 rows × 261 columns

In [24]:

```
result_asm.head()
```

Out[24]:

	ID	HEADER:	.text:	.Pav:	.idata:	.data:	.bss:	.rdata:	.edata:	.rsrc:	...	esi	eax
0	01kcPWA9K2BOxQeS5Rju	0.107345	0.001092	0.0	0.000761	0.000023	0.0	0.000084	0.0	0.000072	...	0.000746	0.000301
1	1E93CpP60RHFNiT5Qfvn	0.096045	0.001230	0.0	0.000617	0.000019	0.0	0.000000	0.0	0.000072	...	0.000328	0.000965
2	3ekVow2ajZHbTnBcsDfX	0.096045	0.000627	0.0	0.000300	0.000017	0.0	0.000038	0.0	0.000072	...	0.000475	0.000201
3	3X2nY7iQaPBIVDrAZqJe	0.096045	0.000333	0.0	0.000258	0.000008	0.0	0.000000	0.0	0.000072	...	0.000090	0.000281
4	46OZzdsSKDCFV8h7XWxf	0.096045	0.000590	0.0	0.000353	0.000068	0.0	0.000000	0.0	0.000072	...	0.000102	0.000362

5 rows × 54 columns

In [25]:

```
print(result.shape)
print(result_asm.shape)
```

```
(10868, 261)
(10868, 54)
```

In [26]:

```

result_x = pd.merge(result,result_asm.drop(['Class'], axis=1),on='ID', how='left')
result_y = result_x['Class']
result_x = result_x.drop(['ID','rtn','.BSS','.CODE','Class'], axis=1)
result_x.head()

```

Out[26]:

	0	1	2	3	4	5	6	7	8	9	...	edx	esi	ea
0	0.002525	0.000082	0.000013	0.000017	0.000016	0.000012	0.000004	0.000003	0.000099	0.000004	...	0.001622	0.000452	0.00233
1	0.010740	0.001771	0.000416	0.000489	0.000886	0.000320	0.000332	0.000662	0.000994	0.001810	...	0.015494	0.012723	0.02684
2	0.005374	0.000624	0.000130	0.000249	0.000129	0.000100	0.000105	0.000168	0.000174	0.000170	...	0.001546	0.000960	0.00108
3	0.008818	0.000957	0.000176	0.000247	0.000174	0.000207	0.000123	0.000221	0.000222	0.000236	...	0.000324	0.010678	0.00038
4	0.037465	0.000991	0.000251	0.000246	0.000315	0.000366	0.000252	0.000369	0.000447	0.000460	...	0.008968	0.004260	0.01489

5 rows × 308 columns

#### 4.5.2. Multivariate Analysis on final features

In [34]:

```
plt.close()

xtsne=TSNE(perplexity=50)
results=xtsne.fit_transform((result_x))
vis_x = results[:, 0]
vis_y = results[:, 1]
plt.scatter(vis_x, vis_y, c=result_y, cmap=plt.cm.get_cmap("jet", 9))
plt.colorbar(ticks=range(9))
plt.clim(0.5, 9)
plt.show()
```

#### 4.5.3. Train and Test split

In [44]:

```
X_train, X_test_merge, y_train, y_test_merge = train_test_split(result_x, result_y,stratify=result_y,test_size=0.20)
X_train_merge, X_cv_merge, y_train_merge, y_cv_merge = train_test_split(X_train, y_train,stratify=y_train,test_size=0.20)
```

#### 4.5.4. Random Forest Classifier on final features

In [90]:

```
%%time
plt.close()
# -----
# default parameters
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,
# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
# min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
# verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forests-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
from sklearn.ensemble import RandomForestClassifier
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=r_cfl.classes_, eps=1e-15))

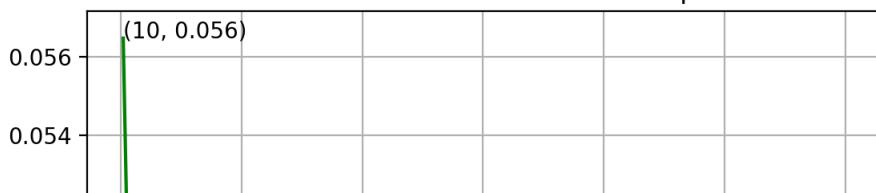
for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

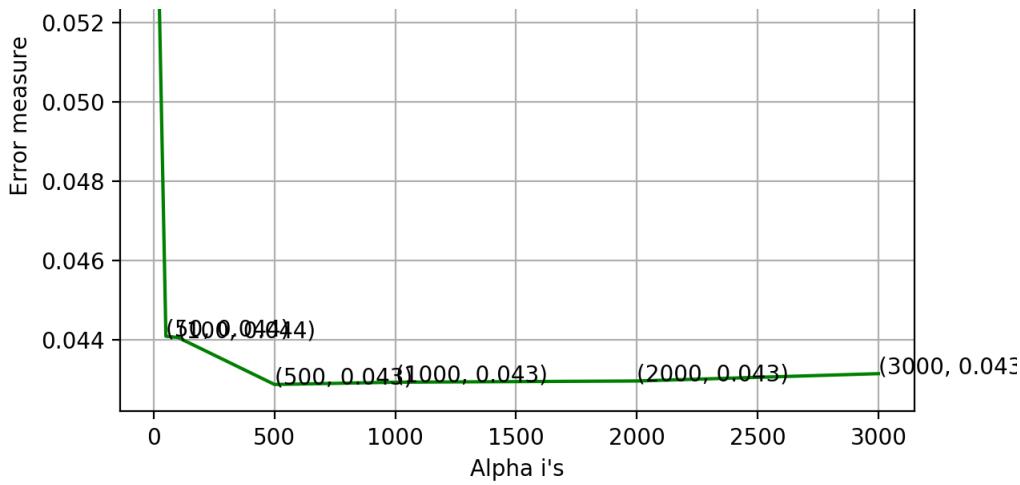
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```

```
log_loss for c = 10 is 0.056458997763404746
log_loss for c = 50 is 0.04409187353003727
log_loss for c = 100 is 0.044061068765813394
log_loss for c = 500 is 0.042874337981406434
log_loss for c = 1000 is 0.042932057098188925
log_loss for c = 2000 is 0.04296267385152051
log_loss for c = 3000 is 0.043147743596854915
```

Cross Validation Error for each alpha





```
CPU times: user 26min 28s, sys: 18.4 s, total: 26min 46s
Wall time: 3min 59s
```

In [91]:

```
%%time
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train_merge,y_train_merge)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

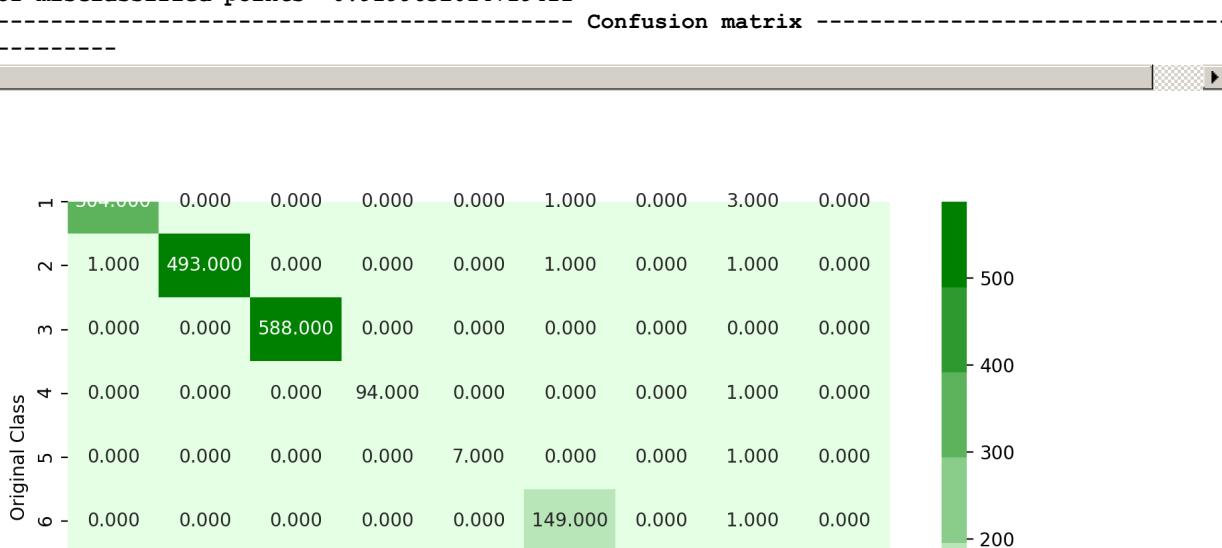
predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))
```

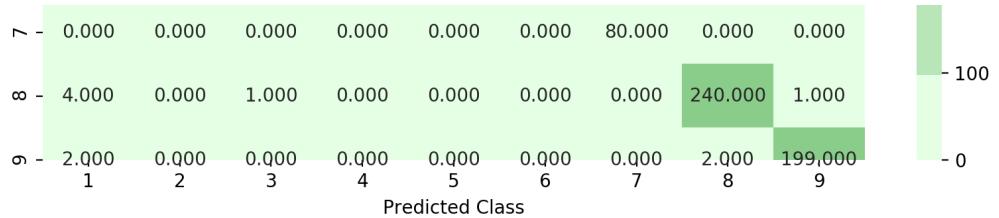
```
For values of best alpha = 500 The train log loss is: 0.014762155486539353
For values of best alpha = 500 The cross validation log loss is: 0.042874337981406434
For values of best alpha = 500 The test log loss is: 0.04192004241532452
CPU times: user 2min 3s, sys: 1.97 s, total: 2min 5s
Wall time: 20.2 s
```

In [92]:

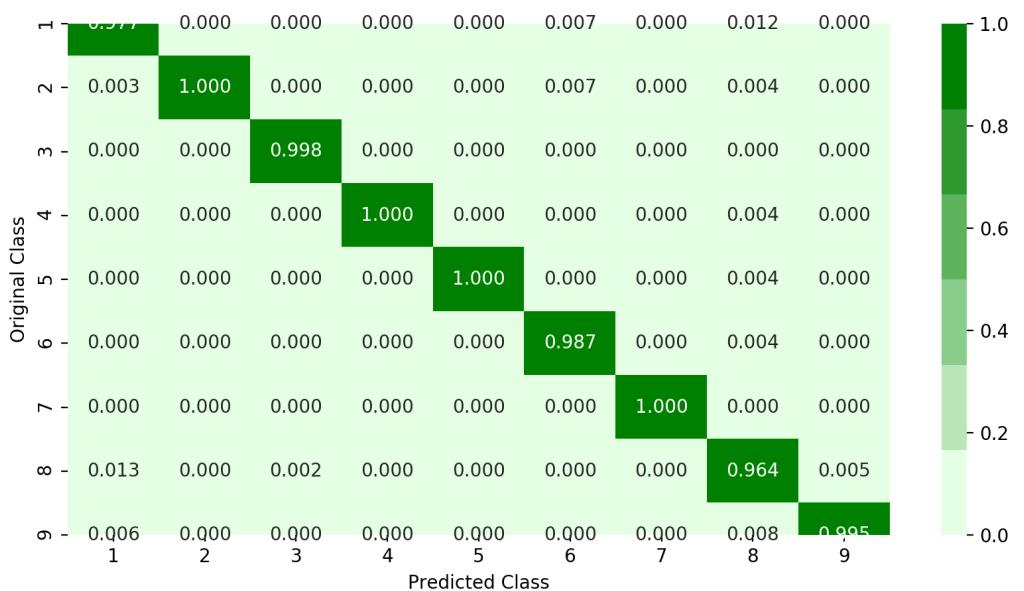
```
plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 0.9199632014719411





Precision matrix



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

Recall matrix



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

#### 4.5.5. XgBoost Classifier on final features

In [32]:

```
%%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClассifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClассifier
# -----
# default parameters
# class xgboost.XGBClассifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# ----

alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClассifier(n_estimators=i)
    x_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=x_cfl.classes_, eps=1e-15))

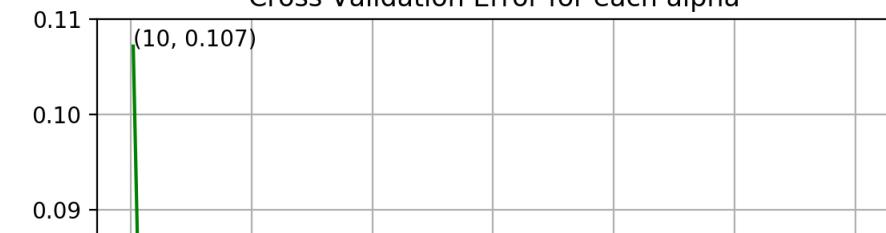
for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

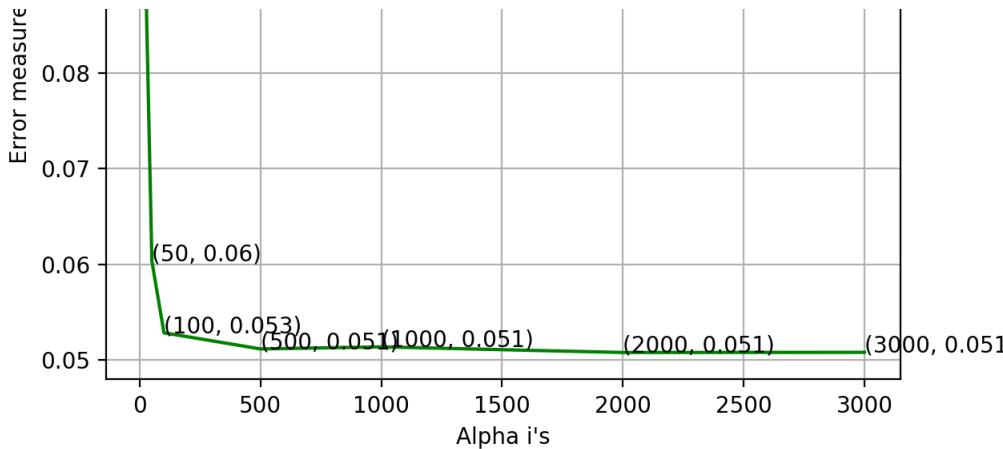
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c =  10 is  0.1071910897830746
log_loss for c =  50 is  0.06031120083021808
log_loss for c =  100 is  0.052799286967522965
log_loss for c =  500 is  0.0511063291087567
log_loss for c =  1000 is  0.05132551659041581
log_loss for c =  2000 is  0.050734482379753906
log_loss for c =  3000 is  0.05075047417606582
```

Cross Validation Error for each alpha





CPU times: user 1h 47min 43s, sys: 552 ms, total: 1h 47min 44s  
Wall time: 1h 47min 44s

In [33]:

```
%%time
plt.close()
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))
```

For values of best alpha = 2000 The train log loss is: 0.009815691068426237  
For values of best alpha = 2000 The cross validation log loss is: 0.05075047417606582  
For values of best alpha = 2000 The test log loss is: 0.042754901893074994  
CPU times: user 42min 9s, sys: 56 ms, total: 42min 9s  
Wall time: 42min 9s

#### 4.5.5. XgBoost Classifier on final features with best hyper parameters using Random search

In [36]:

```
x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
    'subsample':[0.1,0.3,0.5,1]
}
random_cfl=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1, )
random_cfl.fit(X_train_merge, y_train_merge)
```

Fitting 3 folds for each of 10 candidates, totalling 30 fits

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:   46.4s
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:   4.1min
[Parallel(n_jobs=-1)]: Done  19 out of  30 | elapsed:   7.8min remaining:   4.5min
[Parallel(n_jobs=-1)]: Done  23 out of  30 | elapsed:   8.0min remaining:   2.4min
[Parallel(n_jobs=-1)]: Done  27 out of  30 | elapsed:   9.5min remaining:   1.1min
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed: 10.3min finished
```

In [36]:

```
RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                    estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                           colsample_bylevel=1,
                                           colsample_bynode=1,
                                           colsample_bytree=1, gamma=0,
                                           learning_rate=0.1, max_delta_step=0,
                                           max_depth=3, min_child_weight=1,
                                           missing=None, n_estimators=100,
                                           n_jobs=1, nthread=None,
                                           objective='binary:logistic',
                                           random_state=0, reg_alpha=0.5,
                                           seed=None, silent=None, subsample=1,
                                           verbosity=1),
                    iid='warn', n_iter=10, n_jobs=-1,
                    param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                    pre_dispatch='2*n_jobs', random_state=None, refit=True,
                    return_train_score=False, scoring=None, verbose=10)
```

In [37]:

```
print (random_cfl.best_params_)

{'subsample': 0.5, 'n_estimators': 2000, 'max_depth': 3, 'learning_rate': 0.05,
 'colsample_bytree': 0.5}
```

In [38]:

```
%%time
# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----


x_cfl=XGBClassifier(n_estimators=2000,max_depth=3,learning_rate=0.05,colsample_bytree=0.5,subsample=0.5,nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))
```

```

In [38]: log_loss(y_test_merge, predict_y)

For values of best alpha = 2000 The train log loss is: 0.010227838888593169
For values of best alpha = 2000 The cross validation log loss is: 0.046105797578046656
For values of best alpha = 2000 The test log loss is: 0.03857589259645944
CPU times: user 21min 21s, sys: 120 ms, total: 21min 21s
Wall time: 21min 21s

```

In [39]:

```

plt.close()
plot_confusion_matrix(y_test_merge, sig_clf.predict(X_test_merge))

```

Number of misclassified points 0.78196872125115

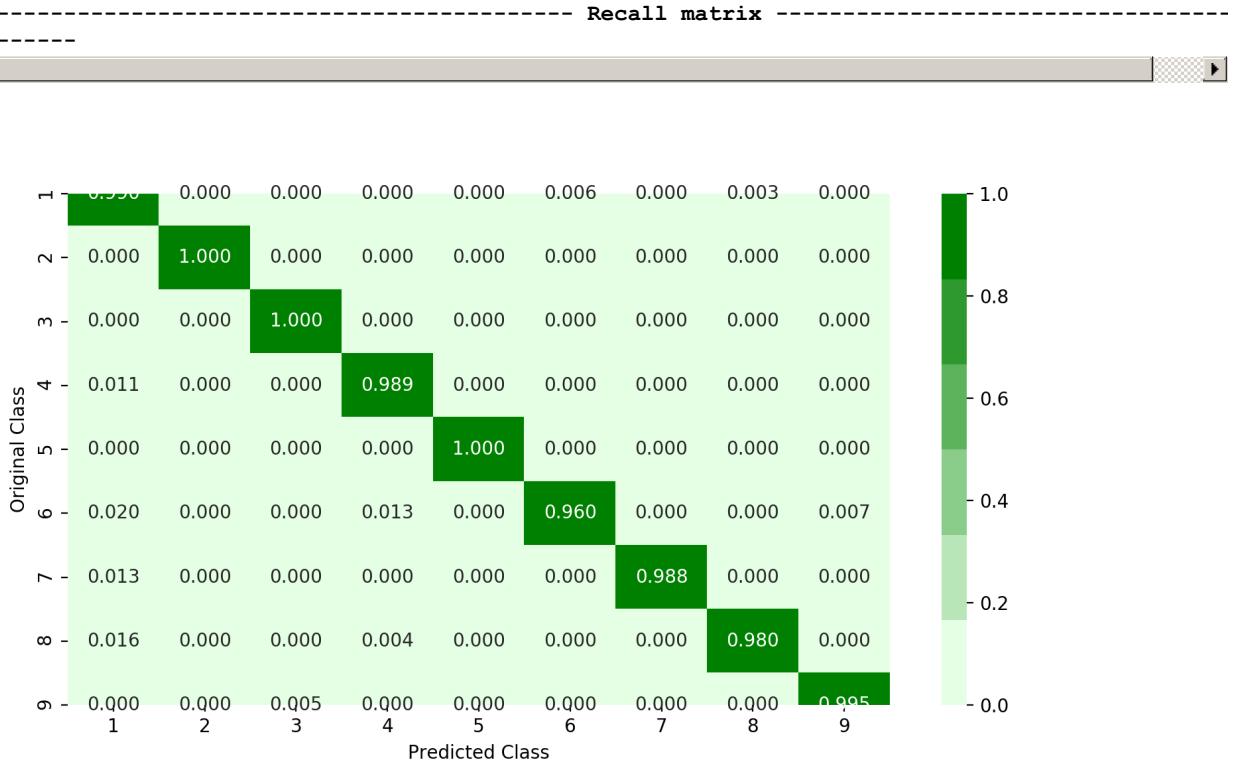
----- Confusion matrix -----



----- Precision matrix -----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1.]



```
Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

## Bytes Files Model Comparision

In [50]:

```
from prettytable import PrettyTable

table = PrettyTable()
table.field_names = ['Model', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']

table.add_row(['Random Model', '-', 2.4765227422803386, 2.5086309644540004, 88.9604415823367])
table.add_row(['K Nearest Neighbour', 0.13309130472155042, 0.2089185330714807, 0.20041775276335702, 5.105795768169273])
table.add_row(['Logistic Regression', 0.5161568454325155, 0.5378658285030614, 0.5287814318994476, 12.051517939282428])
table.add_row(['Random Forest Classifier', 0.028706691980921458, 0.08498094708786469, 0.06130664008293945, 1.3339466421343145])
table.add_row(['XgBoost Classification', 0.023248932222341635, 0.0697311485102128, 0.08183939406086423, 0.8279668813247469])
print(table)
```

Model Number of Misclassified Points	Train Log Loss	CV Log Loss	Test Log Loss	Num
Random Model 88.9604415823367	-	2.4765227422803386	2.5086309644540004	
K Nearest Neighbour 5.105795768169273	0.13309130472155042	0.2089185330714807	0.20041775276335702	
Logistic Regression 12.051517939282428	0.5161568454325155	0.5378658285030614	0.5287814318994476	
Random Forest Classifier 1.3339466421343145	0.028706691980921458	0.08498094708786469	0.06130664008293945	
XgBoost Classification 0.8279668813247469	0.023248932222341635	0.0697311485102128	0.08183939406086423	

## ASM Files Model Comparision

In [51]:

```
table = PrettyTable()
table.field_names = ['Model', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']

table.add_row(['K Nearest Neighbour', 0.037475912101338785, 0.09551157959211777,
0.07985178727072437, 1.1039558417663293])
table.add_row(['Logistic Regression', 0.3951231421379966, 0.40949221859900775, 0.39929985766626985
, 8.693652253909843])
table.add_row(['Random Forest Classifier', 0.021489116411912302, 0.036614779171787376,
0.03239270498531275, 0.5059797608095675])
table.add_row(['XgBoost Classification', 0.019288923540468395, 0.03159265585460439,
0.02576461657919649, 0.45998160073597055])
print(table)
```

Model Number of Misclassified Points	Train Log Loss	CV Log Loss	Test Log Loss	Nu
K Nearest Neighbour 1.1039558417663293	0.037475912101338785	0.09551157959211777	0.07985178727072437	
Logistic Regression 8.693652253909843	0.3951231421379966	0.40949221859900775	0.39929985766626985	
Random Forest Classifier 0.5059797608095675	0.021489116411912302	0.036614779171787376	0.03239270498531275	
XgBoost Classification 0.45998160073597055	0.019288923540468395	0.03159265585460439	0.02576461657919649	

## Bytes + ASM Files Model Comparision

In [53]:

```
table = PrettyTable()
table.field_names = ['Model', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']

table.add_row(['Random Forest Classifier', 0.014762155486539353, 0.042874337981406434,
0.04192004241532452, 0.9199632014719411])
table.add_row(['XgBoost Classification', 0.010227838888593169, 0.046105797578046656,
0.03857589259645944, 0.78196872125115])
print(table)
```

Model Number of Misclassified Points	Train Log Loss	CV Log Loss	Test Log Loss	Nu
Random Forest Classifier 0.9199632014719411	0.014762155486539353	0.042874337981406434	0.04192004241532452	
XgBoost Classification 0.78196872125115	0.010227838888593169	0.046105797578046656	0.03857589259645944	

## 5. Assignments

1. Add bi-grams on byte files and improve the log-loss
2. Watch the video ([video](#)) and include pixel intensity features to improve the logloss

1. you need to download the train from kaggle, which is of size ~17GB, after extracting it will occupy ~128GB data your dirve
2. if you are having computation power limitations, you can try using google colab, with GPU option enabled (you can search for how to enable GPU in colab) or you can work with the Google Cloud, check this tutorials by one of our student:  
[https://www.youtube.com/channel/UCRH\\_z-oM0LROvHPe\\_KYR4Wg](https://www.youtube.com/channel/UCRH_z-oM0LROvHPe_KYR4Wg) (we suggest you to use GCP over Colab)
3. To Extract the .7z file in google cloud, once after you upload the file into server, in your ipython notebook create a new cell and write these commands
  - a. !sudo apt-get install p7zip
  - b. !7z x file\_name.7z -o path/where/you/want/to/extract

<https://askubuntu.com/a/341637>

## Assignment Part 1

- 1 Add bi-grams and n-grams on byte files and improve the log-loss

### 1.1 Process Bigram on ByteFiles

In [4]:

```
# list byteFiles directory
files = os.listdir('byteFiles')
```

In [5]:

```
len(files)
```

Out[5]:

10868

In [7]:

```
# all_keys = []

def calc_bi_grams(file):
    """
    Calculate unigram and bigram
    """
    temp_list = []
    with open('byteFiles/'+file,"r") as byte_file:
        all_lines = []
        for lines in byte_file:
            line=lines.rstrip().split(" ")
            all_lines.extend(line)
        # unigrams
        for hex_code in line:
            temp_list.append(hex_code.lower())
    temp_list = list(set(temp_list))
    # bigrams
    bi_g = [' '.join(x) for x in list(ngrams(all_lines, 2))]
    for hex_code in bi_g:
        temp_list.append(hex_code.lower())
    temp_list = list(set(temp_list))
    return temp_list
```

In [7]:

```
# Process and collect bigram of bytes file
from tqdm import tqdm
for file in tqdm(files):
```

```
all_keys.extend(calc_bi_grams(file))
all_keys = list(set(all_keys))

len(all_keys)
```

In [9]:

```
all_keys
```

In [12]:

```
with open('uni_bigram_keys.pkl', 'wb') as big:
    pickle.dump(all_keys, big)
```

In [13]:

```
with open('uni_bigram_keys.pkl', 'rb') as big:
    all_keys = pickle.load(big)
```

In [14]:

```
len(all_keys)
```

Out[14]:

66306

In [8]:

```
a="00,01,02,03,04,05,06,07,08,09,0a,0b,0c,0d,0e,0f,10,11,12,13,14,15\
,16,17,18,19,1a,1b,1c,1d,1e,1f,20,21,22,23,24,25,26,27,28,29\
,2a,2b,2c,2d,2e,2f,30,31,32,33,34,35,36,37,38,39,3a,3b,3c,3d,\
3e,3f,40,41,42,43,44,45,46,47,48,49,4a,4b,4c,4d,4e,4f,50,51,52,\
53,54,55,56,57,58,59,5a,5b,5c,5d,5e,5f,60,61,62,63,64,65,66,67,68,\
69,6a,6b,6c,6d,6e,6f,70,71,72,73,74,75,76,77,78,79,7a,7b,7c,7d,7e,7f,\
80,81,82,83,84,85,86,87,88,89,8a,8b,8c,8d,8e,8f,90,91,92,93,94,95,96,\
97,98,99,9a,9b,9c,9d,9e,9f,a0,a1,a2,a3,a4,a5,a6,a7,a8,a9,aa,ab,ac,ad,\
ae,af,b0,b1,b2,b3,b4,b5,b6,b7,b8,b9,ba,bb,bc,bd,be,bf,c0,c1,c2,c3,c4,c5, \
c6,c7,c8,c9,ca,cb,cc,cd,ce,cf,d0,d1,d2,d3,d4,d5,d6,d7,d8,d9,da,db,dc,dd,de, \
df,e0,e1,e2,e3,e4,e5,e6,e7,e8,e9,ea,eb,ec,ed,ee,ef,f0,f1,f2,f3,f4,f5, \
f6,f7,f8,f9,fa,fb,fc,fd,fe,ff,??" \
a = a.replace(", ", " ")
a_uni = a.split(" ")
len(a_uni)
```

Out[8]:

257

In [18]:

```
# all_keys.extend(a_uni)
```

In [9]:

```
len(all_keys)
```

Out[9]:

66306

In [9]:

```
all_keys
```

Out[9]:

['1d',

'7f 4b',  
'8a 85',  
'55 c6',  
'4e 68',  
'55 69',  
'bd 52',  
'b8 13',  
'01 91',  
'32 f9',  
'96 a5',  
'fa 51',  
'd6 68',  
'a4 68',  
'31 3d',  
'a5 3c',  
'83 6c',  
'a4 59',  
'45 e5',  
'd2 ce',  
'4f 43',  
'8a 12',  
'05 09',  
'89 4f',  
'a4 d4',  
'd5 5f',  
'08 cd',  
'e8 84',  
'32 5c',  
'1c 1d',  
'cc 23',  
'f9 d6',  
'1e 7c',  
'd7 21',  
'6d 1a',  
'3d e9',  
'69 d9',  
'e5 a3',  
'3f 70',  
'60 bc',  
'74 c3',  
'a0 cc',  
'fd 7b',  
'09 50',  
'bc cd',  
'f8 02',  
'ad 23',  
'5f 23',  
'e2 d4',  
'e0 6e',  
'59 e6',  
'd3 8b',  
'b1 fe',  
'9e 67',  
'68 f6',  
'30 55',  
'd2 a3',  
'43 7c',  
'c5 d2',  
'c9 5e',  
'ff ff',  
'1d fb',  
'fb d3',  
'9f 15',  
'be c0',  
'fc 86',  
'0a 12',  
'ad 50',  
'48 2b',  
'3a e7',  
'd9 8e',  
'94 e3',  
'd1 3a',  
'86 c2',  
'c0 68',  
'ca 74',  
'8d 9b',  
'86 2a',

'c8 df',  
'8b 84',  
'ea 07',  
'a6 ea',  
'6b e1',  
'd1 bf',  
'20 08',  
'ea 46',  
'a7 7e',  
'56 17',  
'ed 70',  
'de e0',  
'09 45',  
'e7 eb',  
'16 4e',  
'a7 99',  
'79 ??',  
'8e 79',  
'a1 a4',  
'1a 2e',  
'66 0a',  
'70 57',  
'6f 09',  
'14 90',  
'89 8e',  
'1e 17',  
'75 52',  
'6a 8f',  
'6d 54',  
'fe 26',  
'7e a0',  
'7e 3e',  
'14 0a',  
'5a 37',  
'77 d4',  
'c0 a5',  
'59 53',  
'2c 2a',  
'1c ce',  
'df 9a',  
'68 7a',  
'cd 01',  
'4f a3',  
'6d 10',  
'4f 6c',  
'ed 8b',  
'79 fd',  
'ae b6',  
'f6 a2',  
'd6 8b',  
'cc bd',  
'dc 3e',  
'db 6a',  
'99 68',  
'aa 3c',  
'03 80',  
'58 4b',  
'a7 60',  
'24 2b',  
'62 34',  
'e6 4b',  
'32 31',  
'74 f1',  
'b9 41',  
'09 6e',  
'95 4b',  
'c2 f8',  
'dc 40',  
'7c 20',  
'4d bf',  
'aa 0d',  
'91 ae',  
'da a5',  
'29 c5',  
'50 c3',  
'84 4c',  
'a2 f1',

'67 e2',  
'63 76',  
'46 a5',  
'68 ff',  
'33 40',  
'df aa',  
'4b aa',  
'f7 b2',  
'c1 25',  
'5b b5',  
'9c 9d',  
'be e2',  
'87 b7',  
'21 2b',  
'dc 34',  
'ff 5a',  
'b7 cd',  
'4e da',  
'30 bb',  
'c5 0e',  
'bd 65',  
'8f 84',  
'b7 18',  
'7f 3a',  
'50 25',  
'5d be',  
'2d a8',  
'26 d4',  
'8f 88',  
'69 a0',  
'ac ff',  
'5b 99',  
'09 0c',  
'7b bd',  
'b1 92',  
'c5 fa',  
'35 cf',  
'29 f6',  
'73 d4',  
'41 2f',  
'0d 0a',  
'a3 6a',  
'76 30',  
'dd 6b',  
'3a 0c',  
'58 68',  
'86 95',  
'0b de',  
'14 31',  
'12 05',  
'8c c0',  
'3b 3f',  
'02 9f',  
'd8 d3',  
'f0 9e',  
'61 6b',  
'6e ae',  
'a8 f5',  
'4c d3',  
'27 33',  
'53 0d',  
'23 43',  
'06 41',  
'3a d4',  
'46 ce',  
'7a a8',  
'65 61',  
'ee 8c',  
'7a 77',  
'95 22',  
'9a 1f',  
'7c 01',  
'c9 35',  
'97 f6',  
'b0 2a',  
'b2 0d',  
'8d 0b',

'c3 86',  
'ab 70',  
'9d 33',  
'6a 40',  
'87 1e',  
'b8 10',  
'6f c4',  
'e1 98',  
'3f 5e',  
'93 01',  
'fa 57',  
'dd 0f',  
'41 83',  
'd7 6c',  
'7d 2b',  
'1e be',  
'd0 0f',  
'bd c4',  
'58 0b',  
'de 5b',  
'a7 7c',  
'11 57',  
'bd aa',  
'4d 27',  
'be 17',  
'1e 1f',  
'3f 2c',  
'66 99',  
'66 59',  
'74 98',  
'f3 24',  
'e2 5d',  
'ad 49',  
'35 0b',  
'07 ec',  
'00 94',  
'eb 09',  
'?? d8',  
'e5 05',  
'30 7e',  
'e4 08',  
'77 48',  
'f6 33',  
'20 2c',  
'92 6e',  
'aa 06',  
'eb b5',  
'8a 74',  
'66 61',  
'62 10',  
'40 07',  
'4b a6',  
'41 58',  
'15 66',  
'00 b4',  
'ea 29',  
'1a eb',  
'ab 96',  
'35 1b',  
'd1 2d',  
'd9',  
'87 57',  
'3b 16',  
'95 7e',  
'16 b5',  
'7b 0e',  
'90 67',  
'7e e0',  
'14 3b',  
'70 aa',  
'fc b6',  
'21 66',  
'5f 89',  
'2d f9',  
'da 53',  
'36 14',  
'3f 00'.

-- --'  
'99 3a',  
'd7 f3',  
'65 23',  
'61 bf',  
'9a fb',  
'bc 84',  
'61 2a',  
'2d fb',  
'85 8b',  
'58 03',  
'22 8b',  
'd9 3e',  
'ea 8c',  
'f6 80',  
'7a e4',  
'9b b8',  
'4f 28',  
'e8 d4',  
'ac f2',  
'c7 9c',  
'ea c7',  
'07 14',  
'ff 3c',  
'10 43',  
'43 64',  
'6a ec',  
'2e 12',  
'66 82',  
'f0 74',  
'40 90',  
'c3 8c',  
'fc 83',  
'c3 c1',  
'11 1e',  
'4b 81',  
'c9 f7',  
'16 ed',  
'18 aa',  
'1b 7d',  
'65 f0',  
'78 47',  
'0e 5b',  
'7f 44',  
'42 de',  
'18 1f',  
'98 78',  
'4f d2',  
'13 1e',  
'2c fb',  
'38 b9',  
'd8 d7',  
'14 14',  
'05 d0',  
'08 f3',  
'b1 d4',  
'52 80',  
'd6 86',  
'7e 9d',  
'89 6c',  
'11 c2',  
'c8 5c',  
'f4 01',  
'6c 70',  
'21 31',  
'4d 04',  
'08 a8',  
'df 71',  
'7b 90',  
'b2 49',  
'7c ab',  
'eb 49',  
'21 d5',  
'd1 68',  
'0f 50',  
'14 1b',  
'88 99',  
'7c 25'

'  
'7b 08',  
'cf 6d',  
'70 80',  
'8f 12',  
'17 3d',  
'fa b7',  
'9c 34',  
'ec ac',  
'47 b9',  
'61 f3',  
'53 9b',  
'99 60',  
'80 f0',  
'38 bf',  
'db be',  
'f0 70',  
'aa 40',  
'a2 7e',  
'fd 73',  
'd7 eb',  
'ca b2',  
'be a4',  
'a0 b1',  
'd0 18',  
'd2 eb',  
'05 4f',  
'98 ef',  
'25 2d',  
'0a 68',  
'00 08',  
'ce 11',  
'44 0d',  
'63 cc',  
'22 ad',  
'05 6d',  
'72 5f',  
'b4 04',  
'bf b7',  
'f2 1d',  
'10 ff',  
'e4 92',  
'56 e5',  
'41 78',  
'19 0c',  
'aa b2',  
'a1 a1',  
'cf 17',  
'86 d7',  
'c0 17',  
'15 cb',  
'bf 0a',  
'27 20',  
'a9 fa',  
'0d 30',  
'9b c1',  
'35 22',  
'8e 7d',  
'44 ed',  
'25 72',  
'56 a2',  
'2e 76',  
'60 4a',  
'59 69',  
'93 35',  
'54 17',  
'da 1c',  
'f3 48',  
'4a 30',  
'23 bf',  
'a4 51',  
'5a 8f',  
'45 46',  
'76 f3',  
'09 e8',  
'd1 40',  
'6e c5',  
'7a 2a'

'a aa',  
'b4 99',  
'f9 d2',  
'65 f3',  
'7c 9b',  
'f9 9e',  
'ee af',  
'e6 85',  
'6d 21',  
'1e 47',  
'0c 50',  
'c1 07',  
'2c 77',  
'32 c7',  
'13 7d',  
'ae f1',  
'11 55',  
'5e 51',  
'2c ee',  
'2e 97',  
'da 02',  
'd6 fa',  
'd1 20',  
'8f 3a',  
'c6 c2',  
'a4 5d',  
'5c b3',  
'15 64',  
'2b f0',  
'4f 54',  
'eb 77',  
'b3 65',  
'3c 89',  
'7e b9',  
'd1 0a',  
'06 d4',  
'32 d2',  
'6d 87',  
'61 89',  
'2b e6',  
'dd 3a',  
'b8 d6',  
'60 0f',  
'a0 19',  
'74 22',  
'5c a8',  
'28 cc',  
'9b d8',  
'63 1c',  
'54 3e',  
'21 74',  
'fe f3',  
'f9 09',  
'2d 93',  
'69 f9',  
'fb 1f',  
'eb 42',  
'80 bf',  
'54 d2',  
'f1 37',  
'7f e6',  
'db 9f',  
'26 ??',  
'90 09',  
'7d dc',  
'f6 aa',  
'4d e2',  
'46 81',  
'44 47',  
'1a 95',  
'1e ac',  
'1a 20',  
'e3 d4',  
'77 8a',  
'9b 4e',  
'9f da',  
'f0 bf',  
'e2 df'

'  
'0f 64',  
'd8 de',  
'e4 b6',  
'ad 92',  
'fa d6',  
'df 50',  
'ca 4c',  
'6d 33',  
'73 62',  
'57 6a',  
'd8 22',  
'94 4d',  
'51 91',  
'17 a4',  
'5a 1a',  
'9d 1b',  
'81 83',  
'1c fe',  
'3d e5',  
'e8 f8',  
'eb 63',  
'a4 31',  
'f5 52',  
'1e 4b',  
'52 54',  
'9c b9',  
'8a ca',  
'88 8f',  
'59 ad',  
'5c 83',  
'74 2e',  
'd3 4e',  
'b3 fb',  
'7f 0d',  
'd5 21',  
'f8 ab',  
'3c 6f',  
'4f 00',  
'c7 71',  
'aa 8b',  
'8f 0f',  
'f9 87',  
'71 1f',  
'a1 c0',  
'61 6d',  
'3e 19',  
'60 d9',  
'b7 f7',  
'49 ae',  
'df 88',  
'8e 6c',  
'63 c5',  
'24 67',  
'd0 24',  
'89 22',  
'27 16',  
'ad 07',  
'87 ef',  
'50 47',  
'ae 6f',  
'ac f5',  
'11 0e',  
'cb 32',  
'1b 43',  
'4d a7',  
'0b a6',  
'4a 46',  
'6d 7a',  
'41',  
'98 08',  
'd1 55',  
'2a 13',  
'2b 9a',  
'1b f2',  
'e6 a7',  
'55 1d',  
'02 ~4'

'**u4** **a4**',  
'**97** **6b**',  
'**21** **c2**',  
'**67** **fc**',  
'**84** **2b**',  
'**a0** **5c**',  
'**0d** **28**',  
'**af** **c7**',  
'**5d** **f6**',  
'**8a** **02**',  
'**e8** **60**',  
'**1b** **87**',  
'**36** **fd**',  
'**96** **12**',  
'**09** **0f**',  
'**d8**',  
'**7c** **9d**',  
'**e5** **6e**',  
'**92** **58**',  
'**42** **94**',  
'**35** **90**',  
'**6a** **e9**',  
'**11** **eb**',  
'**c7** **05**',  
'**52** **56**',  
'**f9** **d5**',  
'**b9** **60**',  
'**10** **84**',  
'**b2** **53**',  
'**89** **ea**',  
'**ac** **6c**',  
'**8d** **60**',  
'**15** **f7**',  
'**22** **e9**',  
'**f1** **23**',  
'**37** **a5**',  
'**4b** **ff**',  
'**a8** **be**',  
'**c5** **5e**',  
'**d3** **c6**',  
'**9b** **63**',  
'**40** **73**',  
'**b4** **cb**',  
'**80** **73**',  
'**23** **3a**',  
'**9a** **cf**',  
'**1c** **e7**',  
'**4a** **dd**',  
'**2d** **24**',  
'**15** **4c**',  
'**37** **d5**',  
'**20** **40**',  
'**20** **c1**',  
'**ae** **1f**',  
'**72** **73**',  
'**62** **f1**',  
'**52** **94**',  
'**9e** **ef**',  
'**60** **b4**',  
'**01** **ea**',  
'**a1** **e3**',  
'**??** **06**',  
'**f5** **4e**',  
'**1d** **20**',  
'**56** **a6**',  
'**dd** **d8**',  
'**4f** **5b**',  
'**3b** **ca**',  
'**81** **73**',  
'**3f** **e1**',  
'**57** **d4**',  
'**ec** **74**',  
'**9a** **c2**',  
'**d1** **01**',  
'**dd** **ba**',  
'**a7** **25**',  
'**f3** **45**',  
'**ce** **5-**'

'**61** **5a'**,  
'83 **3f'**,  
'50 **5d'**,  
'99 **44'**,  
'6f **4c'**,  
'e1 **7e'**,  
'ba **53'**,  
'a1 **8f'**,  
'32 **0d'**,  
'2d **5a'**,  
'f6 **22'**,  
'f2 **ef'**,  
'e6 **07'**,  
'61 **b5'**,  
'a6 **5e'**,  
'ea **01'**,  
'29 **c4'**,  
'80 **78'**,  
'34 **a3'**,  
'6c **3d'**,  
'63 **a9'**,  
'b2 **51'**,  
'13 **50'**,  
'e4 **40'**,  
'39 **b1'**,  
'18 **61'**,  
'e5 **2c'**,  
'12 **5c'**,  
'27 **f9'**,  
'e8 **de'**,  
'29 **7a'**,  
'c4 **92'**,  
'5e **53'**,  
'58 **7d'**,  
'17 **90'**,  
'86 **fb'**,  
'4c **f2'**,  
'12 **87'**,  
'99 **32'**,  
'c2 **c6'**,  
'c4 **5f'**,  
'52 **9a'**,  
'6a **5a'**,  
'e7 **3c'**,  
'08 **57'**,  
'39 **88'**,  
'6b **65'**,  
'eb **51'**,  
'88 **1e'**,  
'62 **e5'**,  
'7b **29'**,  
'99 **05'**,  
'23 **28'**,  
'1d **73'**,  
'6b **46'**,  
'd8 **4a'**,  
'5d **34'**,  
'c6 **5b'**,  
'a2 **03'**,  
'94 **2b'**,  
'00 **34'**,  
'be **b5'**,  
'23 **5a'**,  
'39 **ed'**,  
'00 **01'**,  
'd2 **7f'**,  
'?? **48'**,  
'ba **c9'**,  
'5f **20'**,  
'f1 **f4'**,  
'd7 **fc'**,  
'5e **ee'**,  
'da **5a'**,  
'b2 **04'**,  
'99 **64'**,  
'0d **1e'**,  
'e4 **2b'**,  
'aa **cc'**

'30 06',  
'4d cb',  
'8a e9',  
'29 ac',  
'82 29',  
'd4 c6',  
'ca 28',  
'05 84',  
'4a 56',  
'07 e9',  
'ab f9',  
'cf 26',  
'9e 9a',  
'2f 47',  
'4c 77',  
'81 da',  
'61 ed',  
'eb cd',  
'05 f8',  
'a6 d3',  
'ea 18',  
'42 e2',  
'5d 89',  
'07 e3',  
'5b 29',  
'32 f5',  
'11 df',  
'40 dd',  
'd6 e7',  
'02 75',  
'4d 3d',  
'37 35',  
'93 fc',  
'0c 2a',  
'21 08',  
'96 d6',  
'be 6a',  
'12 50',  
'58 8b',  
'7a d8',  
'1b 6b',  
'5a 94',  
'75 84',  
'c7 8c',  
'b8 8e',  
'a5 0b',  
'9a eb',  
'57 50',  
'33 ??',  
'f9 b1',  
'77 19',  
'd2 36',  
'2f a5',  
'b4 46',  
'ca cb',  
'e8 72',  
'dc a5',  
'11 e0',  
'52 bc',  
'ef 2c',  
'3f e8',  
'05 49',  
'12 14',  
'ae 1a',  
'f0 82',  
'0b 6e',  
'4e d4',  
'77 71',  
'1f 18',  
'a9 ee',  
'00 11',  
'25 7b',  
'16 2a',  
'dc e5',  
'37 51',  
'b8 bc',  
'06 38',  
--- ---

'47 86',  
'b2 a3',  
'2f f8',  
'fc a0',  
'1d 99',  
'e1 e4',  
'27 87',  
'a0 90',  
'05 f4',  
'2b 33',  
'02 cc',  
'c3 7f',  
'6c 86',  
'37 fa',  
'08 8b',  
'06 f8',  
'bb 3f',  
'8c 59',  
'f7 d2',  
'0b 9d',  
'66 f5',  
'87 a0',  
'7d 24',  
'15 86',  
'ac bf',  
'84 f8',  
'67 78',  
'80 2e',  
'98 5d',  
'8a 76',  
'20 5f',  
'02 7a',  
'db 02',  
'61 e6',  
'35 be',  
'4f 05',  
'3e 5b',  
'a3 62',  
'71 13',  
'5f 87',  
'08 44',  
'a5 10',  
'85 1d',  
'94 f7',  
'ec f1',  
'e0 89',  
'13 3a',  
'ac 47',  
'1b db',  
'46 d4',  
'11 d5',  
'e9 d3',  
'2e 74',  
'fc ad',  
'f0 cb',  
'e3 0f',  
'e4 19',  
'fd 90',  
'3d 19',  
'52 a4',  
'26 ea',  
'a6 9a',  
'?? bd',  
'2b 7e',  
'14 d9',  
'7d 22',  
'c0 fc',  
'83 ff',  
'd2 ab',  
'cd da',  
'03 1d',  
'15 d9',  
'b4 83',  
'56 3e',  
'a6 a5',  
'7c 64',  
'd6 38',  
-----

```
'71 fa',
'a4 d2',
'24 fa',
'f5 8d',
'7b d5',
'28 9f',
'd3 94',
'51 93',
'e0 f6',
'39 23',
'5c d3',
'f0 fd',
'70 b0',
'71 dd',
'd8 37',
'ec ee',
'c5 d5',
'eb 23',
'd4 18',
'4b 29',
'ae 43',
'54 03',
'dd 17',
'3e cb',
'c7 c9',
'c0 8c',
'10 2c',
'11 5d',
'52 18',
'bb 23',
'06 b3',
'8f fd',
'a2 65',
'4d 17',
'ed a7',
'92 4e',
'0c ee',
'3f 3b',
'ee b4',
'64 f8',
'bd 8b',
'2b cd',
'78 56',
'b2 d1',
'04 28',
'fa 95',
'7d 60',
'ce a3',
'ab 2c',
'88 46',
'3e de',
'16 e1',
'39 4b',
'cb 7c',
'81 0c',
'6b b5',
'ed 75',
'02 c3',
'41 c8',
'e8 27',
'fc 2e',
'7c d5',
'bc 5a',
'40 78',
'd1 d8',
'42 c8',
'99 a5',
'a6 bd',
'5f 48',
'1b 48',
'a1 2c',
'5f b7',
'0e e5',
'f6 90',
'c2 94',
'04 37',
...]
```

In [10]:

```
import scipy
from scipy.sparse import csr_matrix
from tqdm import tqdm

# bytebigram_vect = csr_matrix((len(files), len(all_keys)))
# from tqdm import tqdm
# for i, file in tqdm(enumerate(os.listdir('byteFiles'))):
#     f = open('byteFiles/' + file)
#     bytebigram_vect[i, :] += csr_matrix(vector.fit_transform([f.read().replace('\n', '').lower()]))
#     f.close()

def extract_bigrams_byte_features(files):
    """
    Extract Bigram features and save the output file in npz file, This method is called in multithreaded environment.
    """
    pid = os.getpid()
    print('Process started with id = ', pid)
    ftot = len(files)

    str_pid = str(pid)

    vector_str_pid = CountVectorizer(lowercase=False, ngram_range=(2, 2), vocabulary=all_keys)
    bytebigram_vect_str_pid = csr_matrix((len(files), len(all_keys)))
    print('vector variable created with name ', 'bytebigram_vect', '_', str_pid)
    for i, file in enumerate(files):
        f = open('byteFiles/' + file)
        bytebigram_vect_str_pid[i, :] += csr_matrix(vector_str_pid.fit_transform([f.read().replace('\n', '').lower()]))
        f.close()
        print(pid, i + 1, 'of', ftot, 'files processed.')

    # Print progress
    # if (i+1) % 100 == 0:
    #     print(pid, i + 1, 'of', ftot, 'files processed.')

    filename = 'bytebigram_'+str_pid+'.npz'
    scipy.sparse.save_npz(filename, bytebigram_vect_str_pid) #'bytebigram.npz'

# extract_bigrams_byte_features('abc')
```

In [11]:

```
# Now divide the train files into four groups for multiprocessing, Six threads not works properly.
import time as tm
start_time = tm.time()
tfiles = os.listdir('byteFiles')
# 1810*6 = 10860 for last slot add 8
# sixth = 1810#int(len(tfiles)/4)
# print(quart)
# train1 = tfiles[:sixth]
# train2 = tfiles[sixth:(2*sixth)]
# train3 = tfiles[(2*sixth):(3*sixth)]
# train4 = tfiles[(3*sixth):(4*sixth)]
# train5 = tfiles[(4*sixth):(5*sixth)]
# train6 = tfiles[(5*sixth):]
# # print(len(train6))
# # print(len(tfiles), sixth,
# (len(train1)+len(train2)+len(train3)+len(train4)+len(train5)+len(train6)))
# trains = [train1, train2, train3, train4, train5, train6]
# p = Pool(6)
quart = int(len(tfiles)/4)
# print(quart)
train1 = tfiles[:quart]
train2 = tfiles[quart:(2*quart)]
train3 = tfiles[(2*quart):(3*quart)]
train4 = tfiles[(3*quart):]
print(len(tfiles), quart, (len(train1)+len(train2)+len(train3)+len(train4)))
trains = [train1, train2, train3, train4]
p = Pool(4)
p.map(extract_bigrams_byte_features, trains)
```

```
print("Elapsed time: {:.2f} hours.".format((tm.time() - start_time)/3600.0))
```

```
10868 2717 10868
Process started with id = 2034
Process started with id = 2035
Process started with id = 2036
Process started with id = 2037
vector variable created with name bytebigram_vect _ 2035
vector variable created with name bytebigram_vect _ 2034
vector variable created with name bytebigram_vect _ 2036
vector variable created with name bytebigram_vect _ 2037
2034 1 of 2717 files processed.
2034 2 of 2717 files processed.
2034 3 of 2717 files processed.
2034 4 of 2717 files processed.
2036 1 of 2717 files processed.
2035 1 of 2717 files processed.
2034 5 of 2717 files processed.
2036 2 of 2717 files processed.
2034 6 of 2717 files processed.
2037 1 of 2717 files processed.
2036 3 of 2717 files processed.
2035 2 of 2717 files processed.
2036 4 of 2717 files processed.
2035 3 of 2717 files processed.
2037 2 of 2717 files processed.
2034 7 of 2717 files processed.
2034 8 of 2717 files processed.
2035 4 of 2717 files processed.
2037 3 of 2717 files processed.
2037 4 of 2717 files processed.
2036 5 of 2717 files processed.
2037 5 of 2717 files processed.
2036 6 of 2717 files processed.
2035 5 of 2717 files processed.
2036 7 of 2717 files processed.
2036 8 of 2717 files processed.
2034 9 of 2717 files processed.
2035 6 of 2717 files processed.
2034 10 of 2717 files processed.
2034 11 of 2717 files processed.
2036 9 of 2717 files processed.
2034 12 of 2717 files processed.
2037 6 of 2717 files processed.
2034 13 of 2717 files processed.
2035 7 of 2717 files processed.
2037 7 of 2717 files processed.
2035 8 of 2717 files processed.
2034 14 of 2717 files processed.
2036 10 of 2717 files processed.
2037 8 of 2717 files processed.
2035 9 of 2717 files processed.
2036 11 of 2717 files processed.
2035 10 of 2717 files processed.
2034 15 of 2717 files processed.
2035 11 of 2717 files processed.
2036 12 of 2717 files processed.
2036 13 of 2717 files processed.
2037 9 of 2717 files processed.
2037 10 of 2717 files processed.
2034 16 of 2717 files processed.
2034 17 of 2717 files processed.
2037 11 of 2717 files processed.
2035 12 of 2717 files processed.
2036 14 of 2717 files processed.
2036 15 of 2717 files processed.
2034 18 of 2717 files processed.
2037 12 of 2717 files processed.
2034 19 of 2717 files processed.
2036 16 of 2717 files processed.
2034 20 of 2717 files processed.
2034 21 of 2717 files processed.
2037 13 of 2717 files processed.
2035 13 of 2717 files processed.
2037 14 of 2717 files processed.
2037 15 of 2717 files processed.
2036 17 of 2717 files processed.
```

2036 1/ or 2/1/ files processed.  
2034 22 of 2717 files processed.  
2034 23 of 2717 files processed.  
2037 16 of 2717 files processed.  
2035 14 of 2717 files processed.  
2037 17 of 2717 files processed.  
2034 24 of 2717 files processed.  
2037 18 of 2717 files processed.  
2035 15 of 2717 files processed.  
2036 18 of 2717 files processed.  
2036 19 of 2717 files processed.  
2037 19 of 2717 files processed.  
2035 16 of 2717 files processed.  
2034 25 of 2717 files processed.  
2035 17 of 2717 files processed.  
2036 20 of 2717 files processed.  
2037 20 of 2717 files processed.  
2034 26 of 2717 files processed.  
2035 18 of 2717 files processed.  
2035 19 of 2717 files processed.  
2035 20 of 2717 files processed.  
2037 21 of 2717 files processed.  
2036 21 of 2717 files processed.  
2037 22 of 2717 files processed.  
2034 27 of 2717 files processed.  
2036 22 of 2717 files processed.  
2037 23 of 2717 files processed.  
2036 23 of 2717 files processed.  
2037 24 of 2717 files processed.  
2036 24 of 2717 files processed.  
2037 25 of 2717 files processed.  
2035 21 of 2717 files processed.  
2034 28 of 2717 files processed.  
2035 22 of 2717 files processed.  
2034 29 of 2717 files processed.  
2037 26 of 2717 files processed.  
2036 25 of 2717 files processed.  
2034 30 of 2717 files processed.  
2037 27 of 2717 files processed.  
2035 23 of 2717 files processed.  
2034 31 of 2717 files processed.  
2036 26 of 2717 files processed.  
2037 28 of 2717 files processed.  
2034 32 of 2717 files processed.  
2035 24 of 2717 files processed.  
2034 33 of 2717 files processed.  
2037 29 of 2717 files processed.  
2034 34 of 2717 files processed.  
2037 30 of 2717 files processed.  
2036 27 of 2717 files processed.  
2036 28 of 2717 files processed.  
2035 25 of 2717 files processed.  
2037 31 of 2717 files processed.  
2037 32 of 2717 files processed.  
2034 35 of 2717 files processed.  
2034 36 of 2717 files processed.  
2036 29 of 2717 files processed.  
2034 37 of 2717 files processed.  
2035 26 of 2717 files processed.  
2036 30 of 2717 files processed.  
2034 38 of 2717 files processed.  
2035 27 of 2717 files processed.  
2037 33 of 2717 files processed.  
2036 31 of 2717 files processed.  
2037 34 of 2717 files processed.  
2036 32 of 2717 files processed.  
2036 33 of 2717 files processed.  
2034 39 of 2717 files processed.  
2034 40 of 2717 files processed.  
2037 35 of 2717 files processed.  
2034 41 of 2717 files processed.  
2034 42 of 2717 files processed.  
2036 34 of 2717 files processed.  
2036 35 of 2717 files processed.  
2035 28 of 2717 files processed.  
2035 29 of 2717 files processed.  
2037 36 of 2717 files processed.

2034 43 of 2717 files processed.  
2036 36 of 2717 files processed.  
2036 37 of 2717 files processed.  
2035 30 of 2717 files processed.  
2037 37 of 2717 files processed.  
2034 44 of 2717 files processed.  
2035 31 of 2717 files processed.  
2037 38 of 2717 files processed.  
2035 32 of 2717 files processed.  
2036 38 of 2717 files processed.  
2037 39 of 2717 files processed.  
2035 33 of 2717 files processed.  
2036 39 of 2717 files processed.  
2035 34 of 2717 files processed.  
2036 40 of 2717 files processed.  
2037 40 of 2717 files processed.  
2037 41 of 2717 files processed.  
2036 41 of 2717 files processed.  
2037 42 of 2717 files processed.  
2034 45 of 2717 files processed.  
2034 46 of 2717 files processed.  
2035 35 of 2717 files processed.  
2037 43 of 2717 files processed.  
2035 36 of 2717 files processed.  
2035 37 of 2717 files processed.  
2035 38 of 2717 files processed.  
2036 42 of 2717 files processed.  
2034 47 of 2717 files processed.  
2036 43 of 2717 files processed.  
2037 44 of 2717 files processed.  
2036 44 of 2717 files processed.  
2036 45 of 2717 files processed.  
2035 39 of 2717 files processed.  
2036 46 of 2717 files processed.  
2037 45 of 2717 files processed.  
2035 40 of 2717 files processed.  
2037 46 of 2717 files processed.  
2037 47 of 2717 files processed.  
2034 48 of 2717 files processed.  
2036 47 of 2717 files processed.  
2036 48 of 2717 files processed.  
2035 41 of 2717 files processed.  
2034 49 of 2717 files processed.  
2037 48 of 2717 files processed.  
2035 42 of 2717 files processed.  
2036 49 of 2717 files processed.  
2034 50 of 2717 files processed.  
2036 50 of 2717 files processed.  
2037 49 of 2717 files processed.  
2036 51 of 2717 files processed.  
2035 43 of 2717 files processed.  
2035 44 of 2717 files processed.  
2034 51 of 2717 files processed.  
2036 52 of 2717 files processed.  
2034 52 of 2717 files processed.  
2034 53 of 2717 files processed.  
2037 50 of 2717 files processed.  
2035 45 of 2717 files processed.  
2037 51 of 2717 files processed.  
2036 53 of 2717 files processed.  
2036 54 of 2717 files processed.  
2035 46 of 2717 files processed.  
2034 54 of 2717 files processed.  
2036 55 of 2717 files processed.  
2037 52 of 2717 files processed.  
2035 47 of 2717 files processed.  
2034 55 of 2717 files processed.  
2037 53 of 2717 files processed.  
2035 48 of 2717 files processed.  
2036 56 of 2717 files processed.  
2034 56 of 2717 files processed.  
2036 57 of 2717 files processed.  
2037 54 of 2717 files processed.  
2037 55 of 2717 files processed.  
2035 49 of 2717 files processed.  
2034 57 of 2717 files processed.  
2036 58 of 2717 files processed.  
-----

2037 56 of 2717 files processed.  
2035 50 of 2717 files processed.  
2036 59 of 2717 files processed.  
2036 60 of 2717 files processed.  
2037 57 of 2717 files processed.  
2034 58 of 2717 files processed.  
2035 51 of 2717 files processed.  
2036 61 of 2717 files processed.  
2037 58 of 2717 files processed.  
2034 59 of 2717 files processed.  
2037 59 of 2717 files processed.  
2035 52 of 2717 files processed.  
2036 62 of 2717 files processed.  
2034 60 of 2717 files processed.  
2037 60 of 2717 files processed.  
2036 63 of 2717 files processed.  
2034 61 of 2717 files processed.  
2037 61 of 2717 files processed.  
2036 64 of 2717 files processed.  
2035 53 of 2717 files processed.  
2034 62 of 2717 files processed.  
2035 54 of 2717 files processed.  
2037 62 of 2717 files processed.  
2037 63 of 2717 files processed.  
2036 65 of 2717 files processed.  
2034 63 of 2717 files processed.  
2037 64 of 2717 files processed.  
2034 64 of 2717 files processed.  
2037 65 of 2717 files processed.  
2036 66 of 2717 files processed.  
2035 55 of 2717 files processed.  
2035 56 of 2717 files processed.  
2034 65 of 2717 files processed.  
2037 66 of 2717 files processed.  
2037 67 of 2717 files processed.  
2036 67 of 2717 files processed.  
2037 68 of 2717 files processed.  
2037 69 of 2717 files processed.  
2035 57 of 2717 files processed.  
2034 66 of 2717 files processed.  
2036 68 of 2717 files processed.  
2036 69 of 2717 files processed.  
2036 70 of 2717 files processed.  
2034 67 of 2717 files processed.  
2035 58 of 2717 files processed.  
2035 59 of 2717 files processed.  
2034 68 of 2717 files processed.  
2035 60 of 2717 files processed.  
2035 61 of 2717 files processed.  
2037 70 of 2717 files processed.  
2036 71 of 2717 files processed.  
2036 72 of 2717 files processed.  
2035 62 of 2717 files processed.  
2036 73 of 2717 files processed.  
2034 69 of 2717 files processed.  
2035 63 of 2717 files processed.  
2037 71 of 2717 files processed.  
2036 74 of 2717 files processed.  
2035 64 of 2717 files processed.  
2036 75 of 2717 files processed.  
2037 72 of 2717 files processed.  
2036 76 of 2717 files processed.  
2037 73 of 2717 files processed.  
2036 77 of 2717 files processed.  
2034 70 of 2717 files processed.  
2036 78 of 2717 files processed.  
2035 65 of 2717 files processed.  
2037 74 of 2717 files processed.  
2034 71 of 2717 files processed.  
2037 75 of 2717 files processed.  
2036 79 of 2717 files processed.  
2034 72 of 2717 files processed.  
2036 80 of 2717 files processed.  
2036 81 of 2717 files processed.  
2036 82 of 2717 files processed.  
2037 76 of 2717 files processed.  
2037 77 of 2717 files processed.  
-----

2035 66 of 2717 files processed.  
2035 67 of 2717 files processed.  
2034 73 of 2717 files processed.  
2037 78 of 2717 files processed.  
2034 74 of 2717 files processed.  
2035 68 of 2717 files processed.  
2035 69 of 2717 files processed.  
2034 75 of 2717 files processed.  
2035 70 of 2717 files processed.  
2036 83 of 2717 files processed.  
2037 79 of 2717 files processed.  
2037 80 of 2717 files processed.  
2035 71 of 2717 files processed.  
2036 84 of 2717 files processed.  
2034 76 of 2717 files processed.  
2037 81 of 2717 files processed.  
2035 72 of 2717 files processed.  
2034 77 of 2717 files processed.  
2036 85 of 2717 files processed.  
2035 73 of 2717 files processed.  
2037 82 of 2717 files processed.  
2035 74 of 2717 files processed.  
2035 75 of 2717 files processed.  
2036 86 of 2717 files processed.  
2037 83 of 2717 files processed.  
2035 76 of 2717 files processed.  
2036 87 of 2717 files processed.  
2034 78 of 2717 files processed.  
2037 84 of 2717 files processed.  
2037 85 of 2717 files processed.  
2035 77 of 2717 files processed.  
2036 88 of 2717 files processed.  
2035 78 of 2717 files processed.  
2036 89 of 2717 files processed.  
2037 86 of 2717 files processed.  
2034 79 of 2717 files processed.  
2034 80 of 2717 files processed.  
2034 81 of 2717 files processed.  
2037 87 of 2717 files processed.  
2034 82 of 2717 files processed.  
2035 79 of 2717 files processed.  
2034 83 of 2717 files processed.  
2036 90 of 2717 files processed.  
2035 80 of 2717 files processed.  
2034 84 of 2717 files processed.  
2035 81 of 2717 files processed.  
2034 85 of 2717 files processed.  
2037 88 of 2717 files processed.  
2035 82 of 2717 files processed.  
2034 86 of 2717 files processed.  
2036 91 of 2717 files processed.  
2036 92 of 2717 files processed.  
2035 83 of 2717 files processed.  
2037 89 of 2717 files processed.  
2036 93 of 2717 files processed.  
2034 87 of 2717 files processed.  
2034 88 of 2717 files processed.  
2035 84 of 2717 files processed.  
2036 94 of 2717 files processed.  
2035 85 of 2717 files processed.  
2034 89 of 2717 files processed.  
2034 90 of 2717 files processed.  
2035 86 of 2717 files processed.  
2036 95 of 2717 files processed.  
2037 90 of 2717 files processed.  
2034 91 of 2717 files processed.  
2036 96 of 2717 files processed.  
2037 91 of 2717 files processed.  
2035 87 of 2717 files processed.  
2036 97 of 2717 files processed.  
2035 88 of 2717 files processed.  
2034 92 of 2717 files processed.  
2035 89 of 2717 files processed.  
2036 98 of 2717 files processed.  
2034 93 of 2717 files processed.  
2037 92 of 2717 files processed.  
2035 90 of 2717 files processed.

2034 94 of 2717 files processed.  
2034 95 of 2717 files processed.  
2036 99 of 2717 files processed.  
2035 91 of 2717 files processed.  
2037 93 of 2717 files processed.  
2034 96 of 2717 files processed.  
2036 100 of 2717 files processed.  
2034 97 of 2717 files processed.  
2034 98 of 2717 files processed.  
2037 94 of 2717 files processed.  
2035 92 of 2717 files processed.  
2037 95 of 2717 files processed.  
2037 96 of 2717 files processed.  
2034 99 of 2717 files processed.  
2036 101 of 2717 files processed.  
2034 100 of 2717 files processed.  
2036 102 of 2717 files processed.  
2034 101 of 2717 files processed.  
2037 97 of 2717 files processed.  
2035 93 of 2717 files processed.  
2034 102 of 2717 files processed.  
2034 103 of 2717 files processed.  
2035 94 of 2717 files processed.  
2034 104 of 2717 files processed.  
2037 98 of 2717 files processed.  
2036 103 of 2717 files processed.  
2037 99 of 2717 files processed.  
2034 105 of 2717 files processed.  
2035 95 of 2717 files processed.  
2035 96 of 2717 files processed.  
2034 106 of 2717 files processed.  
2036 104 of 2717 files processed.  
2037 100 of 2717 files processed.  
2036 105 of 2717 files processed.  
2034 107 of 2717 files processed.  
2036 106 of 2717 files processed.  
2037 101 of 2717 files processed.  
2036 107 of 2717 files processed.  
2037 102 of 2717 files processed.  
2036 108 of 2717 files processed.  
2034 108 of 2717 files processed.  
2035 97 of 2717 files processed.  
2037 103 of 2717 files processed.  
2036 109 of 2717 files processed.  
2034 109 of 2717 files processed.  
2034 110 of 2717 files processed.  
2035 98 of 2717 files processed.  
2034 111 of 2717 files processed.  
2037 104 of 2717 files processed.  
2036 110 of 2717 files processed.  
2035 99 of 2717 files processed.  
2034 112 of 2717 files processed.  
2036 111 of 2717 files processed.  
2034 113 of 2717 files processed.  
2035 100 of 2717 files processed.  
2037 105 of 2717 files processed.  
2035 101 of 2717 files processed.  
2034 114 of 2717 files processed.  
2035 102 of 2717 files processed.  
2036 112 of 2717 files processed.  
2035 103 of 2717 files processed.  
2037 106 of 2717 files processed.  
2034 115 of 2717 files processed.  
2034 116 of 2717 files processed.  
2035 104 of 2717 files processed.  
2037 107 of 2717 files processed.  
2035 105 of 2717 files processed.  
2036 113 of 2717 files processed.  
2037 108 of 2717 files processed.  
2035 106 of 2717 files processed.  
2036 114 of 2717 files processed.  
2034 117 of 2717 files processed.  
2037 109 of 2717 files processed.  
2036 115 of 2717 files processed.  
2034 118 of 2717 files processed.  
2035 107 of 2717 files processed.  
2034 119 of 2717 files processed.

2035 108 of 2717 files processed.  
2037 110 of 2717 files processed.  
2036 116 of 2717 files processed.  
2035 109 of 2717 files processed.  
2036 117 of 2717 files processed.  
2037 111 of 2717 files processed.  
2034 120 of 2717 files processed.  
2037 112 of 2717 files processed.  
2036 118 of 2717 files processed.  
2034 121 of 2717 files processed.  
2037 113 of 2717 files processed.  
2035 110 of 2717 files processed.  
2036 119 of 2717 files processed.  
2037 114 of 2717 files processed.  
2036 120 of 2717 files processed.  
2037 115 of 2717 files processed.  
2036 121 of 2717 files processed.  
2034 122 of 2717 files processed.  
2035 111 of 2717 files processed.  
2034 123 of 2717 files processed.  
2035 112 of 2717 files processed.  
2036 122 of 2717 files processed.  
2037 116 of 2717 files processed.  
2034 124 of 2717 files processed.  
2035 113 of 2717 files processed.  
2037 117 of 2717 files processed.  
2036 123 of 2717 files processed.  
2034 125 of 2717 files processed.  
2036 124 of 2717 files processed.  
2037 118 of 2717 files processed.  
2035 114 of 2717 files processed.  
2034 126 of 2717 files processed.  
2036 125 of 2717 files processed.  
2035 115 of 2717 files processed.  
2034 127 of 2717 files processed.  
2035 116 of 2717 files processed.  
2037 119 of 2717 files processed.  
2034 128 of 2717 files processed.  
2035 117 of 2717 files processed.  
2035 118 of 2717 files processed.  
2036 126 of 2717 files processed.  
2037 120 of 2717 files processed.  
2034 129 of 2717 files processed.  
2037 121 of 2717 files processed.  
2034 130 of 2717 files processed.  
2037 122 of 2717 files processed.  
2036 127 of 2717 files processed.  
2036 128 of 2717 files processed.  
2035 119 of 2717 files processed.  
2037 123 of 2717 files processed.  
2036 129 of 2717 files processed.  
2034 131 of 2717 files processed.  
2035 120 of 2717 files processed.  
2034 132 of 2717 files processed.  
2035 121 of 2717 files processed.  
2034 133 of 2717 files processed.  
2037 124 of 2717 files processed.  
2035 122 of 2717 files processed.  
2036 130 of 2717 files processed.  
2035 123 of 2717 files processed.  
2034 134 of 2717 files processed.  
2037 125 of 2717 files processed.  
2036 131 of 2717 files processed.  
2035 124 of 2717 files processed.  
2035 125 of 2717 files processed.  
2037 126 of 2717 files processed.  
2036 132 of 2717 files processed.  
2037 127 of 2717 files processed.  
2035 126 of 2717 files processed.  
2035 127 of 2717 files processed.  
2034 135 of 2717 files processed.  
2036 133 of 2717 files processed.  
2035 128 of 2717 files processed.  
2037 128 of 2717 files processed.  
2037 129 of 2717 files processed.  
2036 134 of 2717 files processed.  
2034 136 of 2717 files processed.

2037 130 of 2717 files processed.  
2035 129 of 2717 files processed.  
2036 135 of 2717 files processed.  
2034 137 of 2717 files processed.  
2037 131 of 2717 files processed.  
2035 130 of 2717 files processed.  
2034 138 of 2717 files processed.  
2037 132 of 2717 files processed.  
2034 139 of 2717 files processed.  
2036 136 of 2717 files processed.  
2034 140 of 2717 files processed.  
2037 133 of 2717 files processed.  
2035 131 of 2717 files processed.  
2035 132 of 2717 files processed.  
2035 133 of 2717 files processed.  
2035 134 of 2717 files processed.  
2034 141 of 2717 files processed.  
2036 137 of 2717 files processed.  
2037 134 of 2717 files processed.  
2036 138 of 2717 files processed.  
2035 135 of 2717 files processed.  
2036 139 of 2717 files processed.  
2036 140 of 2717 files processed.  
2034 142 of 2717 files processed.  
2034 143 of 2717 files processed.  
2035 136 of 2717 files processed.  
2037 135 of 2717 files processed.  
2036 141 of 2717 files processed.  
2035 137 of 2717 files processed.  
2034 144 of 2717 files processed.  
2036 142 of 2717 files processed.  
2035 138 of 2717 files processed.  
2035 139 of 2717 files processed.  
2037 136 of 2717 files processed.  
2036 143 of 2717 files processed.  
2037 137 of 2717 files processed.  
2034 145 of 2717 files processed.  
2037 138 of 2717 files processed.  
2035 140 of 2717 files processed.  
2035 141 of 2717 files processed.  
2037 139 of 2717 files processed.  
2036 144 of 2717 files processed.  
2035 142 of 2717 files processed.  
2034 146 of 2717 files processed.  
2035 143 of 2717 files processed.  
2036 145 of 2717 files processed.  
2037 140 of 2717 files processed.  
2036 146 of 2717 files processed.  
2037 141 of 2717 files processed.  
2036 147 of 2717 files processed.  
2034 147 of 2717 files processed.  
2035 144 of 2717 files processed.  
2036 148 of 2717 files processed.  
2037 142 of 2717 files processed.  
2036 149 of 2717 files processed.  
2034 148 of 2717 files processed.  
2035 145 of 2717 files processed.  
2034 149 of 2717 files processed.  
2035 146 of 2717 files processed.  
2036 150 of 2717 files processed.  
2035 147 of 2717 files processed.  
2037 143 of 2717 files processed.  
2034 150 of 2717 files processed.  
2035 148 of 2717 files processed.  
2037 144 of 2717 files processed.  
2036 151 of 2717 files processed.  
2037 145 of 2717 files processed.  
2034 151 of 2717 files processed.  
2035 149 of 2717 files processed.  
2037 146 of 2717 files processed.  
2034 152 of 2717 files processed.  
2036 152 of 2717 files processed.  
2034 153 of 2717 files processed.  
2034 154 of 2717 files processed.  
2037 147 of 2717 files processed.  
2036 153 of 2717 files processed.  
2034 155 of 2717 files processed.

2035 150 of 2717 files processed.  
2034 156 of 2717 files processed.  
2035 151 of 2717 files processed.  
2037 148 of 2717 files processed.  
2035 152 of 2717 files processed.  
2037 149 of 2717 files processed.  
2035 153 of 2717 files processed.  
2036 154 of 2717 files processed.  
2035 154 of 2717 files processed.  
2034 157 of 2717 files processed.  
2036 155 of 2717 files processed.  
2037 150 of 2717 files processed.  
2035 155 of 2717 files processed.  
2034 158 of 2717 files processed.  
2036 156 of 2717 files processed.  
2037 151 of 2717 files processed.  
2034 159 of 2717 files processed.  
2034 160 of 2717 files processed.  
2037 152 of 2717 files processed.  
2035 156 of 2717 files processed.  
2036 157 of 2717 files processed.  
2034 161 of 2717 files processed.  
2036 158 of 2717 files processed.  
2034 162 of 2717 files processed.  
2037 153 of 2717 files processed.  
2037 154 of 2717 files processed.  
2035 157 of 2717 files processed.  
2034 163 of 2717 files processed.  
2036 159 of 2717 files processed.  
2035 158 of 2717 files processed.  
2035 159 of 2717 files processed.  
2034 164 of 2717 files processed.  
2037 155 of 2717 files processed.  
2036 160 of 2717 files processed.  
2036 161 of 2717 files processed.  
2034 165 of 2717 files processed.  
2037 156 of 2717 files processed.  
2037 157 of 2717 files processed.  
2034 166 of 2717 files processed.  
2035 160 of 2717 files processed.  
2037 158 of 2717 files processed.  
2037 159 of 2717 files processed.  
2036 162 of 2717 files processed.  
2036 163 of 2717 files processed.  
2037 160 of 2717 files processed.  
2036 164 of 2717 files processed.  
2034 167 of 2717 files processed.  
2035 161 of 2717 files processed.  
2036 165 of 2717 files processed.  
2035 162 of 2717 files processed.  
2036 166 of 2717 files processed.  
2034 168 of 2717 files processed.  
2036 167 of 2717 files processed.  
2037 161 of 2717 files processed.  
2034 169 of 2717 files processed.  
2036 168 of 2717 files processed.  
2035 163 of 2717 files processed.  
2037 162 of 2717 files processed.  
2034 170 of 2717 files processed.  
2037 163 of 2717 files processed.  
2035 164 of 2717 files processed.  
2035 165 of 2717 files processed.  
2034 171 of 2717 files processed.  
2036 169 of 2717 files processed.  
2037 164 of 2717 files processed.  
2036 170 of 2717 files processed.  
2037 165 of 2717 files processed.  
2035 166 of 2717 files processed.  
2037 166 of 2717 files processed.  
2034 172 of 2717 files processed.  
2036 171 of 2717 files processed.  
2037 167 of 2717 files processed.  
2034 173 of 2717 files processed.  
2035 167 of 2717 files processed.  
2037 168 of 2717 files processed.  
2035 168 of 2717 files processed.  
2037 169 of 2717 files processed.

2034 174 of 2717 files processed.  
2035 169 of 2717 files processed.  
2035 170 of 2717 files processed.  
2036 172 of 2717 files processed.  
2034 175 of 2717 files processed.  
2037 170 of 2717 files processed.  
2035 171 of 2717 files processed.  
2036 173 of 2717 files processed.  
2034 176 of 2717 files processed.  
2037 171 of 2717 files processed.  
2034 177 of 2717 files processed.  
2037 172 of 2717 files processed.  
2036 174 of 2717 files processed.  
2035 172 of 2717 files processed.  
2034 178 of 2717 files processed.  
2035 173 of 2717 files processed.  
2037 173 of 2717 files processed.  
2034 179 of 2717 files processed.  
2036 175 of 2717 files processed.  
2035 174 of 2717 files processed.  
2035 175 of 2717 files processed.  
2036 176 of 2717 files processed.  
2034 180 of 2717 files processed.  
2037 174 of 2717 files processed.  
2035 176 of 2717 files processed.  
2036 177 of 2717 files processed.  
2035 177 of 2717 files processed.  
2036 178 of 2717 files processed.  
2035 178 of 2717 files processed.  
2037 175 of 2717 files processed.  
2034 181 of 2717 files processed.  
2037 176 of 2717 files processed.  
2035 179 of 2717 files processed.  
2036 179 of 2717 files processed.  
2034 182 of 2717 files processed.  
2037 177 of 2717 files processed.  
2035 180 of 2717 files processed.  
2037 178 of 2717 files processed.  
2035 181 of 2717 files processed.  
2034 183 of 2717 files processed.  
2035 182 of 2717 files processed.  
2036 180 of 2717 files processed.  
2034 184 of 2717 files processed.  
2035 183 of 2717 files processed.  
2034 185 of 2717 files processed.  
2037 179 of 2717 files processed.  
2034 186 of 2717 files processed.  
2036 181 of 2717 files processed.  
2037 180 of 2717 files processed.  
2034 187 of 2717 files processed.  
2037 181 of 2717 files processed.  
2034 188 of 2717 files processed.  
2035 184 of 2717 files processed.  
2036 182 of 2717 files processed.  
2037 182 of 2717 files processed.  
2035 185 of 2717 files processed.  
2034 189 of 2717 files processed.  
2035 186 of 2717 files processed.  
2035 187 of 2717 files processed.  
2037 183 of 2717 files processed.  
2034 190 of 2717 files processed.  
2036 183 of 2717 files processed.  
2037 184 of 2717 files processed.  
2034 191 of 2717 files processed.  
2035 188 of 2717 files processed.  
2037 185 of 2717 files processed.  
2036 184 of 2717 files processed.  
2034 192 of 2717 files processed.  
2035 189 of 2717 files processed.  
2036 185 of 2717 files processed.  
2034 193 of 2717 files processed.  
2037 186 of 2717 files processed.  
2037 187 of 2717 files processed.  
2035 190 of 2717 files processed.  
2036 186 of 2717 files processed.  
2037 188 of 2717 files processed.  
2035 191 of 2717 files processed.

2034 194 of 2717 files processed.  
2036 187 of 2717 files processed.  
2037 189 of 2717 files processed.  
2034 195 of 2717 files processed.  
2037 190 of 2717 files processed.  
2036 188 of 2717 files processed.  
2035 192 of 2717 files processed.  
2034 196 of 2717 files processed.  
2036 189 of 2717 files processed.  
2035 193 of 2717 files processed.  
2037 191 of 2717 files processed.  
2035 194 of 2717 files processed.  
2034 197 of 2717 files processed.  
2037 192 of 2717 files processed.  
2036 190 of 2717 files processed.  
2035 195 of 2717 files processed.  
2034 198 of 2717 files processed.  
2035 196 of 2717 files processed.  
2037 193 of 2717 files processed.  
2036 191 of 2717 files processed.  
2037 194 of 2717 files processed.  
2034 199 of 2717 files processed.  
2036 192 of 2717 files processed.  
2035 197 of 2717 files processed.  
2037 195 of 2717 files processed.  
2034 200 of 2717 files processed.  
2034 201 of 2717 files processed.  
2035 198 of 2717 files processed.  
2036 193 of 2717 files processed.  
2037 196 of 2717 files processed.  
2034 202 of 2717 files processed.  
2035 199 of 2717 files processed.  
2037 197 of 2717 files processed.  
2037 198 of 2717 files processed.  
2035 200 of 2717 files processed.  
2036 194 of 2717 files processed.  
2034 203 of 2717 files processed.  
2035 201 of 2717 files processed.  
2036 195 of 2717 files processed.  
2034 204 of 2717 files processed.  
2037 199 of 2717 files processed.  
2035 202 of 2717 files processed.  
2037 200 of 2717 files processed.  
2036 196 of 2717 files processed.  
2034 205 of 2717 files processed.  
2035 203 of 2717 files processed.  
2034 206 of 2717 files processed.  
2035 204 of 2717 files processed.  
2037 201 of 2717 files processed.  
2036 197 of 2717 files processed.  
2035 205 of 2717 files processed.  
2034 207 of 2717 files processed.  
2037 202 of 2717 files processed.  
2035 206 of 2717 files processed.  
2034 208 of 2717 files processed.  
2036 198 of 2717 files processed.  
2037 203 of 2717 files processed.  
2035 207 of 2717 files processed.  
2034 209 of 2717 files processed.  
2036 199 of 2717 files processed.  
2037 204 of 2717 files processed.  
2035 208 of 2717 files processed.  
2036 200 of 2717 files processed.  
2034 210 of 2717 files processed.  
2035 209 of 2717 files processed.  
2034 211 of 2717 files processed.  
2036 201 of 2717 files processed.  
2037 205 of 2717 files processed.  
2034 212 of 2717 files processed.  
2035 210 of 2717 files processed.  
2036 202 of 2717 files processed.  
2037 206 of 2717 files processed.  
2035 211 of 2717 files processed.  
2037 207 of 2717 files processed.  
2036 203 of 2717 files processed.  
2034 213 of 2717 files processed.  
2036 204 of 2717 files processed.

2035 212 of 2717 files processed.  
2037 208 of 2717 files processed.  
2035 213 of 2717 files processed.  
2034 214 of 2717 files processed.  
2036 205 of 2717 files processed.  
2037 209 of 2717 files processed.  
2035 214 of 2717 files processed.  
2036 206 of 2717 files processed.  
2034 215 of 2717 files processed.  
2037 210 of 2717 files processed.  
2035 215 of 2717 files processed.  
2034 216 of 2717 files processed.  
2036 207 of 2717 files processed.  
2035 216 of 2717 files processed.  
2037 211 of 2717 files processed.  
2034 217 of 2717 files processed.  
2035 217 of 2717 files processed.  
2036 208 of 2717 files processed.  
2034 218 of 2717 files processed.  
2037 212 of 2717 files processed.  
2034 219 of 2717 files processed.  
2035 218 of 2717 files processed.  
2035 219 of 2717 files processed.  
2036 209 of 2717 files processed.  
2037 213 of 2717 files processed.  
2034 220 of 2717 files processed.  
2034 221 of 2717 files processed.  
2037 214 of 2717 files processed.  
2036 210 of 2717 files processed.  
2035 220 of 2717 files processed.  
2037 215 of 2717 files processed.  
2034 222 of 2717 files processed.  
2035 221 of 2717 files processed.  
2036 211 of 2717 files processed.  
2037 216 of 2717 files processed.  
2036 212 of 2717 files processed.  
2034 223 of 2717 files processed.  
2035 222 of 2717 files processed.  
2036 213 of 2717 files processed.  
2036 214 of 2717 files processed.  
2037 217 of 2717 files processed.  
2035 223 of 2717 files processed.  
2034 224 of 2717 files processed.  
2037 218 of 2717 files processed.  
2036 215 of 2717 files processed.  
2034 225 of 2717 files processed.  
2037 219 of 2717 files processed.  
2036 216 of 2717 files processed.  
2035 224 of 2717 files processed.  
2034 226 of 2717 files processed.  
2037 220 of 2717 files processed.  
2036 217 of 2717 files processed.  
2034 227 of 2717 files processed.  
2035 225 of 2717 files processed.  
2037 221 of 2717 files processed.  
2036 218 of 2717 files processed.  
2035 226 of 2717 files processed.  
2034 228 of 2717 files processed.  
2035 227 of 2717 files processed.  
2035 228 of 2717 files processed.  
2037 222 of 2717 files processed.  
2036 219 of 2717 files processed.  
2037 223 of 2717 files processed.  
2034 229 of 2717 files processed.  
2035 229 of 2717 files processed.  
2034 230 of 2717 files processed.  
2037 224 of 2717 files processed.  
2034 231 of 2717 files processed.  
2037 225 of 2717 files processed.  
2035 230 of 2717 files processed.  
2036 220 of 2717 files processed.  
2037 226 of 2717 files processed.  
2035 231 of 2717 files processed.  
2037 227 of 2717 files processed.  
2034 232 of 2717 files processed.  
2036 221 of 2717 files processed.  
2035 232 of 2717 files processed.

2034 233 of 2717 files processed.  
2035 233 of 2717 files processed.  
2037 228 of 2717 files processed.  
2034 234 of 2717 files processed.  
2035 234 of 2717 files processed.  
2037 229 of 2717 files processed.  
2036 222 of 2717 files processed.  
2037 230 of 2717 files processed.  
2034 235 of 2717 files processed.  
2035 235 of 2717 files processed.  
2034 236 of 2717 files processed.  
2036 223 of 2717 files processed.  
2037 231 of 2717 files processed.  
2036 224 of 2717 files processed.  
2034 237 of 2717 files processed.  
2035 236 of 2717 files processed.  
2036 225 of 2717 files processed.  
2037 232 of 2717 files processed.  
2036 226 of 2717 files processed.  
2035 237 of 2717 files processed.  
2037 233 of 2717 files processed.  
2035 238 of 2717 files processed.  
2037 234 of 2717 files processed.  
2036 227 of 2717 files processed.  
2034 238 of 2717 files processed.  
2034 239 of 2717 files processed.  
2037 235 of 2717 files processed.  
2035 239 of 2717 files processed.  
2034 240 of 2717 files processed.  
2036 228 of 2717 files processed.  
2035 240 of 2717 files processed.  
2034 241 of 2717 files processed.  
2036 229 of 2717 files processed.  
2037 236 of 2717 files processed.  
2035 241 of 2717 files processed.  
2034 242 of 2717 files processed.  
2035 242 of 2717 files processed.  
2036 230 of 2717 files processed.  
2034 243 of 2717 files processed.  
2037 237 of 2717 files processed.  
2035 243 of 2717 files processed.  
2035 244 of 2717 files processed.  
2036 231 of 2717 files processed.  
2034 244 of 2717 files processed.  
2037 238 of 2717 files processed.  
2036 232 of 2717 files processed.  
2035 245 of 2717 files processed.  
2035 246 of 2717 files processed.  
2037 239 of 2717 files processed.  
2036 233 of 2717 files processed.  
2034 245 of 2717 files processed.  
2035 247 of 2717 files processed.  
2037 240 of 2717 files processed.  
2034 246 of 2717 files processed.  
2036 234 of 2717 files processed.  
2035 248 of 2717 files processed.  
2037 241 of 2717 files processed.  
2034 247 of 2717 files processed.  
2036 235 of 2717 files processed.  
2037 242 of 2717 files processed.  
2035 249 of 2717 files processed.  
2036 236 of 2717 files processed.  
2035 250 of 2717 files processed.  
2037 243 of 2717 files processed.  
2036 237 of 2717 files processed.  
2034 248 of 2717 files processed.  
2035 251 of 2717 files processed.  
2034 249 of 2717 files processed.  
2036 238 of 2717 files processed.  
2037 244 of 2717 files processed.  
2035 252 of 2717 files processed.  
2037 245 of 2717 files processed.  
2034 250 of 2717 files processed.  
2036 239 of 2717 files processed.  
2037 246 of 2717 files processed.  
2034 251 of 2717 files processed.  
2035 253 of 2717 files processed.

----  
2036 240 of 2717 files processed.  
2037 247 of 2717 files processed.  
2034 252 of 2717 files processed.  
2036 241 of 2717 files processed.  
2037 248 of 2717 files processed.  
2035 254 of 2717 files processed.  
2036 242 of 2717 files processed.  
2037 249 of 2717 files processed.  
2035 255 of 2717 files processed.  
2036 243 of 2717 files processed.  
2034 253 of 2717 files processed.  
2035 256 of 2717 files processed.  
2037 250 of 2717 files processed.  
2036 244 of 2717 files processed.  
2034 254 of 2717 files processed.  
2036 245 of 2717 files processed.  
2034 255 of 2717 files processed.  
2037 251 of 2717 files processed.  
2035 257 of 2717 files processed.  
2034 256 of 2717 files processed.  
2036 246 of 2717 files processed.  
2034 257 of 2717 files processed.  
2035 258 of 2717 files processed.  
2037 252 of 2717 files processed.  
2036 247 of 2717 files processed.  
2035 259 of 2717 files processed.  
2036 248 of 2717 files processed.  
2037 253 of 2717 files processed.  
2036 249 of 2717 files processed.  
2037 254 of 2717 files processed.  
2034 258 of 2717 files processed.  
2035 260 of 2717 files processed.  
2036 250 of 2717 files processed.  
2037 255 of 2717 files processed.  
2037 256 of 2717 files processed.  
2035 261 of 2717 files processed.  
2034 259 of 2717 files processed.  
2035 262 of 2717 files processed.  
2036 251 of 2717 files processed.  
2036 252 of 2717 files processed.  
2037 257 of 2717 files processed.  
2034 260 of 2717 files processed.  
2035 263 of 2717 files processed.  
2037 258 of 2717 files processed.  
2036 253 of 2717 files processed.  
2034 261 of 2717 files processed.  
2035 264 of 2717 files processed.  
2037 259 of 2717 files processed.  
2035 265 of 2717 files processed.  
2036 254 of 2717 files processed.  
2034 262 of 2717 files processed.  
2037 260 of 2717 files processed.  
2035 266 of 2717 files processed.  
2037 261 of 2717 files processed.  
2037 262 of 2717 files processed.  
2035 267 of 2717 files processed.  
2034 263 of 2717 files processed.  
2036 255 of 2717 files processed.  
2036 256 of 2717 files processed.  
2037 263 of 2717 files processed.  
2034 264 of 2717 files processed.  
2037 264 of 2717 files processed.  
2035 268 of 2717 files processed.  
2036 257 of 2717 files processed.  
2034 265 of 2717 files processed.  
2034 266 of 2717 files processed.  
2037 265 of 2717 files processed.  
2035 269 of 2717 files processed.  
2036 258 of 2717 files processed.  
2035 270 of 2717 files processed.  
2037 266 of 2717 files processed.  
2034 267 of 2717 files processed.  
2036 259 of 2717 files processed.  
2037 267 of 2717 files processed.  
2035 271 of 2717 files processed.  
2035 272 of 2717 files processed.  
2037 268 of 2717 files processed.

.....  
2034 268 of 2717 files processed.  
2036 260 of 2717 files processed.  
2034 269 of 2717 files processed.  
2035 273 of 2717 files processed.  
2037 269 of 2717 files processed.  
2035 274 of 2717 files processed.  
2035 275 of 2717 files processed.  
2034 270 of 2717 files processed.  
2036 261 of 2717 files processed.  
2035 276 of 2717 files processed.  
2037 270 of 2717 files processed.  
2036 262 of 2717 files processed.  
2037 271 of 2717 files processed.  
2034 271 of 2717 files processed.  
2035 277 of 2717 files processed.  
2037 272 of 2717 files processed.  
2034 272 of 2717 files processed.  
2036 263 of 2717 files processed.  
2035 278 of 2717 files processed.  
2037 273 of 2717 files processed.  
2035 279 of 2717 files processed.  
2034 273 of 2717 files processed.  
2035 280 of 2717 files processed.  
2036 264 of 2717 files processed.  
2036 265 of 2717 files processed.  
2034 274 of 2717 files processed.  
2037 274 of 2717 files processed.  
2035 281 of 2717 files processed.  
2036 266 of 2717 files processed.  
2037 275 of 2717 files processed.  
2034 275 of 2717 files processed.  
2036 267 of 2717 files processed.  
2035 282 of 2717 files processed.  
2034 276 of 2717 files processed.  
2037 276 of 2717 files processed.  
2035 283 of 2717 files processed.  
2036 268 of 2717 files processed.  
2037 277 of 2717 files processed.  
2035 284 of 2717 files processed.  
2036 269 of 2717 files processed.  
2034 277 of 2717 files processed.  
2036 270 of 2717 files processed.  
2035 285 of 2717 files processed.  
2037 278 of 2717 files processed.  
2036 271 of 2717 files processed.  
2034 278 of 2717 files processed.  
2036 272 of 2717 files processed.  
2035 286 of 2717 files processed.  
2037 279 of 2717 files processed.  
2036 273 of 2717 files processed.  
2034 279 of 2717 files processed.  
2035 287 of 2717 files processed.  
2036 274 of 2717 files processed.  
2037 280 of 2717 files processed.  
2035 288 of 2717 files processed.  
2036 275 of 2717 files processed.  
2035 289 of 2717 files processed.  
2037 281 of 2717 files processed.  
2036 276 of 2717 files processed.  
2034 280 of 2717 files processed.  
2036 277 of 2717 files processed.  
2037 282 of 2717 files processed.  
2034 281 of 2717 files processed.  
2035 290 of 2717 files processed.  
2036 278 of 2717 files processed.  
2034 282 of 2717 files processed.  
2036 279 of 2717 files processed.  
2037 283 of 2717 files processed.  
2035 291 of 2717 files processed.  
2034 283 of 2717 files processed.  
2037 284 of 2717 files processed.  
2034 284 of 2717 files processed.  
2034 285 of 2717 files processed.  
2035 292 of 2717 files processed.  
2036 280 of 2717 files processed.  
2037 285 of 2717 files processed.  
2034 286 of 2717 files processed.

2034 200 of 2717 files processed.  
2035 293 of 2717 files processed.  
2036 281 of 2717 files processed.  
2037 286 of 2717 files processed.  
2035 294 of 2717 files processed.  
2037 287 of 2717 files processed.  
2034 287 of 2717 files processed.  
2036 282 of 2717 files processed.  
2037 288 of 2717 files processed.  
2037 289 of 2717 files processed.  
2035 295 of 2717 files processed.  
2034 288 of 2717 files processed.  
2036 283 of 2717 files processed.  
2037 290 of 2717 files processed.  
2035 296 of 2717 files processed.  
2035 297 of 2717 files processed.  
2034 289 of 2717 files processed.  
2037 291 of 2717 files processed.  
2036 284 of 2717 files processed.  
2035 298 of 2717 files processed.  
2036 285 of 2717 files processed.  
2035 299 of 2717 files processed.  
2036 286 of 2717 files processed.  
2034 290 of 2717 files processed.  
2037 292 of 2717 files processed.  
2036 287 of 2717 files processed.  
2034 291 of 2717 files processed.  
2035 300 of 2717 files processed.  
2036 288 of 2717 files processed.  
2037 293 of 2717 files processed.  
2035 301 of 2717 files processed.  
2034 292 of 2717 files processed.  
2037 294 of 2717 files processed.  
2034 293 of 2717 files processed.  
2035 302 of 2717 files processed.  
2036 289 of 2717 files processed.  
2034 294 of 2717 files processed.  
2034 295 of 2717 files processed.  
2036 290 of 2717 files processed.  
2035 303 of 2717 files processed.  
2037 295 of 2717 files processed.  
2035 304 of 2717 files processed.  
2034 296 of 2717 files processed.  
2037 296 of 2717 files processed.  
2034 297 of 2717 files processed.  
2034 298 of 2717 files processed.  
2035 305 of 2717 files processed.  
2036 291 of 2717 files processed.  
2037 297 of 2717 files processed.  
2034 299 of 2717 files processed.  
2035 306 of 2717 files processed.  
2037 298 of 2717 files processed.  
2034 300 of 2717 files processed.  
2035 307 of 2717 files processed.  
2036 292 of 2717 files processed.  
2034 301 of 2717 files processed.  
2036 293 of 2717 files processed.  
2037 299 of 2717 files processed.  
2035 308 of 2717 files processed.  
2034 302 of 2717 files processed.  
2035 309 of 2717 files processed.  
2036 294 of 2717 files processed.  
2037 300 of 2717 files processed.  
2034 303 of 2717 files processed.  
2035 310 of 2717 files processed.  
2036 295 of 2717 files processed.  
2036 296 of 2717 files processed.  
2037 301 of 2717 files processed.  
2035 311 of 2717 files processed.  
2034 304 of 2717 files processed.  
2036 297 of 2717 files processed.  
2035 312 of 2717 files processed.  
2036 298 of 2717 files processed.  
2036 299 of 2717 files processed.  
2037 302 of 2717 files processed.  
2035 313 of 2717 files processed.  
2034 305 of 2717 files processed.  
2037 303 of 2717 files processed

2037 305 of 2717 files processed.  
2034 306 of 2717 files processed.  
2035 314 of 2717 files processed.  
2036 300 of 2717 files processed.  
2037 304 of 2717 files processed.  
2035 315 of 2717 files processed.  
2036 301 of 2717 files processed.  
2034 307 of 2717 files processed.  
2037 305 of 2717 files processed.  
2036 302 of 2717 files processed.  
2034 308 of 2717 files processed.  
2035 316 of 2717 files processed.  
2036 303 of 2717 files processed.  
2037 306 of 2717 files processed.  
2035 317 of 2717 files processed.  
2034 309 of 2717 files processed.  
2035 318 of 2717 files processed.  
2036 304 of 2717 files processed.  
2037 307 of 2717 files processed.  
2034 310 of 2717 files processed.  
2036 305 of 2717 files processed.  
2037 308 of 2717 files processed.  
2035 319 of 2717 files processed.  
2034 311 of 2717 files processed.  
2036 306 of 2717 files processed.  
2037 309 of 2717 files processed.  
2034 312 of 2717 files processed.  
2037 310 of 2717 files processed.  
2035 320 of 2717 files processed.  
2034 313 of 2717 files processed.  
2037 311 of 2717 files processed.  
2036 307 of 2717 files processed.  
2034 314 of 2717 files processed.  
2037 312 of 2717 files processed.  
2036 308 of 2717 files processed.  
2035 321 of 2717 files processed.  
2037 313 of 2717 files processed.  
2036 309 of 2717 files processed.  
2037 314 of 2717 files processed.  
2035 322 of 2717 files processed.  
2034 315 of 2717 files processed.  
2035 323 of 2717 files processed.  
2034 316 of 2717 files processed.  
2037 315 of 2717 files processed.  
2034 317 of 2717 files processed.  
2036 310 of 2717 files processed.  
2037 316 of 2717 files processed.  
2034 318 of 2717 files processed.  
2035 324 of 2717 files processed.  
2034 319 of 2717 files processed.  
2036 311 of 2717 files processed.  
2037 317 of 2717 files processed.  
2035 325 of 2717 files processed.  
2036 312 of 2717 files processed.  
2034 320 of 2717 files processed.  
2036 313 of 2717 files processed.  
2037 318 of 2717 files processed.  
2035 326 of 2717 files processed.  
2036 314 of 2717 files processed.  
2034 321 of 2717 files processed.  
2035 327 of 2717 files processed.  
2034 322 of 2717 files processed.  
2037 319 of 2717 files processed.  
2036 315 of 2717 files processed.  
2034 323 of 2717 files processed.  
2035 328 of 2717 files processed.  
2037 320 of 2717 files processed.  
2034 324 of 2717 files processed.  
2035 329 of 2717 files processed.  
2036 316 of 2717 files processed.  
2037 321 of 2717 files processed.  
2036 317 of 2717 files processed.  
2034 325 of 2717 files processed.  
2035 330 of 2717 files processed.  
2036 318 of 2717 files processed.  
2037 322 of 2717 files processed.  
2036 319 of 2717 files processed.  
2034 326 of 2717 files processed.

2034 320 of 2717 files processed.  
2035 331 of 2717 files processed.  
2037 323 of 2717 files processed.  
2035 332 of 2717 files processed.  
2034 327 of 2717 files processed.  
2037 324 of 2717 files processed.  
2036 320 of 2717 files processed.  
2035 333 of 2717 files processed.  
2034 328 of 2717 files processed.  
2036 321 of 2717 files processed.  
2034 329 of 2717 files processed.  
2037 325 of 2717 files processed.  
2035 334 of 2717 files processed.  
2036 322 of 2717 files processed.  
2037 326 of 2717 files processed.  
2034 330 of 2717 files processed.  
2035 335 of 2717 files processed.  
2037 327 of 2717 files processed.  
2036 323 of 2717 files processed.  
2037 328 of 2717 files processed.  
2035 336 of 2717 files processed.  
2034 331 of 2717 files processed.  
2036 324 of 2717 files processed.  
2037 329 of 2717 files processed.  
2035 337 of 2717 files processed.  
2037 330 of 2717 files processed.  
2034 332 of 2717 files processed.  
2036 325 of 2717 files processed.  
2034 333 of 2717 files processed.  
2037 331 of 2717 files processed.  
2035 338 of 2717 files processed.  
2037 332 of 2717 files processed.  
2036 326 of 2717 files processed.  
2036 327 of 2717 files processed.  
2034 334 of 2717 files processed.  
2035 339 of 2717 files processed.  
2036 328 of 2717 files processed.  
2034 335 of 2717 files processed.  
2035 340 of 2717 files processed.  
2037 333 of 2717 files processed.  
2037 334 of 2717 files processed.  
2036 329 of 2717 files processed.  
2034 336 of 2717 files processed.  
2035 341 of 2717 files processed.  
2037 335 of 2717 files processed.  
2036 330 of 2717 files processed.  
2037 336 of 2717 files processed.  
2034 337 of 2717 files processed.  
2035 342 of 2717 files processed.  
2035 343 of 2717 files processed.  
2036 331 of 2717 files processed.  
2037 337 of 2717 files processed.  
2034 338 of 2717 files processed.  
2037 338 of 2717 files processed.  
2035 344 of 2717 files processed.  
2034 339 of 2717 files processed.  
2036 332 of 2717 files processed.  
2037 339 of 2717 files processed.  
2034 340 of 2717 files processed.  
2036 333 of 2717 files processed.  
2035 345 of 2717 files processed.  
2034 341 of 2717 files processed.  
2036 334 of 2717 files processed.  
2037 340 of 2717 files processed.  
2036 335 of 2717 files processed.  
2037 341 of 2717 files processed.  
2035 346 of 2717 files processed.  
2034 342 of 2717 files processed.  
2034 343 of 2717 files processed.  
2036 336 of 2717 files processed.  
2035 347 of 2717 files processed.  
2034 344 of 2717 files processed.  
2036 337 of 2717 files processed.  
2037 342 of 2717 files processed.  
2035 348 of 2717 files processed.  
2036 338 of 2717 files processed.  
2036 339 of 2717 files processed.

2035 349 of 2717 files processed.  
2034 345 of 2717 files processed.  
2036 340 of 2717 files processed.  
2037 343 of 2717 files processed.  
2035 350 of 2717 files processed.  
2037 344 of 2717 files processed.  
2036 341 of 2717 files processed.  
2034 346 of 2717 files processed.  
2035 351 of 2717 files processed.  
2037 345 of 2717 files processed.  
2036 342 of 2717 files processed.  
2034 347 of 2717 files processed.  
2035 352 of 2717 files processed.  
2037 346 of 2717 files processed.  
2036 343 of 2717 files processed.  
2034 348 of 2717 files processed.  
2035 353 of 2717 files processed.  
2037 347 of 2717 files processed.  
2036 344 of 2717 files processed.  
2037 348 of 2717 files processed.  
2034 349 of 2717 files processed.  
2037 349 of 2717 files processed.  
2035 354 of 2717 files processed.  
2036 345 of 2717 files processed.  
2034 350 of 2717 files processed.  
2037 350 of 2717 files processed.  
2035 355 of 2717 files processed.  
2036 346 of 2717 files processed.  
2037 351 of 2717 files processed.  
2034 351 of 2717 files processed.  
2037 352 of 2717 files processed.  
2036 347 of 2717 files processed.  
2035 356 of 2717 files processed.  
2037 353 of 2717 files processed.  
2035 357 of 2717 files processed.  
2037 354 of 2717 files processed.  
2036 348 of 2717 files processed.  
2034 352 of 2717 files processed.  
2037 355 of 2717 files processed.  
2034 353 of 2717 files processed.  
2035 358 of 2717 files processed.  
2036 349 of 2717 files processed.  
2037 356 of 2717 files processed.  
2036 350 of 2717 files processed.  
2034 354 of 2717 files processed.  
2035 359 of 2717 files processed.  
2037 357 of 2717 files processed.  
2036 351 of 2717 files processed.  
2034 355 of 2717 files processed.  
2037 358 of 2717 files processed.  
2035 360 of 2717 files processed.  
2036 352 of 2717 files processed.  
2034 356 of 2717 files processed.  
2036 353 of 2717 files processed.  
2037 359 of 2717 files processed.  
2034 357 of 2717 files processed.  
2035 361 of 2717 files processed.  
2036 354 of 2717 files processed.  
2037 360 of 2717 files processed.  
2034 358 of 2717 files processed.  
2035 362 of 2717 files processed.  
2037 361 of 2717 files processed.  
2036 355 of 2717 files processed.  
2037 362 of 2717 files processed.  
2034 359 of 2717 files processed.  
2035 363 of 2717 files processed.  
2034 360 of 2717 files processed.  
2035 364 of 2717 files processed.  
2036 356 of 2717 files processed.  
2037 363 of 2717 files processed.  
2034 361 of 2717 files processed.  
2036 357 of 2717 files processed.  
2035 365 of 2717 files processed.  
2037 364 of 2717 files processed.  
2037 365 of 2717 files processed.  
2036 358 of 2717 files processed.  
2034 362 of 2717 files processed.

2036 359 of 2717 files processed.  
2037 366 of 2717 files processed.  
2036 360 of 2717 files processed.  
2037 367 of 2717 files processed.  
2034 363 of 2717 files processed.  
2035 366 of 2717 files processed.  
2034 364 of 2717 files processed.  
2037 368 of 2717 files processed.  
2035 367 of 2717 files processed.  
2036 361 of 2717 files processed.  
2037 369 of 2717 files processed.  
2036 362 of 2717 files processed.  
2035 368 of 2717 files processed.  
2034 365 of 2717 files processed.  
2036 363 of 2717 files processed.  
2037 370 of 2717 files processed.  
2035 369 of 2717 files processed.  
2034 366 of 2717 files processed.  
2036 364 of 2717 files processed.  
2037 371 of 2717 files processed.  
2035 370 of 2717 files processed.  
2037 372 of 2717 files processed.  
2036 365 of 2717 files processed.  
2035 371 of 2717 files processed.  
2034 367 of 2717 files processed.  
2036 366 of 2717 files processed.  
2037 373 of 2717 files processed.  
2035 372 of 2717 files processed.  
2034 368 of 2717 files processed.  
2036 367 of 2717 files processed.  
2037 374 of 2717 files processed.  
2036 368 of 2717 files processed.  
2035 373 of 2717 files processed.  
2034 369 of 2717 files processed.  
2037 375 of 2717 files processed.  
2035 374 of 2717 files processed.  
2037 376 of 2717 files processed.  
2036 369 of 2717 files processed.  
2035 375 of 2717 files processed.  
2037 377 of 2717 files processed.  
2036 370 of 2717 files processed.  
2035 376 of 2717 files processed.  
2034 370 of 2717 files processed.  
2037 378 of 2717 files processed.  
2035 377 of 2717 files processed.  
2034 371 of 2717 files processed.  
2036 371 of 2717 files processed.  
2037 379 of 2717 files processed.  
2034 372 of 2717 files processed.  
2035 378 of 2717 files processed.  
2034 373 of 2717 files processed.  
2036 372 of 2717 files processed.  
2035 379 of 2717 files processed.  
2034 374 of 2717 files processed.  
2037 380 of 2717 files processed.  
2036 373 of 2717 files processed.  
2034 375 of 2717 files processed.  
2036 374 of 2717 files processed.  
2035 380 of 2717 files processed.  
2034 376 of 2717 files processed.  
2037 381 of 2717 files processed.  
2035 381 of 2717 files processed.  
2036 375 of 2717 files processed.  
2034 377 of 2717 files processed.  
2035 382 of 2717 files processed.  
2036 376 of 2717 files processed.  
2037 382 of 2717 files processed.  
2035 383 of 2717 files processed.  
2034 378 of 2717 files processed.  
2037 383 of 2717 files processed.  
2036 377 of 2717 files processed.  
2034 379 of 2717 files processed.  
2035 384 of 2717 files processed.  
2036 378 of 2717 files processed.  
2034 380 of 2717 files processed.  
2035 385 of 2717 files processed.  
2037 384 of 2717 files processed.  
-----

2034 381 of 2717 files processed.  
2035 386 of 2717 files processed.  
2036 379 of 2717 files processed.  
2034 382 of 2717 files processed.  
2037 385 of 2717 files processed.  
2035 387 of 2717 files processed.  
2036 380 of 2717 files processed.  
2037 386 of 2717 files processed.  
2034 383 of 2717 files processed.  
2035 388 of 2717 files processed.  
2036 381 of 2717 files processed.  
2034 384 of 2717 files processed.  
2036 382 of 2717 files processed.  
2037 387 of 2717 files processed.  
2034 385 of 2717 files processed.  
2036 383 of 2717 files processed.  
2037 388 of 2717 files processed.  
2035 389 of 2717 files processed.  
2037 389 of 2717 files processed.  
2034 386 of 2717 files processed.  
2036 384 of 2717 files processed.  
2035 390 of 2717 files processed.  
2037 390 of 2717 files processed.  
2034 387 of 2717 files processed.  
2036 385 of 2717 files processed.  
2034 388 of 2717 files processed.  
2036 386 of 2717 files processed.  
2035 391 of 2717 files processed.  
2037 391 of 2717 files processed.  
2034 389 of 2717 files processed.  
2036 387 of 2717 files processed.  
2035 392 of 2717 files processed.  
2037 392 of 2717 files processed.  
2036 388 of 2717 files processed.  
2035 393 of 2717 files processed.  
2034 390 of 2717 files processed.  
2036 389 of 2717 files processed.  
2035 394 of 2717 files processed.  
2037 393 of 2717 files processed.  
2036 390 of 2717 files processed.  
2035 395 of 2717 files processed.  
2034 391 of 2717 files processed.  
2037 394 of 2717 files processed.  
2036 391 of 2717 files processed.  
2035 396 of 2717 files processed.  
2034 392 of 2717 files processed.  
2036 392 of 2717 files processed.  
2037 395 of 2717 files processed.  
2034 393 of 2717 files processed.  
2035 397 of 2717 files processed.  
2036 393 of 2717 files processed.  
2034 394 of 2717 files processed.  
2035 398 of 2717 files processed.  
2037 396 of 2717 files processed.  
2035 399 of 2717 files processed.  
2037 397 of 2717 files processed.  
2034 395 of 2717 files processed.  
2035 400 of 2717 files processed.  
2037 398 of 2717 files processed.  
2036 394 of 2717 files processed.  
2035 401 of 2717 files processed.  
2036 395 of 2717 files processed.  
2037 399 of 2717 files processed.  
2034 396 of 2717 files processed.  
2035 402 of 2717 files processed.  
2037 400 of 2717 files processed.  
2035 403 of 2717 files processed.  
2036 396 of 2717 files processed.  
2034 397 of 2717 files processed.  
2037 401 of 2717 files processed.  
2036 397 of 2717 files processed.  
2034 398 of 2717 files processed.  
2037 402 of 2717 files processed.  
2035 404 of 2717 files processed.  
2035 405 of 2717 files processed.  
2034 399 of 2717 files processed.  
2037 403 of 2717 files processed.  
-----

2036 398 of 2717 files processed.  
2035 406 of 2717 files processed.  
2034 400 of 2717 files processed.  
2036 399 of 2717 files processed.  
2037 404 of 2717 files processed.  
2035 407 of 2717 files processed.  
2034 401 of 2717 files processed.  
2036 400 of 2717 files processed.  
2036 401 of 2717 files processed.  
2035 408 of 2717 files processed.  
2034 402 of 2717 files processed.  
2036 402 of 2717 files processed.  
2034 403 of 2717 files processed.  
2037 405 of 2717 files processed.  
2035 409 of 2717 files processed.  
2036 403 of 2717 files processed.  
2034 404 of 2717 files processed.  
2034 405 of 2717 files processed.  
2035 410 of 2717 files processed.  
2037 406 of 2717 files processed.  
2036 404 of 2717 files processed.  
2034 406 of 2717 files processed.  
2035 411 of 2717 files processed.  
2036 405 of 2717 files processed.  
2037 407 of 2717 files processed.  
2034 407 of 2717 files processed.  
2035 412 of 2717 files processed.  
2036 406 of 2717 files processed.  
2037 408 of 2717 files processed.  
2034 408 of 2717 files processed.  
2035 413 of 2717 files processed.  
2036 407 of 2717 files processed.  
2034 409 of 2717 files processed.  
2037 409 of 2717 files processed.  
2035 414 of 2717 files processed.  
2036 408 of 2717 files processed.  
2034 410 of 2717 files processed.  
2035 415 of 2717 files processed.  
2036 409 of 2717 files processed.  
2037 410 of 2717 files processed.  
2036 410 of 2717 files processed.  
2035 416 of 2717 files processed.  
2034 411 of 2717 files processed.  
2037 411 of 2717 files processed.  
2036 411 of 2717 files processed.  
2037 412 of 2717 files processed.  
2035 417 of 2717 files processed.  
2034 412 of 2717 files processed.  
2036 412 of 2717 files processed.  
2035 418 of 2717 files processed.  
2036 413 of 2717 files processed.  
2034 413 of 2717 files processed.  
2037 413 of 2717 files processed.  
2035 419 of 2717 files processed.  
2036 414 of 2717 files processed.  
2037 414 of 2717 files processed.  
2034 414 of 2717 files processed.  
2037 415 of 2717 files processed.  
2035 420 of 2717 files processed.  
2036 415 of 2717 files processed.  
2037 416 of 2717 files processed.  
2034 415 of 2717 files processed.  
2036 416 of 2717 files processed.  
2035 421 of 2717 files processed.  
2034 416 of 2717 files processed.  
2036 417 of 2717 files processed.  
2037 417 of 2717 files processed.  
2036 418 of 2717 files processed.  
2034 417 of 2717 files processed.  
2035 422 of 2717 files processed.  
2037 418 of 2717 files processed.  
2034 418 of 2717 files processed.  
2036 419 of 2717 files processed.  
2034 419 of 2717 files processed.  
2037 419 of 2717 files processed.  
2036 420 of 2717 files processed.  
2035 423 of 2717 files processed.

2034 420 of 2717 files processed.  
2037 420 of 2717 files processed.  
2036 421 of 2717 files processed.  
2035 424 of 2717 files processed.  
2037 421 of 2717 files processed.  
2036 422 of 2717 files processed.  
2034 421 of 2717 files processed.  
2035 425 of 2717 files processed.  
2037 422 of 2717 files processed.  
2036 423 of 2717 files processed.  
2037 423 of 2717 files processed.  
2037 424 of 2717 files processed.  
2034 422 of 2717 files processed.  
2036 424 of 2717 files processed.  
2035 426 of 2717 files processed.  
2037 425 of 2717 files processed.  
2034 423 of 2717 files processed.  
2035 427 of 2717 files processed.  
2036 425 of 2717 files processed.  
2034 424 of 2717 files processed.  
2037 426 of 2717 files processed.  
2035 428 of 2717 files processed.  
2036 426 of 2717 files processed.  
2034 425 of 2717 files processed.  
2035 429 of 2717 files processed.  
2036 427 of 2717 files processed.  
2037 427 of 2717 files processed.  
2035 430 of 2717 files processed.  
2034 426 of 2717 files processed.  
2035 431 of 2717 files processed.  
2037 428 of 2717 files processed.  
2036 428 of 2717 files processed.  
2034 427 of 2717 files processed.  
2034 428 of 2717 files processed.  
2036 429 of 2717 files processed.  
2035 432 of 2717 files processed.  
2037 429 of 2717 files processed.  
2034 429 of 2717 files processed.  
2037 430 of 2717 files processed.  
2036 430 of 2717 files processed.  
2035 433 of 2717 files processed.  
2037 431 of 2717 files processed.  
2034 430 of 2717 files processed.  
2036 431 of 2717 files processed.  
2035 434 of 2717 files processed.  
2036 432 of 2717 files processed.  
2037 432 of 2717 files processed.  
2035 435 of 2717 files processed.  
2034 431 of 2717 files processed.  
2036 433 of 2717 files processed.  
2035 436 of 2717 files processed.  
2034 432 of 2717 files processed.  
2036 434 of 2717 files processed.  
2037 433 of 2717 files processed.  
2035 437 of 2717 files processed.  
2037 434 of 2717 files processed.  
2034 433 of 2717 files processed.  
2036 435 of 2717 files processed.  
2035 438 of 2717 files processed.  
2034 434 of 2717 files processed.  
2036 436 of 2717 files processed.  
2037 435 of 2717 files processed.  
2034 435 of 2717 files processed.  
2035 439 of 2717 files processed.  
2037 436 of 2717 files processed.  
2034 436 of 2717 files processed.  
2035 440 of 2717 files processed.  
2037 437 of 2717 files processed.  
2036 437 of 2717 files processed.  
2035 441 of 2717 files processed.  
2037 438 of 2717 files processed.  
2034 437 of 2717 files processed.  
2035 442 of 2717 files processed.  
2036 438 of 2717 files processed.  
2034 438 of 2717 files processed.  
2037 439 of 2717 files processed.  
2036 439 of 2717 files processed.

2037 440 of 2717 files processed.  
2035 443 of 2717 files processed.  
2034 439 of 2717 files processed.  
2037 441 of 2717 files processed.  
2034 440 of 2717 files processed.  
2035 444 of 2717 files processed.  
2036 440 of 2717 files processed.  
2035 445 of 2717 files processed.  
2034 441 of 2717 files processed.  
2036 441 of 2717 files processed.  
2034 442 of 2717 files processed.  
2037 442 of 2717 files processed.  
2035 446 of 2717 files processed.  
2036 442 of 2717 files processed.  
2035 447 of 2717 files processed.  
2037 443 of 2717 files processed.  
2034 443 of 2717 files processed.  
2035 448 of 2717 files processed.  
2036 443 of 2717 files processed.  
2037 444 of 2717 files processed.  
2035 449 of 2717 files processed.  
2037 445 of 2717 files processed.  
2036 444 of 2717 files processed.  
2037 446 of 2717 files processed.  
2034 444 of 2717 files processed.  
2035 450 of 2717 files processed.  
2037 447 of 2717 files processed.  
2036 445 of 2717 files processed.  
2034 445 of 2717 files processed.  
2035 451 of 2717 files processed.  
2034 446 of 2717 files processed.  
2037 448 of 2717 files processed.  
2036 446 of 2717 files processed.  
2036 447 of 2717 files processed.  
2035 452 of 2717 files processed.  
2037 449 of 2717 files processed.  
2034 447 of 2717 files processed.  
2035 453 of 2717 files processed.  
2036 448 of 2717 files processed.  
2034 448 of 2717 files processed.  
2035 454 of 2717 files processed.  
2037 450 of 2717 files processed.  
2035 455 of 2717 files processed.  
2036 449 of 2717 files processed.  
2034 449 of 2717 files processed.  
2037 451 of 2717 files processed.  
2034 450 of 2717 files processed.  
2035 456 of 2717 files processed.  
2036 450 of 2717 files processed.  
2034 451 of 2717 files processed.  
2037 452 of 2717 files processed.  
2034 452 of 2717 files processed.  
2036 451 of 2717 files processed.  
2035 457 of 2717 files processed.  
2034 453 of 2717 files processed.  
2036 452 of 2717 files processed.  
2037 453 of 2717 files processed.  
2034 454 of 2717 files processed.  
2037 454 of 2717 files processed.  
2036 453 of 2717 files processed.  
2035 458 of 2717 files processed.  
2034 455 of 2717 files processed.  
2035 459 of 2717 files processed.  
2037 455 of 2717 files processed.  
2035 460 of 2717 files processed.  
2036 454 of 2717 files processed.  
2034 456 of 2717 files processed.  
2035 461 of 2717 files processed.  
2037 456 of 2717 files processed.  
2036 455 of 2717 files processed.  
2037 457 of 2717 files processed.  
2035 462 of 2717 files processed.  
2034 457 of 2717 files processed.  
2036 456 of 2717 files processed.  
2037 458 of 2717 files processed.  
2034 458 of 2717 files processed.  
2035 463 of 2717 files processed.

2037 459 of 2717 files processed.  
2034 459 of 2717 files processed.  
2036 457 of 2717 files processed.  
2037 460 of 2717 files processed.  
2034 460 of 2717 files processed.  
2035 464 of 2717 files processed.  
2034 461 of 2717 files processed.  
2035 465 of 2717 files processed.  
2037 461 of 2717 files processed.  
2036 458 of 2717 files processed.  
2034 462 of 2717 files processed.  
2037 462 of 2717 files processed.  
2035 466 of 2717 files processed.  
2034 463 of 2717 files processed.  
2037 463 of 2717 files processed.  
2036 459 of 2717 files processed.  
2034 464 of 2717 files processed.  
2034 465 of 2717 files processed.  
2035 467 of 2717 files processed.  
2036 460 of 2717 files processed.  
2034 466 of 2717 files processed.  
2037 464 of 2717 files processed.  
2035 468 of 2717 files processed.  
2036 461 of 2717 files processed.  
2034 467 of 2717 files processed.  
2035 469 of 2717 files processed.  
2035 470 of 2717 files processed.  
2036 462 of 2717 files processed.  
2037 465 of 2717 files processed.  
2035 471 of 2717 files processed.  
2036 463 of 2717 files processed.  
2034 468 of 2717 files processed.  
2036 464 of 2717 files processed.  
2037 466 of 2717 files processed.  
2035 472 of 2717 files processed.  
2037 467 of 2717 files processed.  
2034 469 of 2717 files processed.  
2035 473 of 2717 files processed.  
2037 468 of 2717 files processed.  
2034 470 of 2717 files processed.  
2035 474 of 2717 files processed.  
2036 465 of 2717 files processed.  
2034 471 of 2717 files processed.  
2035 475 of 2717 files processed.  
2037 469 of 2717 files processed.  
2036 466 of 2717 files processed.  
2034 472 of 2717 files processed.  
2034 473 of 2717 files processed.  
2035 476 of 2717 files processed.  
2036 467 of 2717 files processed.  
2037 470 of 2717 files processed.  
2034 474 of 2717 files processed.  
2035 477 of 2717 files processed.  
2036 468 of 2717 files processed.  
2034 475 of 2717 files processed.  
2035 478 of 2717 files processed.  
2034 476 of 2717 files processed.  
2037 471 of 2717 files processed.  
2036 469 of 2717 files processed.  
2034 477 of 2717 files processed.  
2036 470 of 2717 files processed.  
2037 472 of 2717 files processed.  
2035 479 of 2717 files processed.  
2034 478 of 2717 files processed.  
2035 480 of 2717 files processed.  
2034 479 of 2717 files processed.  
2037 473 of 2717 files processed.  
2036 471 of 2717 files processed.  
2034 480 of 2717 files processed.  
2037 474 of 2717 files processed.  
2035 481 of 2717 files processed.  
2036 472 of 2717 files processed.  
2037 475 of 2717 files processed.  
2035 482 of 2717 files processed.  
2034 481 of 2717 files processed.  
2036 473 of 2717 files processed.  
2037 476 of 2717 files processed.

2034 482 of 2717 files processed.  
2035 483 of 2717 files processed.  
2037 477 of 2717 files processed.  
2036 474 of 2717 files processed.  
2034 483 of 2717 files processed.  
2035 484 of 2717 files processed.  
2037 478 of 2717 files processed.  
2036 475 of 2717 files processed.  
2034 484 of 2717 files processed.  
2037 479 of 2717 files processed.  
2034 485 of 2717 files processed.  
2035 485 of 2717 files processed.  
2036 476 of 2717 files processed.  
2037 480 of 2717 files processed.  
2036 477 of 2717 files processed.  
2037 481 of 2717 files processed.  
2034 486 of 2717 files processed.  
2035 486 of 2717 files processed.  
2037 482 of 2717 files processed.  
2036 478 of 2717 files processed.  
2037 483 of 2717 files processed.  
2034 487 of 2717 files processed.  
2035 487 of 2717 files processed.  
2034 488 of 2717 files processed.  
2036 479 of 2717 files processed.  
2035 488 of 2717 files processed.  
2035 489 of 2717 files processed.  
2034 489 of 2717 files processed.  
2037 484 of 2717 files processed.  
2035 490 of 2717 files processed.  
2034 490 of 2717 files processed.  
2037 485 of 2717 files processed.  
2036 480 of 2717 files processed.  
2035 491 of 2717 files processed.  
2034 491 of 2717 files processed.  
2036 481 of 2717 files processed.  
2037 486 of 2717 files processed.  
2035 492 of 2717 files processed.  
2034 492 of 2717 files processed.  
2036 482 of 2717 files processed.  
2037 487 of 2717 files processed.  
2035 493 of 2717 files processed.  
2034 493 of 2717 files processed.  
2036 483 of 2717 files processed.  
2037 488 of 2717 files processed.  
2035 494 of 2717 files processed.  
2034 494 of 2717 files processed.  
2036 484 of 2717 files processed.  
2037 489 of 2717 files processed.  
2037 490 of 2717 files processed.  
2035 495 of 2717 files processed.  
2034 495 of 2717 files processed.  
2036 485 of 2717 files processed.  
2035 496 of 2717 files processed.  
2037 491 of 2717 files processed.  
2036 486 of 2717 files processed.  
2034 496 of 2717 files processed.  
2035 497 of 2717 files processed.  
2036 487 of 2717 files processed.  
2035 498 of 2717 files processed.  
2034 497 of 2717 files processed.  
2037 492 of 2717 files processed.  
2034 498 of 2717 files processed.  
2036 488 of 2717 files processed.  
2035 499 of 2717 files processed.  
2034 499 of 2717 files processed.  
2036 489 of 2717 files processed.  
2037 493 of 2717 files processed.  
2035 500 of 2717 files processed.  
2034 500 of 2717 files processed.  
2036 490 of 2717 files processed.  
2035 501 of 2717 files processed.  
2034 501 of 2717 files processed.  
2036 491 of 2717 files processed.  
2037 494 of 2717 files processed.  
2036 492 of 2717 files processed.  
2035 502 of 2717 files processed.

2037 495 of 2717 files processed.  
2034 502 of 2717 files processed.  
2036 493 of 2717 files processed.  
2034 503 of 2717 files processed.  
2035 503 of 2717 files processed.  
2037 496 of 2717 files processed.  
2036 494 of 2717 files processed.  
2035 504 of 2717 files processed.  
2037 497 of 2717 files processed.  
2036 495 of 2717 files processed.  
2034 504 of 2717 files processed.  
2037 498 of 2717 files processed.  
2035 505 of 2717 files processed.  
2036 496 of 2717 files processed.  
2037 499 of 2717 files processed.  
2034 505 of 2717 files processed.  
2034 506 of 2717 files processed.  
2037 500 of 2717 files processed.  
2036 497 of 2717 files processed.  
2034 507 of 2717 files processed.  
2037 501 of 2717 files processed.  
2035 506 of 2717 files processed.  
2034 508 of 2717 files processed.  
2037 502 of 2717 files processed.  
2036 498 of 2717 files processed.  
2035 507 of 2717 files processed.  
2037 503 of 2717 files processed.  
2036 499 of 2717 files processed.  
2034 509 of 2717 files processed.  
2037 504 of 2717 files processed.  
2035 508 of 2717 files processed.  
2034 510 of 2717 files processed.  
2036 500 of 2717 files processed.  
2037 505 of 2717 files processed.  
2035 509 of 2717 files processed.  
2036 501 of 2717 files processed.  
2037 506 of 2717 files processed.  
2034 511 of 2717 files processed.  
2035 510 of 2717 files processed.  
2037 507 of 2717 files processed.  
2036 502 of 2717 files processed.  
2037 508 of 2717 files processed.  
2034 512 of 2717 files processed.  
2036 503 of 2717 files processed.  
2037 509 of 2717 files processed.  
2035 511 of 2717 files processed.  
2034 513 of 2717 files processed.  
2037 510 of 2717 files processed.  
2035 512 of 2717 files processed.  
2036 504 of 2717 files processed.  
2037 511 of 2717 files processed.  
2034 514 of 2717 files processed.  
2035 513 of 2717 files processed.  
2036 505 of 2717 files processed.  
2037 512 of 2717 files processed.  
2036 506 of 2717 files processed.  
2035 514 of 2717 files processed.  
2034 515 of 2717 files processed.  
2037 513 of 2717 files processed.  
2035 515 of 2717 files processed.  
2036 507 of 2717 files processed.  
2037 514 of 2717 files processed.  
2034 516 of 2717 files processed.  
2035 516 of 2717 files processed.  
2037 515 of 2717 files processed.  
2034 517 of 2717 files processed.  
2036 508 of 2717 files processed.  
2035 517 of 2717 files processed.  
2034 518 of 2717 files processed.  
2036 509 of 2717 files processed.  
2037 516 of 2717 files processed.  
2035 518 of 2717 files processed.  
2036 510 of 2717 files processed.  
2037 517 of 2717 files processed.  
2034 519 of 2717 files processed.  
2035 519 of 2717 files processed.  
2037 518 of 2717 files processed.

2034 520 of 2717 files processed.  
2036 511 of 2717 files processed.  
2037 519 of 2717 files processed.  
2035 520 of 2717 files processed.  
2034 521 of 2717 files processed.  
2037 520 of 2717 files processed.  
2035 521 of 2717 files processed.  
2036 512 of 2717 files processed.  
2037 521 of 2717 files processed.  
2035 522 of 2717 files processed.  
2034 522 of 2717 files processed.  
2036 513 of 2717 files processed.  
2037 522 of 2717 files processed.  
2034 523 of 2717 files processed.  
2035 523 of 2717 files processed.  
2036 514 of 2717 files processed.  
2037 523 of 2717 files processed.  
2036 515 of 2717 files processed.  
2035 524 of 2717 files processed.  
2037 524 of 2717 files processed.  
2034 524 of 2717 files processed.  
2035 525 of 2717 files processed.  
2037 525 of 2717 files processed.  
2036 516 of 2717 files processed.  
2034 525 of 2717 files processed.  
2036 517 of 2717 files processed.  
2035 526 of 2717 files processed.  
2034 526 of 2717 files processed.  
2037 526 of 2717 files processed.  
2035 527 of 2717 files processed.  
2037 527 of 2717 files processed.  
2036 518 of 2717 files processed.  
2034 527 of 2717 files processed.  
2037 528 of 2717 files processed.  
2034 528 of 2717 files processed.  
2035 528 of 2717 files processed.  
2036 519 of 2717 files processed.  
2037 529 of 2717 files processed.  
2034 529 of 2717 files processed.  
2035 529 of 2717 files processed.  
2034 530 of 2717 files processed.  
2037 530 of 2717 files processed.  
2035 530 of 2717 files processed.  
2036 520 of 2717 files processed.  
2034 531 of 2717 files processed.  
2035 531 of 2717 files processed.  
2036 521 of 2717 files processed.  
2034 532 of 2717 files processed.  
2035 532 of 2717 files processed.  
2036 522 of 2717 files processed.  
2037 531 of 2717 files processed.  
2035 533 of 2717 files processed.  
2034 533 of 2717 files processed.  
2037 532 of 2717 files processed.  
2035 534 of 2717 files processed.  
2036 523 of 2717 files processed.  
2034 534 of 2717 files processed.  
2035 535 of 2717 files processed.  
2036 524 of 2717 files processed.  
2037 533 of 2717 files processed.  
2035 536 of 2717 files processed.  
2036 525 of 2717 files processed.  
2034 535 of 2717 files processed.  
2037 534 of 2717 files processed.  
2035 537 of 2717 files processed.  
2036 526 of 2717 files processed.  
2034 536 of 2717 files processed.  
2035 538 of 2717 files processed.  
2036 527 of 2717 files processed.  
2037 535 of 2717 files processed.  
2034 537 of 2717 files processed.  
2034 538 of 2717 files processed.  
2037 536 of 2717 files processed.  
2035 539 of 2717 files processed.  
2036 528 of 2717 files processed.  
2035 540 of 2717 files processed.  
2037 537 of 2717 files processed.

2036 529 of 2717 files processed.  
2037 538 of 2717 files processed.  
2036 530 of 2717 files processed.  
2034 539 of 2717 files processed.  
2035 541 of 2717 files processed.  
2037 539 of 2717 files processed.  
2034 540 of 2717 files processed.  
2036 531 of 2717 files processed.  
2037 540 of 2717 files processed.  
2035 542 of 2717 files processed.  
2034 541 of 2717 files processed.  
2036 532 of 2717 files processed.  
2036 533 of 2717 files processed.  
2035 543 of 2717 files processed.  
2034 542 of 2717 files processed.  
2037 541 of 2717 files processed.  
2034 543 of 2717 files processed.  
2036 534 of 2717 files processed.  
2035 544 of 2717 files processed.  
2037 542 of 2717 files processed.  
2034 544 of 2717 files processed.  
2036 535 of 2717 files processed.  
2035 545 of 2717 files processed.  
2037 543 of 2717 files processed.  
2034 545 of 2717 files processed.  
2035 546 of 2717 files processed.  
2037 544 of 2717 files processed.  
2036 536 of 2717 files processed.  
2034 546 of 2717 files processed.  
2037 545 of 2717 files processed.  
2036 537 of 2717 files processed.  
2035 547 of 2717 files processed.  
2034 547 of 2717 files processed.  
2037 546 of 2717 files processed.  
2036 538 of 2717 files processed.  
2034 548 of 2717 files processed.  
2035 548 of 2717 files processed.  
2037 547 of 2717 files processed.  
2034 549 of 2717 files processed.  
2036 539 of 2717 files processed.  
2035 549 of 2717 files processed.  
2037 548 of 2717 files processed.  
2036 540 of 2717 files processed.  
2035 550 of 2717 files processed.  
2037 549 of 2717 files processed.  
2034 550 of 2717 files processed.  
2036 541 of 2717 files processed.  
2034 551 of 2717 files processed.  
2035 551 of 2717 files processed.  
2037 550 of 2717 files processed.  
2036 542 of 2717 files processed.  
2035 552 of 2717 files processed.  
2034 552 of 2717 files processed.  
2037 551 of 2717 files processed.  
2036 543 of 2717 files processed.  
2035 553 of 2717 files processed.  
2034 553 of 2717 files processed.  
2037 552 of 2717 files processed.  
2036 544 of 2717 files processed.  
2037 553 of 2717 files processed.  
2034 554 of 2717 files processed.  
2037 554 of 2717 files processed.  
2035 554 of 2717 files processed.  
2034 555 of 2717 files processed.  
2036 545 of 2717 files processed.  
2037 555 of 2717 files processed.  
2036 546 of 2717 files processed.  
2035 555 of 2717 files processed.  
2034 556 of 2717 files processed.  
2037 556 of 2717 files processed.  
2036 547 of 2717 files processed.  
2035 556 of 2717 files processed.  
2034 557 of 2717 files processed.  
2037 557 of 2717 files processed.  
2036 548 of 2717 files processed.  
2037 558 of 2717 files processed.  
2035 557 of 2717 files processed.

2034 558 of 2717 files processed.  
2034 559 of 2717 files processed.  
2036 549 of 2717 files processed.  
2035 558 of 2717 files processed.  
2037 559 of 2717 files processed.  
2034 560 of 2717 files processed.  
2035 559 of 2717 files processed.  
2036 550 of 2717 files processed.  
2037 560 of 2717 files processed.  
2034 561 of 2717 files processed.  
2035 560 of 2717 files processed.  
2037 561 of 2717 files processed.  
2034 562 of 2717 files processed.  
2036 551 of 2717 files processed.  
2035 561 of 2717 files processed.  
2037 562 of 2717 files processed.  
2034 563 of 2717 files processed.  
2036 552 of 2717 files processed.  
2034 564 of 2717 files processed.  
2037 563 of 2717 files processed.  
2035 562 of 2717 files processed.  
2036 553 of 2717 files processed.  
2034 565 of 2717 files processed.  
2037 564 of 2717 files processed.  
2035 563 of 2717 files processed.  
2034 566 of 2717 files processed.  
2036 554 of 2717 files processed.  
2037 565 of 2717 files processed.  
2035 564 of 2717 files processed.  
2034 567 of 2717 files processed.  
2036 555 of 2717 files processed.  
2037 566 of 2717 files processed.  
2035 565 of 2717 files processed.  
2036 556 of 2717 files processed.  
2037 567 of 2717 files processed.  
2035 566 of 2717 files processed.  
2037 568 of 2717 files processed.  
2034 568 of 2717 files processed.  
2035 567 of 2717 files processed.  
2036 557 of 2717 files processed.  
2034 569 of 2717 files processed.  
2036 558 of 2717 files processed.  
2037 569 of 2717 files processed.  
2035 568 of 2717 files processed.  
2034 570 of 2717 files processed.  
2035 569 of 2717 files processed.  
2036 559 of 2717 files processed.  
2037 570 of 2717 files processed.  
2034 571 of 2717 files processed.  
2035 570 of 2717 files processed.  
2037 571 of 2717 files processed.  
2034 572 of 2717 files processed.  
2036 560 of 2717 files processed.  
2037 572 of 2717 files processed.  
2035 571 of 2717 files processed.  
2036 561 of 2717 files processed.  
2034 573 of 2717 files processed.  
2036 562 of 2717 files processed.  
2035 572 of 2717 files processed.  
2037 573 of 2717 files processed.  
2034 574 of 2717 files processed.  
2035 573 of 2717 files processed.  
2036 563 of 2717 files processed.  
2037 574 of 2717 files processed.  
2034 575 of 2717 files processed.  
2035 574 of 2717 files processed.  
2036 564 of 2717 files processed.  
2037 575 of 2717 files processed.  
2034 576 of 2717 files processed.  
2037 576 of 2717 files processed.  
2035 575 of 2717 files processed.  
2036 565 of 2717 files processed.  
2034 577 of 2717 files processed.  
2037 577 of 2717 files processed.  
2035 576 of 2717 files processed.  
2036 566 of 2717 files processed.  
2035 577 of 2717 files processed.

-----  
2034 578 of 2717 files processed.  
2037 578 of 2717 files processed.  
2036 567 of 2717 files processed.  
2034 579 of 2717 files processed.  
2036 568 of 2717 files processed.  
2035 578 of 2717 files processed.  
2037 579 of 2717 files processed.  
2034 580 of 2717 files processed.  
2036 569 of 2717 files processed.  
2035 579 of 2717 files processed.  
2037 580 of 2717 files processed.  
2036 570 of 2717 files processed.  
2035 580 of 2717 files processed.  
2034 581 of 2717 files processed.  
2037 581 of 2717 files processed.  
2036 571 of 2717 files processed.  
2034 582 of 2717 files processed.  
2035 581 of 2717 files processed.  
2037 582 of 2717 files processed.  
2034 583 of 2717 files processed.  
2036 572 of 2717 files processed.  
2037 583 of 2717 files processed.  
2034 584 of 2717 files processed.  
2035 582 of 2717 files processed.  
2036 573 of 2717 files processed.  
2034 585 of 2717 files processed.  
2035 583 of 2717 files processed.  
2037 584 of 2717 files processed.  
2036 574 of 2717 files processed.  
2035 584 of 2717 files processed.  
2037 585 of 2717 files processed.  
2034 586 of 2717 files processed.  
2036 575 of 2717 files processed.  
2036 576 of 2717 files processed.  
2035 585 of 2717 files processed.  
2037 586 of 2717 files processed.  
2034 587 of 2717 files processed.  
2034 588 of 2717 files processed.  
2035 586 of 2717 files processed.  
2036 577 of 2717 files processed.  
2037 587 of 2717 files processed.  
2037 588 of 2717 files processed.  
2034 589 of 2717 files processed.  
2036 578 of 2717 files processed.  
2035 587 of 2717 files processed.  
2037 589 of 2717 files processed.  
2034 590 of 2717 files processed.  
2036 579 of 2717 files processed.  
2035 588 of 2717 files processed.  
2034 591 of 2717 files processed.  
2036 580 of 2717 files processed.  
2037 590 of 2717 files processed.  
2035 589 of 2717 files processed.  
2037 591 of 2717 files processed.  
2036 581 of 2717 files processed.  
2035 590 of 2717 files processed.  
2034 592 of 2717 files processed.  
2035 591 of 2717 files processed.  
2037 592 of 2717 files processed.  
2036 582 of 2717 files processed.  
2034 593 of 2717 files processed.  
2037 593 of 2717 files processed.  
2035 592 of 2717 files processed.  
2036 583 of 2717 files processed.  
2035 593 of 2717 files processed.  
2034 594 of 2717 files processed.  
2036 584 of 2717 files processed.  
2037 594 of 2717 files processed.  
2035 594 of 2717 files processed.  
2034 595 of 2717 files processed.  
2037 595 of 2717 files processed.  
2034 596 of 2717 files processed.  
2036 585 of 2717 files processed.  
2037 596 of 2717 files processed.  
2035 595 of 2717 files processed.  
2034 597 of 2717 files processed.  
2035 596 of 2717 files processed.

.....  
2036 586 of 2717 files processed.  
2037 597 of 2717 files processed.  
2034 598 of 2717 files processed.  
2037 598 of 2717 files processed.  
2036 587 of 2717 files processed.  
2035 597 of 2717 files processed.  
2034 599 of 2717 files processed.  
2037 599 of 2717 files processed.  
2035 598 of 2717 files processed.  
2036 588 of 2717 files processed.  
2037 600 of 2717 files processed.  
2034 600 of 2717 files processed.  
2035 599 of 2717 files processed.  
2036 589 of 2717 files processed.  
2037 601 of 2717 files processed.  
2034 601 of 2717 files processed.  
2035 600 of 2717 files processed.  
2037 602 of 2717 files processed.  
2036 590 of 2717 files processed.  
2034 602 of 2717 files processed.  
2037 603 of 2717 files processed.  
2035 601 of 2717 files processed.  
2034 603 of 2717 files processed.  
2036 591 of 2717 files processed.  
2037 604 of 2717 files processed.  
2036 592 of 2717 files processed.  
2035 602 of 2717 files processed.  
2037 605 of 2717 files processed.  
2034 604 of 2717 files processed.  
2035 603 of 2717 files processed.  
2036 593 of 2717 files processed.  
2037 606 of 2717 files processed.  
2034 605 of 2717 files processed.  
2035 604 of 2717 files processed.  
2036 594 of 2717 files processed.  
2035 605 of 2717 files processed.  
2036 595 of 2717 files processed.  
2037 607 of 2717 files processed.  
2034 606 of 2717 files processed.  
2036 596 of 2717 files processed.  
2037 608 of 2717 files processed.  
2035 606 of 2717 files processed.  
2036 597 of 2717 files processed.  
2034 607 of 2717 files processed.  
2035 607 of 2717 files processed.  
2037 609 of 2717 files processed.  
2036 598 of 2717 files processed.  
2035 608 of 2717 files processed.  
2036 599 of 2717 files processed.  
2037 610 of 2717 files processed.  
2034 608 of 2717 files processed.  
2035 609 of 2717 files processed.  
2037 611 of 2717 files processed.  
2035 610 of 2717 files processed.  
2034 609 of 2717 files processed.  
2036 600 of 2717 files processed.  
2037 612 of 2717 files processed.  
2034 610 of 2717 files processed.  
2036 601 of 2717 files processed.  
2037 613 of 2717 files processed.  
2035 611 of 2717 files processed.  
2036 602 of 2717 files processed.  
2037 614 of 2717 files processed.  
2035 612 of 2717 files processed.  
2034 611 of 2717 files processed.  
2036 603 of 2717 files processed.  
2035 613 of 2717 files processed.  
2037 615 of 2717 files processed.  
2036 604 of 2717 files processed.  
2034 612 of 2717 files processed.  
2037 616 of 2717 files processed.  
2035 614 of 2717 files processed.  
2036 605 of 2717 files processed.  
2034 613 of 2717 files processed.  
2035 615 of 2717 files processed.  
2034 614 of 2717 files processed.  
2037 617 of 2717 files processed

2037 617 of 2717 files processed.  
2034 615 of 2717 files processed.  
2035 616 of 2717 files processed.  
2036 606 of 2717 files processed.  
2034 616 of 2717 files processed.  
2037 618 of 2717 files processed.  
2036 607 of 2717 files processed.  
2035 617 of 2717 files processed.  
2037 619 of 2717 files processed.  
2034 617 of 2717 files processed.  
2036 608 of 2717 files processed.  
2035 618 of 2717 files processed.  
2034 618 of 2717 files processed.  
2036 609 of 2717 files processed.  
2037 620 of 2717 files processed.  
2035 619 of 2717 files processed.  
2034 619 of 2717 files processed.  
2036 610 of 2717 files processed.  
2037 621 of 2717 files processed.  
2034 620 of 2717 files processed.  
2035 620 of 2717 files processed.  
2036 611 of 2717 files processed.  
2037 622 of 2717 files processed.  
2034 621 of 2717 files processed.  
2036 612 of 2717 files processed.  
2035 621 of 2717 files processed.  
2037 623 of 2717 files processed.  
2034 622 of 2717 files processed.  
2036 613 of 2717 files processed.  
2035 622 of 2717 files processed.  
2034 623 of 2717 files processed.  
2037 624 of 2717 files processed.  
2036 614 of 2717 files processed.  
2035 623 of 2717 files processed.  
2034 624 of 2717 files processed.  
2035 624 of 2717 files processed.  
2036 615 of 2717 files processed.  
2034 625 of 2717 files processed.  
2037 625 of 2717 files processed.  
2035 625 of 2717 files processed.  
2036 616 of 2717 files processed.  
2037 626 of 2717 files processed.  
2034 626 of 2717 files processed.  
2035 626 of 2717 files processed.  
2036 617 of 2717 files processed.  
2034 627 of 2717 files processed.  
2037 627 of 2717 files processed.  
2036 618 of 2717 files processed.  
2035 627 of 2717 files processed.  
2034 628 of 2717 files processed.  
2037 628 of 2717 files processed.  
2036 619 of 2717 files processed.  
2035 628 of 2717 files processed.  
2034 629 of 2717 files processed.  
2036 620 of 2717 files processed.  
2035 629 of 2717 files processed.  
2037 629 of 2717 files processed.  
2036 621 of 2717 files processed.  
2035 630 of 2717 files processed.  
2037 630 of 2717 files processed.  
2036 622 of 2717 files processed.  
2034 630 of 2717 files processed.  
2037 631 of 2717 files processed.  
2036 623 of 2717 files processed.  
2035 631 of 2717 files processed.  
2034 631 of 2717 files processed.  
2037 632 of 2717 files processed.  
2036 624 of 2717 files processed.  
2035 632 of 2717 files processed.  
2037 633 of 2717 files processed.  
2034 632 of 2717 files processed.  
2035 633 of 2717 files processed.  
2036 625 of 2717 files processed.  
2035 634 of 2717 files processed.  
2036 626 of 2717 files processed.  
2037 634 of 2717 files processed.  
2034 633 of 2717 files processed.  
2037 635 of 2717 files processed

2035 635 of 2717 files processed.  
2035 635 of 2717 files processed.  
2036 627 of 2717 files processed.  
2034 634 of 2717 files processed.  
2037 636 of 2717 files processed.  
2036 628 of 2717 files processed.  
2035 636 of 2717 files processed.  
2037 637 of 2717 files processed.  
2036 629 of 2717 files processed.  
2035 637 of 2717 files processed.  
2037 638 of 2717 files processed.  
2034 635 of 2717 files processed.  
2036 630 of 2717 files processed.  
2037 639 of 2717 files processed.  
2036 631 of 2717 files processed.  
2035 638 of 2717 files processed.  
2034 636 of 2717 files processed.  
2036 632 of 2717 files processed.  
2037 640 of 2717 files processed.  
2035 639 of 2717 files processed.  
2034 637 of 2717 files processed.  
2036 633 of 2717 files processed.  
2037 641 of 2717 files processed.  
2034 638 of 2717 files processed.  
2035 640 of 2717 files processed.  
2037 642 of 2717 files processed.  
2036 634 of 2717 files processed.  
2035 641 of 2717 files processed.  
2034 639 of 2717 files processed.  
2037 643 of 2717 files processed.  
2035 642 of 2717 files processed.  
2037 644 of 2717 files processed.  
2034 640 of 2717 files processed.  
2035 643 of 2717 files processed.  
2036 635 of 2717 files processed.  
2037 645 of 2717 files processed.  
2035 644 of 2717 files processed.  
2036 636 of 2717 files processed.  
2034 641 of 2717 files processed.  
2037 646 of 2717 files processed.  
2035 645 of 2717 files processed.  
2036 637 of 2717 files processed.  
2037 647 of 2717 files processed.  
2035 646 of 2717 files processed.  
2034 642 of 2717 files processed.  
2036 638 of 2717 files processed.  
2035 647 of 2717 files processed.  
2037 648 of 2717 files processed.  
2036 639 of 2717 files processed.  
2034 643 of 2717 files processed.  
2035 648 of 2717 files processed.  
2037 649 of 2717 files processed.  
2034 644 of 2717 files processed.  
2036 640 of 2717 files processed.  
2035 649 of 2717 files processed.  
2037 650 of 2717 files processed.  
2034 645 of 2717 files processed.  
2036 641 of 2717 files processed.  
2035 650 of 2717 files processed.  
2034 646 of 2717 files processed.  
2037 651 of 2717 files processed.  
2035 651 of 2717 files processed.  
2034 647 of 2717 files processed.  
2036 642 of 2717 files processed.  
2037 652 of 2717 files processed.  
2034 648 of 2717 files processed.  
2036 643 of 2717 files processed.  
2035 652 of 2717 files processed.  
2034 649 of 2717 files processed.  
2036 644 of 2717 files processed.  
2037 653 of 2717 files processed.  
2035 653 of 2717 files processed.  
2036 645 of 2717 files processed.  
2034 650 of 2717 files processed.  
2035 654 of 2717 files processed.  
2037 654 of 2717 files processed.  
2034 651 of 2717 files processed.  
2036 646 of 2717 files processed.

2030 640 of 2717 files processed.  
2035 655 of 2717 files processed.  
2037 655 of 2717 files processed.  
2036 647 of 2717 files processed.  
2037 656 of 2717 files processed.  
2034 652 of 2717 files processed.  
2036 648 of 2717 files processed.  
2035 656 of 2717 files processed.  
2037 657 of 2717 files processed.  
2034 653 of 2717 files processed.  
2036 649 of 2717 files processed.  
2035 657 of 2717 files processed.  
2037 658 of 2717 files processed.  
2036 650 of 2717 files processed.  
2034 654 of 2717 files processed.  
2037 659 of 2717 files processed.  
2036 651 of 2717 files processed.  
2035 658 of 2717 files processed.  
2034 655 of 2717 files processed.  
2037 660 of 2717 files processed.  
2036 652 of 2717 files processed.  
2035 659 of 2717 files processed.  
2037 661 of 2717 files processed.  
2034 656 of 2717 files processed.  
2036 653 of 2717 files processed.  
2034 657 of 2717 files processed.  
2035 660 of 2717 files processed.  
2036 654 of 2717 files processed.  
2037 662 of 2717 files processed.  
2034 658 of 2717 files processed.  
2035 661 of 2717 files processed.  
2036 655 of 2717 files processed.  
2034 659 of 2717 files processed.  
2037 663 of 2717 files processed.  
2035 662 of 2717 files processed.  
2034 660 of 2717 files processed.  
2036 656 of 2717 files processed.  
2037 664 of 2717 files processed.  
2034 661 of 2717 files processed.  
2036 657 of 2717 files processed.  
2035 663 of 2717 files processed.  
2037 665 of 2717 files processed.  
2034 662 of 2717 files processed.  
2036 658 of 2717 files processed.  
2035 664 of 2717 files processed.  
2036 659 of 2717 files processed.  
2037 666 of 2717 files processed.  
2034 663 of 2717 files processed.  
2035 665 of 2717 files processed.  
2034 664 of 2717 files processed.  
2036 660 of 2717 files processed.  
2037 667 of 2717 files processed.  
2035 666 of 2717 files processed.  
2036 661 of 2717 files processed.  
2034 665 of 2717 files processed.  
2037 668 of 2717 files processed.  
2035 667 of 2717 files processed.  
2034 666 of 2717 files processed.  
2036 662 of 2717 files processed.  
2037 669 of 2717 files processed.  
2036 663 of 2717 files processed.  
2035 668 of 2717 files processed.  
2037 670 of 2717 files processed.  
2034 667 of 2717 files processed.  
2036 664 of 2717 files processed.  
2034 668 of 2717 files processed.  
2036 665 of 2717 files processed.  
2035 669 of 2717 files processed.  
2037 671 of 2717 files processed.  
2034 669 of 2717 files processed.  
2034 670 of 2717 files processed.  
2036 666 of 2717 files processed.  
2035 670 of 2717 files processed.  
2034 671 of 2717 files processed.  
2037 672 of 2717 files processed.  
2035 671 of 2717 files processed.  
2036 667 of 2717 files processed.  
2035 670 of 2717 files processed.

2035 674 of 2717 files processed.  
2037 673 of 2717 files processed.  
2034 672 of 2717 files processed.  
2036 668 of 2717 files processed.  
2034 673 of 2717 files processed.  
2035 673 of 2717 files processed.  
2037 674 of 2717 files processed.  
2034 674 of 2717 files processed.  
2035 674 of 2717 files processed.  
2036 669 of 2717 files processed.  
2037 675 of 2717 files processed.  
2034 675 of 2717 files processed.  
2036 670 of 2717 files processed.  
2037 676 of 2717 files processed.  
2035 675 of 2717 files processed.  
2036 671 of 2717 files processed.  
2034 676 of 2717 files processed.  
2035 676 of 2717 files processed.  
2037 677 of 2717 files processed.  
2034 677 of 2717 files processed.  
2036 672 of 2717 files processed.  
2035 677 of 2717 files processed.  
2037 678 of 2717 files processed.  
2034 678 of 2717 files processed.  
2036 673 of 2717 files processed.  
2035 678 of 2717 files processed.  
2037 679 of 2717 files processed.  
2036 674 of 2717 files processed.  
2035 679 of 2717 files processed.  
2034 679 of 2717 files processed.  
2037 680 of 2717 files processed.  
2034 680 of 2717 files processed.  
2036 675 of 2717 files processed.  
2037 681 of 2717 files processed.  
2036 676 of 2717 files processed.  
2035 680 of 2717 files processed.  
2034 681 of 2717 files processed.  
2035 681 of 2717 files processed.  
2034 682 of 2717 files processed.  
2037 682 of 2717 files processed.  
2036 677 of 2717 files processed.  
2037 683 of 2717 files processed.  
2035 682 of 2717 files processed.  
2034 683 of 2717 files processed.  
2036 678 of 2717 files processed.  
2037 684 of 2717 files processed.  
2035 683 of 2717 files processed.  
2034 684 of 2717 files processed.  
2036 679 of 2717 files processed.  
2035 684 of 2717 files processed.  
2037 685 of 2717 files processed.  
2037 686 of 2717 files processed.  
2036 680 of 2717 files processed.  
2034 685 of 2717 files processed.  
2035 685 of 2717 files processed.  
2037 687 of 2717 files processed.  
2036 681 of 2717 files processed.  
2035 686 of 2717 files processed.  
2036 682 of 2717 files processed.  
2037 688 of 2717 files processed.  
2034 686 of 2717 files processed.  
2036 683 of 2717 files processed.  
2035 687 of 2717 files processed.  
2037 689 of 2717 files processed.  
2034 687 of 2717 files processed.  
2035 688 of 2717 files processed.  
2035 689 of 2717 files processed.  
2036 684 of 2717 files processed.  
2037 690 of 2717 files processed.  
2034 688 of 2717 files processed.  
2035 690 of 2717 files processed.  
2034 689 of 2717 files processed.  
2036 685 of 2717 files processed.  
2037 691 of 2717 files processed.  
2037 692 of 2717 files processed.  
2034 690 of 2717 files processed.  
2035 691 of 2717 files processed.

2036 686 of 2717 files processed.  
2034 691 of 2717 files processed.  
2037 693 of 2717 files processed.  
2035 692 of 2717 files processed.  
2036 687 of 2717 files processed.  
2034 692 of 2717 files processed.  
2037 694 of 2717 files processed.  
2036 688 of 2717 files processed.  
2035 693 of 2717 files processed.  
2034 693 of 2717 files processed.  
2037 695 of 2717 files processed.  
2035 694 of 2717 files processed.  
2036 689 of 2717 files processed.  
2034 694 of 2717 files processed.  
2035 695 of 2717 files processed.  
2037 696 of 2717 files processed.  
2036 690 of 2717 files processed.  
2035 696 of 2717 files processed.  
2034 695 of 2717 files processed.  
2037 697 of 2717 files processed.  
2036 691 of 2717 files processed.  
2034 696 of 2717 files processed.  
2035 697 of 2717 files processed.  
2037 698 of 2717 files processed.  
2036 692 of 2717 files processed.  
2034 697 of 2717 files processed.  
2037 699 of 2717 files processed.  
2036 693 of 2717 files processed.  
2035 698 of 2717 files processed.  
2037 700 of 2717 files processed.  
2034 698 of 2717 files processed.  
2035 699 of 2717 files processed.  
2037 701 of 2717 files processed.  
2036 694 of 2717 files processed.  
2034 699 of 2717 files processed.  
2037 702 of 2717 files processed.  
2035 700 of 2717 files processed.  
2036 695 of 2717 files processed.  
2034 700 of 2717 files processed.  
2036 696 of 2717 files processed.  
2034 701 of 2717 files processed.  
2037 703 of 2717 files processed.  
2036 697 of 2717 files processed.  
2035 701 of 2717 files processed.  
2034 702 of 2717 files processed.  
2037 704 of 2717 files processed.  
2036 698 of 2717 files processed.  
2034 703 of 2717 files processed.  
2035 702 of 2717 files processed.  
2037 705 of 2717 files processed.  
2034 704 of 2717 files processed.  
2035 703 of 2717 files processed.  
2036 699 of 2717 files processed.  
2037 706 of 2717 files processed.  
2034 705 of 2717 files processed.  
2036 700 of 2717 files processed.  
2035 704 of 2717 files processed.  
2037 707 of 2717 files processed.  
2035 705 of 2717 files processed.  
2034 706 of 2717 files processed.  
2036 701 of 2717 files processed.  
2037 708 of 2717 files processed.  
2037 709 of 2717 files processed.  
2036 702 of 2717 files processed.  
2034 707 of 2717 files processed.  
2035 706 of 2717 files processed.  
2037 710 of 2717 files processed.  
2036 703 of 2717 files processed.  
2037 711 of 2717 files processed.  
2034 708 of 2717 files processed.  
2035 707 of 2717 files processed.  
2037 712 of 2717 files processed.  
2036 704 of 2717 files processed.  
2034 709 of 2717 files processed.  
2035 708 of 2717 files processed.  
2037 713 of 2717 files processed.  
2036 705 of 2717 files processed.  
-----

2035 709 of 2717 files processed.  
2034 710 of 2717 files processed.  
2036 706 of 2717 files processed.  
2035 710 of 2717 files processed.  
2037 714 of 2717 files processed.  
2034 711 of 2717 files processed.  
2035 711 of 2717 files processed.  
2037 715 of 2717 files processed.  
2036 707 of 2717 files processed.  
2034 712 of 2717 files processed.  
2037 716 of 2717 files processed.  
2035 712 of 2717 files processed.  
2036 708 of 2717 files processed.  
2036 709 of 2717 files processed.  
2037 717 of 2717 files processed.  
2034 713 of 2717 files processed.  
2035 713 of 2717 files processed.  
2034 714 of 2717 files processed.  
2036 710 of 2717 files processed.  
2035 714 of 2717 files processed.  
2037 718 of 2717 files processed.  
2034 715 of 2717 files processed.  
2035 715 of 2717 files processed.  
2036 711 of 2717 files processed.  
2037 719 of 2717 files processed.  
2034 716 of 2717 files processed.  
2035 716 of 2717 files processed.  
2036 712 of 2717 files processed.  
2037 720 of 2717 files processed.  
2035 717 of 2717 files processed.  
2034 717 of 2717 files processed.  
2036 713 of 2717 files processed.  
2037 721 of 2717 files processed.  
2035 718 of 2717 files processed.  
2034 718 of 2717 files processed.  
2037 722 of 2717 files processed.  
2036 714 of 2717 files processed.  
2035 719 of 2717 files processed.  
2034 719 of 2717 files processed.  
2037 723 of 2717 files processed.  
2036 715 of 2717 files processed.  
2035 720 of 2717 files processed.  
2034 720 of 2717 files processed.  
2037 724 of 2717 files processed.  
2036 716 of 2717 files processed.  
2035 721 of 2717 files processed.  
2034 721 of 2717 files processed.  
2037 725 of 2717 files processed.  
2034 722 of 2717 files processed.  
2035 722 of 2717 files processed.  
2037 726 of 2717 files processed.  
2036 717 of 2717 files processed.  
2035 723 of 2717 files processed.  
2034 723 of 2717 files processed.  
2037 727 of 2717 files processed.  
2036 718 of 2717 files processed.  
2035 724 of 2717 files processed.  
2034 724 of 2717 files processed.  
2037 728 of 2717 files processed.  
2034 725 of 2717 files processed.  
2036 719 of 2717 files processed.  
2035 725 of 2717 files processed.  
2037 729 of 2717 files processed.  
2034 726 of 2717 files processed.  
2036 720 of 2717 files processed.  
2035 726 of 2717 files processed.  
2034 727 of 2717 files processed.  
2037 730 of 2717 files processed.  
2035 727 of 2717 files processed.  
2036 721 of 2717 files processed.  
2034 728 of 2717 files processed.  
2037 731 of 2717 files processed.  
2035 728 of 2717 files processed.  
2037 732 of 2717 files processed.  
2034 729 of 2717 files processed.  
2036 722 of 2717 files processed.  
2037 733 of 2717 files processed.  
-----

2036 723 of 2717 files processed.  
2035 729 of 2717 files processed.  
2034 730 of 2717 files processed.  
2036 724 of 2717 files processed.  
2037 734 of 2717 files processed.  
2034 731 of 2717 files processed.  
2036 725 of 2717 files processed.  
2035 730 of 2717 files processed.  
2037 735 of 2717 files processed.  
2034 732 of 2717 files processed.  
2036 726 of 2717 files processed.  
2035 731 of 2717 files processed.  
2037 736 of 2717 files processed.  
2036 727 of 2717 files processed.  
2034 733 of 2717 files processed.  
2035 732 of 2717 files processed.  
2034 734 of 2717 files processed.  
2037 737 of 2717 files processed.  
2036 728 of 2717 files processed.  
2035 733 of 2717 files processed.  
2034 735 of 2717 files processed.  
2037 738 of 2717 files processed.  
2036 729 of 2717 files processed.  
2035 734 of 2717 files processed.  
2035 735 of 2717 files processed.  
2034 736 of 2717 files processed.  
2036 730 of 2717 files processed.  
2037 739 of 2717 files processed.  
2035 736 of 2717 files processed.  
2034 737 of 2717 files processed.  
2035 737 of 2717 files processed.  
2034 738 of 2717 files processed.  
2036 731 of 2717 files processed.  
2037 740 of 2717 files processed.  
2034 739 of 2717 files processed.  
2036 732 of 2717 files processed.  
2035 738 of 2717 files processed.  
2037 741 of 2717 files processed.  
2034 740 of 2717 files processed.  
2036 733 of 2717 files processed.  
2037 742 of 2717 files processed.  
2035 739 of 2717 files processed.  
2034 741 of 2717 files processed.  
2035 740 of 2717 files processed.  
2037 743 of 2717 files processed.  
2036 734 of 2717 files processed.  
2034 742 of 2717 files processed.  
2036 735 of 2717 files processed.  
2037 744 of 2717 files processed.  
2034 743 of 2717 files processed.  
2035 741 of 2717 files processed.  
2037 745 of 2717 files processed.  
2036 736 of 2717 files processed.  
2034 744 of 2717 files processed.  
2035 742 of 2717 files processed.  
2037 746 of 2717 files processed.  
2035 743 of 2717 files processed.  
2036 737 of 2717 files processed.  
2037 747 of 2717 files processed.  
2034 745 of 2717 files processed.  
2035 744 of 2717 files processed.  
2037 748 of 2717 files processed.  
2036 738 of 2717 files processed.  
2034 746 of 2717 files processed.  
2037 749 of 2717 files processed.  
2035 745 of 2717 files processed.  
2037 750 of 2717 files processed.  
2035 746 of 2717 files processed.  
2034 747 of 2717 files processed.  
2036 739 of 2717 files processed.  
2035 747 of 2717 files processed.  
2037 751 of 2717 files processed.  
2034 748 of 2717 files processed.  
2035 748 of 2717 files processed.  
2037 752 of 2717 files processed.  
2035 749 of 2717 files processed.  
2036 740 of 2717 files processed.

2037 753 of 2717 files processed.  
2034 749 of 2717 files processed.  
2036 741 of 2717 files processed.  
2037 754 of 2717 files processed.  
2034 750 of 2717 files processed.  
2035 750 of 2717 files processed.  
2036 742 of 2717 files processed.  
2035 751 of 2717 files processed.  
2037 755 of 2717 files processed.  
2036 743 of 2717 files processed.  
2035 752 of 2717 files processed.  
2034 751 of 2717 files processed.  
2036 744 of 2717 files processed.  
2037 756 of 2717 files processed.  
2034 752 of 2717 files processed.  
2035 753 of 2717 files processed.  
2034 753 of 2717 files processed.  
2036 745 of 2717 files processed.  
2035 754 of 2717 files processed.  
2037 757 of 2717 files processed.  
2035 755 of 2717 files processed.  
2034 754 of 2717 files processed.  
2036 746 of 2717 files processed.  
2034 755 of 2717 files processed.  
2035 756 of 2717 files processed.  
2037 758 of 2717 files processed.  
2036 747 of 2717 files processed.  
2034 756 of 2717 files processed.  
2035 757 of 2717 files processed.  
2037 759 of 2717 files processed.  
2036 748 of 2717 files processed.  
2035 758 of 2717 files processed.  
2034 757 of 2717 files processed.  
2036 749 of 2717 files processed.  
2037 760 of 2717 files processed.  
2034 758 of 2717 files processed.  
2037 761 of 2717 files processed.  
2035 759 of 2717 files processed.  
2036 750 of 2717 files processed.  
2034 759 of 2717 files processed.  
2034 760 of 2717 files processed.  
2035 760 of 2717 files processed.  
2037 762 of 2717 files processed.  
2036 751 of 2717 files processed.  
2035 761 of 2717 files processed.  
2034 761 of 2717 files processed.  
2037 763 of 2717 files processed.  
2036 752 of 2717 files processed.  
2034 762 of 2717 files processed.  
2035 762 of 2717 files processed.  
2037 764 of 2717 files processed.  
2036 753 of 2717 files processed.  
2036 754 of 2717 files processed.  
2037 765 of 2717 files processed.  
2034 763 of 2717 files processed.  
2035 763 of 2717 files processed.  
2034 764 of 2717 files processed.  
2036 755 of 2717 files processed.  
2035 764 of 2717 files processed.  
2037 766 of 2717 files processed.  
2034 765 of 2717 files processed.  
2036 756 of 2717 files processed.  
2035 765 of 2717 files processed.  
2037 767 of 2717 files processed.  
2036 757 of 2717 files processed.  
2035 766 of 2717 files processed.  
2034 766 of 2717 files processed.  
2037 768 of 2717 files processed.  
2036 758 of 2717 files processed.  
2034 767 of 2717 files processed.  
2035 767 of 2717 files processed.  
2037 769 of 2717 files processed.  
2034 768 of 2717 files processed.  
2036 759 of 2717 files processed.  
2035 768 of 2717 files processed.  
2037 770 of 2717 files processed.  
2034 769 of 2717 files processed.

2035 769 of 2717 files processed.  
2036 760 of 2717 files processed.  
2037 771 of 2717 files processed.  
2035 770 of 2717 files processed.  
2034 770 of 2717 files processed.  
2036 761 of 2717 files processed.  
2037 772 of 2717 files processed.  
2035 771 of 2717 files processed.  
2034 771 of 2717 files processed.  
2037 773 of 2717 files processed.  
2036 762 of 2717 files processed.  
2035 772 of 2717 files processed.  
2036 763 of 2717 files processed.  
2034 772 of 2717 files processed.  
2037 774 of 2717 files processed.  
2034 773 of 2717 files processed.  
2036 764 of 2717 files processed.  
2035 773 of 2717 files processed.  
2037 775 of 2717 files processed.  
2034 774 of 2717 files processed.  
2036 765 of 2717 files processed.  
2035 774 of 2717 files processed.  
2037 776 of 2717 files processed.  
2034 775 of 2717 files processed.  
2036 766 of 2717 files processed.  
2035 775 of 2717 files processed.  
2034 776 of 2717 files processed.  
2037 777 of 2717 files processed.  
2036 767 of 2717 files processed.  
2034 777 of 2717 files processed.  
2037 778 of 2717 files processed.  
2035 776 of 2717 files processed.  
2036 768 of 2717 files processed.  
2034 778 of 2717 files processed.  
2037 779 of 2717 files processed.  
2035 777 of 2717 files processed.  
2036 769 of 2717 files processed.  
2034 779 of 2717 files processed.  
2037 780 of 2717 files processed.  
2035 778 of 2717 files processed.  
2036 770 of 2717 files processed.  
2034 780 of 2717 files processed.  
2036 771 of 2717 files processed.  
2037 781 of 2717 files processed.  
2035 779 of 2717 files processed.  
2034 781 of 2717 files processed.  
2037 782 of 2717 files processed.  
2035 780 of 2717 files processed.  
2036 772 of 2717 files processed.  
2035 781 of 2717 files processed.  
2034 782 of 2717 files processed.  
2036 773 of 2717 files processed.  
2037 783 of 2717 files processed.  
2035 782 of 2717 files processed.  
2036 774 of 2717 files processed.  
2037 784 of 2717 files processed.  
2034 783 of 2717 files processed.  
2035 783 of 2717 files processed.  
2036 775 of 2717 files processed.  
2037 785 of 2717 files processed.  
2035 784 of 2717 files processed.  
2034 784 of 2717 files processed.  
2037 786 of 2717 files processed.  
2036 776 of 2717 files processed.  
2035 785 of 2717 files processed.  
2034 785 of 2717 files processed.  
2037 787 of 2717 files processed.  
2034 786 of 2717 files processed.  
2035 786 of 2717 files processed.  
2036 777 of 2717 files processed.  
2037 788 of 2717 files processed.  
2034 787 of 2717 files processed.  
2035 787 of 2717 files processed.  
2037 789 of 2717 files processed.  
2036 778 of 2717 files processed.  
2034 788 of 2717 files processed.  
2034 789 of 2717 files processed.

2035 788 of 2717 files processed.  
2037 790 of 2717 files processed.  
2036 779 of 2717 files processed.  
2036 780 of 2717 files processed.  
2034 790 of 2717 files processed.  
2035 789 of 2717 files processed.  
2037 791 of 2717 files processed.  
2036 781 of 2717 files processed.  
2034 791 of 2717 files processed.  
2035 790 of 2717 files processed.  
2037 792 of 2717 files processed.  
2036 782 of 2717 files processed.  
2035 791 of 2717 files processed.  
2034 792 of 2717 files processed.  
2037 793 of 2717 files processed.  
2036 783 of 2717 files processed.  
2037 794 of 2717 files processed.  
2035 792 of 2717 files processed.  
2036 784 of 2717 files processed.  
2034 793 of 2717 files processed.  
2037 795 of 2717 files processed.  
2036 785 of 2717 files processed.  
2035 793 of 2717 files processed.  
2037 796 of 2717 files processed.  
2034 794 of 2717 files processed.  
2036 786 of 2717 files processed.  
2035 794 of 2717 files processed.  
2037 797 of 2717 files processed.  
2034 795 of 2717 files processed.  
2036 787 of 2717 files processed.  
2037 798 of 2717 files processed.  
2034 796 of 2717 files processed.  
2037 799 of 2717 files processed.  
2035 795 of 2717 files processed.  
2036 788 of 2717 files processed.  
2037 800 of 2717 files processed.  
2034 797 of 2717 files processed.  
2035 796 of 2717 files processed.  
2036 789 of 2717 files processed.  
2037 801 of 2717 files processed.  
2034 798 of 2717 files processed.  
2035 797 of 2717 files processed.  
2037 802 of 2717 files processed.  
2036 790 of 2717 files processed.  
2034 799 of 2717 files processed.  
2035 798 of 2717 files processed.  
2037 803 of 2717 files processed.  
2034 800 of 2717 files processed.  
2037 804 of 2717 files processed.  
2036 791 of 2717 files processed.  
2035 799 of 2717 files processed.  
2034 801 of 2717 files processed.  
2037 805 of 2717 files processed.  
2036 792 of 2717 files processed.  
2034 802 of 2717 files processed.  
2037 806 of 2717 files processed.  
2036 793 of 2717 files processed.  
2037 807 of 2717 files processed.  
2035 800 of 2717 files processed.  
2034 803 of 2717 files processed.  
2035 801 of 2717 files processed.  
2036 794 of 2717 files processed.  
2037 808 of 2717 files processed.  
2034 804 of 2717 files processed.  
2036 795 of 2717 files processed.  
2035 802 of 2717 files processed.  
2034 805 of 2717 files processed.  
2037 809 of 2717 files processed.  
2036 796 of 2717 files processed.  
2035 803 of 2717 files processed.  
2037 810 of 2717 files processed.  
2036 797 of 2717 files processed.  
2034 806 of 2717 files processed.  
2035 804 of 2717 files processed.  
2037 811 of 2717 files processed.  
2036 798 of 2717 files processed.  
2034 807 of 2717 files processed.

2035 805 of 2717 files processed.  
2037 812 of 2717 files processed.  
2034 808 of 2717 files processed.  
2036 799 of 2717 files processed.  
2035 806 of 2717 files processed.  
2037 813 of 2717 files processed.  
2034 809 of 2717 files processed.  
2036 800 of 2717 files processed.  
2035 807 of 2717 files processed.  
2037 814 of 2717 files processed.  
2034 810 of 2717 files processed.  
2036 801 of 2717 files processed.  
2035 808 of 2717 files processed.  
2037 815 of 2717 files processed.  
2034 811 of 2717 files processed.  
2036 802 of 2717 files processed.  
2035 809 of 2717 files processed.  
2037 816 of 2717 files processed.  
2034 812 of 2717 files processed.  
2036 803 of 2717 files processed.  
2037 817 of 2717 files processed.  
2035 810 of 2717 files processed.  
2036 804 of 2717 files processed.  
2034 813 of 2717 files processed.  
2037 818 of 2717 files processed.  
2035 811 of 2717 files processed.  
2036 805 of 2717 files processed.  
2034 814 of 2717 files processed.  
2037 819 of 2717 files processed.  
2036 806 of 2717 files processed.  
2034 815 of 2717 files processed.  
2035 812 of 2717 files processed.  
2037 820 of 2717 files processed.  
2034 816 of 2717 files processed.  
2035 813 of 2717 files processed.  
2036 807 of 2717 files processed.  
2034 817 of 2717 files processed.  
2037 821 of 2717 files processed.  
2035 814 of 2717 files processed.  
2034 818 of 2717 files processed.  
2036 808 of 2717 files processed.  
2037 822 of 2717 files processed.  
2034 819 of 2717 files processed.  
2035 815 of 2717 files processed.  
2034 820 of 2717 files processed.  
2036 809 of 2717 files processed.  
2035 816 of 2717 files processed.  
2034 821 of 2717 files processed.  
2037 823 of 2717 files processed.  
2035 817 of 2717 files processed.  
2036 810 of 2717 files processed.  
2034 822 of 2717 files processed.  
2035 818 of 2717 files processed.  
2036 811 of 2717 files processed.  
2037 824 of 2717 files processed.  
2034 823 of 2717 files processed.  
2036 812 of 2717 files processed.  
2035 819 of 2717 files processed.  
2034 824 of 2717 files processed.  
2037 825 of 2717 files processed.  
2035 820 of 2717 files processed.  
2036 813 of 2717 files processed.  
2034 825 of 2717 files processed.  
2037 826 of 2717 files processed.  
2035 821 of 2717 files processed.  
2036 814 of 2717 files processed.  
2037 827 of 2717 files processed.  
2035 822 of 2717 files processed.  
2034 826 of 2717 files processed.  
2036 815 of 2717 files processed.  
2034 827 of 2717 files processed.  
2035 823 of 2717 files processed.  
2037 828 of 2717 files processed.  
2036 816 of 2717 files processed.  
2034 828 of 2717 files processed.  
2037 829 of 2717 files processed.  
2035 824 of 2717 files processed.

2036 817 of 2717 files processed.  
2034 829 of 2717 files processed.  
2035 825 of 2717 files processed.  
2034 830 of 2717 files processed.  
2036 818 of 2717 files processed.  
2037 830 of 2717 files processed.  
2034 831 of 2717 files processed.  
2035 826 of 2717 files processed.  
2036 819 of 2717 files processed.  
2037 831 of 2717 files processed.  
2034 832 of 2717 files processed.  
2037 832 of 2717 files processed.  
2036 820 of 2717 files processed.  
2035 827 of 2717 files processed.  
2034 833 of 2717 files processed.  
2036 821 of 2717 files processed.  
2035 828 of 2717 files processed.  
2037 833 of 2717 files processed.  
2034 834 of 2717 files processed.  
2036 822 of 2717 files processed.  
2035 829 of 2717 files processed.  
2037 834 of 2717 files processed.  
2034 835 of 2717 files processed.  
2036 823 of 2717 files processed.  
2035 830 of 2717 files processed.  
2037 835 of 2717 files processed.  
2034 836 of 2717 files processed.  
2035 831 of 2717 files processed.  
2036 824 of 2717 files processed.  
2037 836 of 2717 files processed.  
2036 825 of 2717 files processed.  
2034 837 of 2717 files processed.  
2035 832 of 2717 files processed.  
2036 826 of 2717 files processed.  
2034 838 of 2717 files processed.  
2037 837 of 2717 files processed.  
2035 833 of 2717 files processed.  
2034 839 of 2717 files processed.  
2036 827 of 2717 files processed.  
2037 838 of 2717 files processed.  
2035 834 of 2717 files processed.  
2034 840 of 2717 files processed.  
2037 839 of 2717 files processed.  
2036 828 of 2717 files processed.  
2035 835 of 2717 files processed.  
2034 841 of 2717 files processed.  
2036 829 of 2717 files processed.  
2037 840 of 2717 files processed.  
2035 836 of 2717 files processed.  
2036 830 of 2717 files processed.  
2034 842 of 2717 files processed.  
2035 837 of 2717 files processed.  
2037 841 of 2717 files processed.  
2034 843 of 2717 files processed.  
2036 831 of 2717 files processed.  
2035 838 of 2717 files processed.  
2037 842 of 2717 files processed.  
2034 844 of 2717 files processed.  
2036 832 of 2717 files processed.  
2035 839 of 2717 files processed.  
2037 843 of 2717 files processed.  
2034 845 of 2717 files processed.  
2035 840 of 2717 files processed.  
2037 844 of 2717 files processed.  
2036 833 of 2717 files processed.  
2034 846 of 2717 files processed.  
2036 834 of 2717 files processed.  
2035 841 of 2717 files processed.  
2037 845 of 2717 files processed.  
2035 842 of 2717 files processed.  
2036 835 of 2717 files processed.  
2037 846 of 2717 files processed.  
2034 847 of 2717 files processed.  
2035 843 of 2717 files processed.  
2036 836 of 2717 files processed.  
2037 847 of 2717 files processed.  
2034 848 of 2717 files processed.

2035 844 of 2717 files processed.  
2037 848 of 2717 files processed.  
2036 837 of 2717 files processed.  
2034 849 of 2717 files processed.  
2037 849 of 2717 files processed.  
2035 845 of 2717 files processed.  
2036 838 of 2717 files processed.  
2034 850 of 2717 files processed.  
2035 846 of 2717 files processed.  
2036 839 of 2717 files processed.  
2034 851 of 2717 files processed.  
2037 850 of 2717 files processed.  
2035 847 of 2717 files processed.  
2034 852 of 2717 files processed.  
2037 851 of 2717 files processed.  
2036 840 of 2717 files processed.  
2037 852 of 2717 files processed.  
2035 848 of 2717 files processed.  
2034 853 of 2717 files processed.  
2036 841 of 2717 files processed.  
2037 853 of 2717 files processed.  
2035 849 of 2717 files processed.  
2036 842 of 2717 files processed.  
2034 854 of 2717 files processed.  
2035 850 of 2717 files processed.  
2034 855 of 2717 files processed.  
2037 854 of 2717 files processed.  
2036 843 of 2717 files processed.  
2037 855 of 2717 files processed.  
2035 851 of 2717 files processed.  
2036 844 of 2717 files processed.  
2035 852 of 2717 files processed.  
2034 856 of 2717 files processed.  
2036 845 of 2717 files processed.  
2037 856 of 2717 files processed.  
2035 853 of 2717 files processed.  
2036 846 of 2717 files processed.  
2037 857 of 2717 files processed.  
2034 857 of 2717 files processed.  
2035 854 of 2717 files processed.  
2037 858 of 2717 files processed.  
2034 858 of 2717 files processed.  
2036 847 of 2717 files processed.  
2037 859 of 2717 files processed.  
2036 848 of 2717 files processed.  
2035 855 of 2717 files processed.  
2034 859 of 2717 files processed.  
2035 856 of 2717 files processed.  
2037 860 of 2717 files processed.  
2036 849 of 2717 files processed.  
2034 860 of 2717 files processed.  
2036 850 of 2717 files processed.  
2034 861 of 2717 files processed.  
2037 861 of 2717 files processed.  
2035 857 of 2717 files processed.  
2037 862 of 2717 files processed.  
2036 851 of 2717 files processed.  
2035 858 of 2717 files processed.  
2034 862 of 2717 files processed.  
2036 852 of 2717 files processed.  
2037 863 of 2717 files processed.  
2034 863 of 2717 files processed.  
2035 859 of 2717 files processed.  
2036 853 of 2717 files processed.  
2037 864 of 2717 files processed.  
2035 860 of 2717 files processed.  
2034 864 of 2717 files processed.  
2037 865 of 2717 files processed.  
2035 861 of 2717 files processed.  
2036 854 of 2717 files processed.  
2034 865 of 2717 files processed.  
2037 866 of 2717 files processed.  
2035 862 of 2717 files processed.  
2036 855 of 2717 files processed.  
2037 867 of 2717 files processed.  
2034 866 of 2717 files processed.  
2035 863 of 2717 files processed.

2036 856 of 2717 files processed.  
2037 868 of 2717 files processed.  
2034 867 of 2717 files processed.  
2037 869 of 2717 files processed.  
2036 857 of 2717 files processed.  
2035 864 of 2717 files processed.  
2034 868 of 2717 files processed.  
2037 870 of 2717 files processed.  
2034 869 of 2717 files processed.  
2036 858 of 2717 files processed.  
2035 865 of 2717 files processed.  
2034 870 of 2717 files processed.  
2037 871 of 2717 files processed.  
2036 859 of 2717 files processed.  
2035 866 of 2717 files processed.  
2034 871 of 2717 files processed.  
2037 872 of 2717 files processed.  
2036 860 of 2717 files processed.  
2035 867 of 2717 files processed.  
2034 872 of 2717 files processed.  
2036 861 of 2717 files processed.  
2037 873 of 2717 files processed.  
2036 862 of 2717 files processed.  
2035 868 of 2717 files processed.  
2034 873 of 2717 files processed.  
2037 874 of 2717 files processed.  
2036 863 of 2717 files processed.  
2037 875 of 2717 files processed.  
2035 869 of 2717 files processed.  
2034 874 of 2717 files processed.  
2036 864 of 2717 files processed.  
2037 876 of 2717 files processed.  
2035 870 of 2717 files processed.  
2034 875 of 2717 files processed.  
2036 865 of 2717 files processed.  
2037 877 of 2717 files processed.  
2035 871 of 2717 files processed.  
2036 866 of 2717 files processed.  
2034 876 of 2717 files processed.  
2035 872 of 2717 files processed.  
2036 867 of 2717 files processed.  
2037 878 of 2717 files processed.  
2036 868 of 2717 files processed.  
2035 873 of 2717 files processed.  
2034 877 of 2717 files processed.  
2037 879 of 2717 files processed.  
2036 869 of 2717 files processed.  
2034 878 of 2717 files processed.  
2035 874 of 2717 files processed.  
2037 880 of 2717 files processed.  
2035 875 of 2717 files processed.  
2034 879 of 2717 files processed.  
2036 870 of 2717 files processed.  
2037 881 of 2717 files processed.  
2035 876 of 2717 files processed.  
2034 880 of 2717 files processed.  
2036 871 of 2717 files processed.  
2037 882 of 2717 files processed.  
2037 883 of 2717 files processed.  
2035 877 of 2717 files processed.  
2036 872 of 2717 files processed.  
2034 881 of 2717 files processed.  
2037 884 of 2717 files processed.  
2035 878 of 2717 files processed.  
2036 873 of 2717 files processed.  
2034 882 of 2717 files processed.  
2035 879 of 2717 files processed.  
2037 885 of 2717 files processed.  
2036 874 of 2717 files processed.  
2035 880 of 2717 files processed.  
2037 886 of 2717 files processed.  
2034 883 of 2717 files processed.  
2035 881 of 2717 files processed.  
2036 875 of 2717 files processed.  
2037 887 of 2717 files processed.  
2034 884 of 2717 files processed.  
2035 882 of 2717 files processed.

2036 876 of 2717 files processed.  
2034 885 of 2717 files processed.  
2037 888 of 2717 files processed.  
2035 883 of 2717 files processed.  
2036 877 of 2717 files processed.  
2034 886 of 2717 files processed.  
2037 889 of 2717 files processed.  
2036 878 of 2717 files processed.  
2037 890 of 2717 files processed.  
2035 884 of 2717 files processed.  
2034 887 of 2717 files processed.  
2037 891 of 2717 files processed.  
2036 879 of 2717 files processed.  
2035 885 of 2717 files processed.  
2034 888 of 2717 files processed.  
2037 892 of 2717 files processed.  
2036 880 of 2717 files processed.  
2035 886 of 2717 files processed.  
2037 893 of 2717 files processed.  
2036 881 of 2717 files processed.  
2034 889 of 2717 files processed.  
2035 887 of 2717 files processed.  
2034 890 of 2717 files processed.  
2036 882 of 2717 files processed.  
2037 894 of 2717 files processed.  
2034 891 of 2717 files processed.  
2035 888 of 2717 files processed.  
2036 883 of 2717 files processed.  
2037 895 of 2717 files processed.  
2034 892 of 2717 files processed.  
2035 889 of 2717 files processed.  
2036 884 of 2717 files processed.  
2037 896 of 2717 files processed.  
2034 893 of 2717 files processed.  
2037 897 of 2717 files processed.  
2036 885 of 2717 files processed.  
2035 890 of 2717 files processed.  
2034 894 of 2717 files processed.  
2037 898 of 2717 files processed.  
2036 886 of 2717 files processed.  
2034 895 of 2717 files processed.  
2035 891 of 2717 files processed.  
2036 887 of 2717 files processed.  
2034 896 of 2717 files processed.  
2037 899 of 2717 files processed.  
2035 892 of 2717 files processed.  
2036 888 of 2717 files processed.  
2037 900 of 2717 files processed.  
2034 897 of 2717 files processed.  
2035 893 of 2717 files processed.  
2037 901 of 2717 files processed.  
2034 898 of 2717 files processed.  
2036 889 of 2717 files processed.  
2037 902 of 2717 files processed.  
2035 894 of 2717 files processed.  
2034 899 of 2717 files processed.  
2036 890 of 2717 files processed.  
2037 903 of 2717 files processed.  
2035 895 of 2717 files processed.  
2034 900 of 2717 files processed.  
2036 891 of 2717 files processed.  
2037 904 of 2717 files processed.  
2035 896 of 2717 files processed.  
2034 901 of 2717 files processed.  
2036 892 of 2717 files processed.  
2035 897 of 2717 files processed.  
2036 893 of 2717 files processed.  
2034 902 of 2717 files processed.  
2037 905 of 2717 files processed.  
2035 898 of 2717 files processed.  
2036 894 of 2717 files processed.  
2034 903 of 2717 files processed.  
2037 906 of 2717 files processed.  
2035 899 of 2717 files processed.  
2036 895 of 2717 files processed.  
2034 904 of 2717 files processed.  
2035 900 of 2717 files processed.

-----  
2037 907 of 2717 files processed.  
2036 896 of 2717 files processed.  
2034 905 of 2717 files processed.  
2035 901 of 2717 files processed.  
2037 908 of 2717 files processed.  
2034 906 of 2717 files processed.  
2035 902 of 2717 files processed.  
2036 897 of 2717 files processed.  
2034 907 of 2717 files processed.  
2037 909 of 2717 files processed.  
2036 898 of 2717 files processed.  
2034 908 of 2717 files processed.  
2035 903 of 2717 files processed.  
2037 910 of 2717 files processed.  
2036 899 of 2717 files processed.  
2035 904 of 2717 files processed.  
2034 909 of 2717 files processed.  
2037 911 of 2717 files processed.  
2036 900 of 2717 files processed.  
2035 905 of 2717 files processed.  
2034 910 of 2717 files processed.  
2035 906 of 2717 files processed.  
2037 912 of 2717 files processed.  
2036 901 of 2717 files processed.  
2037 913 of 2717 files processed.  
2034 911 of 2717 files processed.  
2036 902 of 2717 files processed.  
2035 907 of 2717 files processed.  
2034 912 of 2717 files processed.  
2037 914 of 2717 files processed.  
2034 913 of 2717 files processed.  
2036 903 of 2717 files processed.  
2037 915 of 2717 files processed.  
2035 908 of 2717 files processed.  
2034 914 of 2717 files processed.  
2036 904 of 2717 files processed.  
2037 916 of 2717 files processed.  
2035 909 of 2717 files processed.  
2036 905 of 2717 files processed.  
2034 915 of 2717 files processed.  
2035 910 of 2717 files processed.  
2037 917 of 2717 files processed.  
2036 906 of 2717 files processed.  
2034 916 of 2717 files processed.  
2037 918 of 2717 files processed.  
2035 911 of 2717 files processed.  
2036 907 of 2717 files processed.  
2034 917 of 2717 files processed.  
2037 919 of 2717 files processed.  
2036 908 of 2717 files processed.  
2035 912 of 2717 files processed.  
2034 918 of 2717 files processed.  
2036 909 of 2717 files processed.  
2035 913 of 2717 files processed.  
2037 920 of 2717 files processed.  
2034 919 of 2717 files processed.  
2036 910 of 2717 files processed.  
2034 920 of 2717 files processed.  
2037 921 of 2717 files processed.  
2035 914 of 2717 files processed.  
2036 911 of 2717 files processed.  
2034 921 of 2717 files processed.  
2035 915 of 2717 files processed.  
2036 912 of 2717 files processed.  
2037 922 of 2717 files processed.  
2034 922 of 2717 files processed.  
2035 916 of 2717 files processed.  
2036 913 of 2717 files processed.  
2037 923 of 2717 files processed.  
2034 923 of 2717 files processed.  
2035 917 of 2717 files processed.  
2036 914 of 2717 files processed.  
2037 924 of 2717 files processed.  
2036 915 of 2717 files processed.  
2034 924 of 2717 files processed.  
2035 918 of 2717 files processed.  
2037 925 of 2717 files processed.

.....  
2036 916 of 2717 files processed.  
2035 919 of 2717 files processed.  
2037 926 of 2717 files processed.  
2034 925 of 2717 files processed.  
2036 917 of 2717 files processed.  
2035 920 of 2717 files processed.  
2034 926 of 2717 files processed.  
2037 927 of 2717 files processed.  
2036 918 of 2717 files processed.  
2034 927 of 2717 files processed.  
2035 921 of 2717 files processed.  
2037 928 of 2717 files processed.  
2036 919 of 2717 files processed.  
2034 928 of 2717 files processed.  
2036 920 of 2717 files processed.  
2037 929 of 2717 files processed.  
2035 922 of 2717 files processed.  
2034 929 of 2717 files processed.  
2035 923 of 2717 files processed.  
2037 930 of 2717 files processed.  
2034 930 of 2717 files processed.  
2036 921 of 2717 files processed.  
2037 931 of 2717 files processed.  
2034 931 of 2717 files processed.  
2035 924 of 2717 files processed.  
2036 922 of 2717 files processed.  
2034 932 of 2717 files processed.  
2037 932 of 2717 files processed.  
2035 925 of 2717 files processed.  
2036 923 of 2717 files processed.  
2034 933 of 2717 files processed.  
2037 933 of 2717 files processed.  
2036 924 of 2717 files processed.  
2035 926 of 2717 files processed.  
2034 934 of 2717 files processed.  
2037 934 of 2717 files processed.  
2036 925 of 2717 files processed.  
2035 927 of 2717 files processed.  
2034 935 of 2717 files processed.  
2037 935 of 2717 files processed.  
2035 928 of 2717 files processed.  
2036 926 of 2717 files processed.  
2034 936 of 2717 files processed.  
2035 929 of 2717 files processed.  
2036 927 of 2717 files processed.  
2037 936 of 2717 files processed.  
2034 937 of 2717 files processed.  
2035 930 of 2717 files processed.  
2037 937 of 2717 files processed.  
2036 928 of 2717 files processed.  
2034 938 of 2717 files processed.  
2035 931 of 2717 files processed.  
2037 938 of 2717 files processed.  
2036 929 of 2717 files processed.  
2035 932 of 2717 files processed.  
2034 939 of 2717 files processed.  
2036 930 of 2717 files processed.  
2037 939 of 2717 files processed.  
2035 933 of 2717 files processed.  
2036 931 of 2717 files processed.  
2034 940 of 2717 files processed.  
2035 934 of 2717 files processed.  
2037 940 of 2717 files processed.  
2036 932 of 2717 files processed.  
2034 941 of 2717 files processed.  
2035 935 of 2717 files processed.  
2037 941 of 2717 files processed.  
2036 933 of 2717 files processed.  
2034 942 of 2717 files processed.  
2035 936 of 2717 files processed.  
2037 942 of 2717 files processed.  
2035 937 of 2717 files processed.  
2034 943 of 2717 files processed.  
2036 934 of 2717 files processed.  
2037 943 of 2717 files processed.  
2036 935 of 2717 files processed.  
2035 938 of 2717 files processed

2035 950 of 2717 files processed.  
2034 944 of 2717 files processed.  
2037 944 of 2717 files processed.  
2036 936 of 2717 files processed.  
2034 945 of 2717 files processed.  
2037 945 of 2717 files processed.  
2035 939 of 2717 files processed.  
2036 937 of 2717 files processed.  
2037 946 of 2717 files processed.  
2035 940 of 2717 files processed.  
2034 946 of 2717 files processed.  
2036 938 of 2717 files processed.  
2035 941 of 2717 files processed.  
2037 947 of 2717 files processed.  
2036 939 of 2717 files processed.  
2034 947 of 2717 files processed.  
2037 948 of 2717 files processed.  
2035 942 of 2717 files processed.  
2036 940 of 2717 files processed.  
2034 948 of 2717 files processed.  
2037 949 of 2717 files processed.  
2036 941 of 2717 files processed.  
2035 943 of 2717 files processed.  
2037 950 of 2717 files processed.  
2034 949 of 2717 files processed.  
2035 944 of 2717 files processed.  
2036 942 of 2717 files processed.  
2037 951 of 2717 files processed.  
2034 950 of 2717 files processed.  
2035 945 of 2717 files processed.  
2036 943 of 2717 files processed.  
2037 952 of 2717 files processed.  
2034 951 of 2717 files processed.  
2035 946 of 2717 files processed.  
2037 953 of 2717 files processed.  
2036 944 of 2717 files processed.  
2034 952 of 2717 files processed.  
2037 954 of 2717 files processed.  
2035 947 of 2717 files processed.  
2036 945 of 2717 files processed.  
2034 953 of 2717 files processed.  
2037 955 of 2717 files processed.  
2036 946 of 2717 files processed.  
2034 954 of 2717 files processed.  
2035 948 of 2717 files processed.  
2037 956 of 2717 files processed.  
2036 947 of 2717 files processed.  
2035 949 of 2717 files processed.  
2037 957 of 2717 files processed.  
2034 955 of 2717 files processed.  
2036 948 of 2717 files processed.  
2034 956 of 2717 files processed.  
2037 958 of 2717 files processed.  
2035 950 of 2717 files processed.  
2036 949 of 2717 files processed.  
2035 951 of 2717 files processed.  
2034 957 of 2717 files processed.  
2037 959 of 2717 files processed.  
2036 950 of 2717 files processed.  
2034 958 of 2717 files processed.  
2035 952 of 2717 files processed.  
2037 960 of 2717 files processed.  
2036 951 of 2717 files processed.  
2034 959 of 2717 files processed.  
2035 953 of 2717 files processed.  
2037 961 of 2717 files processed.  
2036 952 of 2717 files processed.  
2034 960 of 2717 files processed.  
2036 953 of 2717 files processed.  
2035 954 of 2717 files processed.  
2037 962 of 2717 files processed.  
2034 961 of 2717 files processed.  
2035 955 of 2717 files processed.  
2037 963 of 2717 files processed.  
2036 954 of 2717 files processed.  
2035 956 of 2717 files processed.  
2036 955 of 2717 files processed.  
2034 962 of 2717 files processed.

2034 962 of 2717 files processed.  
2037 964 of 2717 files processed.  
2035 957 of 2717 files processed.  
2037 965 of 2717 files processed.  
2036 956 of 2717 files processed.  
2035 958 of 2717 files processed.  
2034 963 of 2717 files processed.  
2037 966 of 2717 files processed.  
2036 957 of 2717 files processed.  
2034 964 of 2717 files processed.  
2036 958 of 2717 files processed.  
2035 959 of 2717 files processed.  
2037 967 of 2717 files processed.  
2036 959 of 2717 files processed.  
2034 965 of 2717 files processed.  
2035 960 of 2717 files processed.  
2036 960 of 2717 files processed.  
2037 968 of 2717 files processed.  
2034 966 of 2717 files processed.  
2035 961 of 2717 files processed.  
2036 961 of 2717 files processed.  
2034 967 of 2717 files processed.  
2037 969 of 2717 files processed.  
2035 962 of 2717 files processed.  
2036 962 of 2717 files processed.  
2034 968 of 2717 files processed.  
2037 970 of 2717 files processed.  
2036 963 of 2717 files processed.  
2035 963 of 2717 files processed.  
2034 969 of 2717 files processed.  
2036 964 of 2717 files processed.  
2037 971 of 2717 files processed.  
2034 970 of 2717 files processed.  
2035 964 of 2717 files processed.  
2036 965 of 2717 files processed.  
2037 972 of 2717 files processed.  
2034 971 of 2717 files processed.  
2035 965 of 2717 files processed.  
2036 966 of 2717 files processed.  
2034 972 of 2717 files processed.  
2037 973 of 2717 files processed.  
2035 966 of 2717 files processed.  
2036 967 of 2717 files processed.  
2035 967 of 2717 files processed.  
2037 974 of 2717 files processed.  
2034 973 of 2717 files processed.  
2036 968 of 2717 files processed.  
2037 975 of 2717 files processed.  
2034 974 of 2717 files processed.  
2036 969 of 2717 files processed.  
2035 968 of 2717 files processed.  
2037 976 of 2717 files processed.  
2034 975 of 2717 files processed.  
2036 970 of 2717 files processed.  
2035 969 of 2717 files processed.  
2034 976 of 2717 files processed.  
2036 971 of 2717 files processed.  
2037 977 of 2717 files processed.  
2035 970 of 2717 files processed.  
2036 972 of 2717 files processed.  
2037 978 of 2717 files processed.  
2034 977 of 2717 files processed.  
2035 971 of 2717 files processed.  
2036 973 of 2717 files processed.  
2037 979 of 2717 files processed.  
2034 978 of 2717 files processed.  
2035 972 of 2717 files processed.  
2036 974 of 2717 files processed.  
2037 980 of 2717 files processed.  
2035 973 of 2717 files processed.  
2034 979 of 2717 files processed.  
2036 975 of 2717 files processed.  
2037 981 of 2717 files processed.  
2035 974 of 2717 files processed.  
2034 980 of 2717 files processed.  
2036 976 of 2717 files processed.  
2037 982 of 2717 files processed.  
2036 977 of 2717 files processed.

2030 911 of 2717 files processed.  
2035 975 of 2717 files processed.  
2034 981 of 2717 files processed.  
2037 983 of 2717 files processed.  
2034 982 of 2717 files processed.  
2036 978 of 2717 files processed.  
2035 976 of 2717 files processed.  
2037 984 of 2717 files processed.  
2036 979 of 2717 files processed.  
2034 983 of 2717 files processed.  
2035 977 of 2717 files processed.  
2037 985 of 2717 files processed.  
2034 984 of 2717 files processed.  
2036 980 of 2717 files processed.  
2035 978 of 2717 files processed.  
2034 985 of 2717 files processed.  
2036 981 of 2717 files processed.  
2037 986 of 2717 files processed.  
2036 982 of 2717 files processed.  
2034 986 of 2717 files processed.  
2035 979 of 2717 files processed.  
2037 987 of 2717 files processed.  
2036 983 of 2717 files processed.  
2034 987 of 2717 files processed.  
2035 980 of 2717 files processed.  
2037 988 of 2717 files processed.  
2036 984 of 2717 files processed.  
2034 988 of 2717 files processed.  
2037 989 of 2717 files processed.  
2035 981 of 2717 files processed.  
2036 985 of 2717 files processed.  
2034 989 of 2717 files processed.  
2035 982 of 2717 files processed.  
2037 990 of 2717 files processed.  
2036 986 of 2717 files processed.  
2037 991 of 2717 files processed.  
2034 990 of 2717 files processed.  
2035 983 of 2717 files processed.  
2036 987 of 2717 files processed.  
2034 991 of 2717 files processed.  
2035 984 of 2717 files processed.  
2037 992 of 2717 files processed.  
2036 988 of 2717 files processed.  
2034 992 of 2717 files processed.  
2035 985 of 2717 files processed.  
2037 993 of 2717 files processed.  
2036 989 of 2717 files processed.  
2034 993 of 2717 files processed.  
2035 986 of 2717 files processed.  
2037 994 of 2717 files processed.  
2034 994 of 2717 files processed.  
2035 987 of 2717 files processed.  
2036 990 of 2717 files processed.  
2034 995 of 2717 files processed.  
2037 995 of 2717 files processed.  
2035 988 of 2717 files processed.  
2036 991 of 2717 files processed.  
2034 996 of 2717 files processed.  
2037 996 of 2717 files processed.  
2035 989 of 2717 files processed.  
2034 997 of 2717 files processed.  
2036 992 of 2717 files processed.  
2035 990 of 2717 files processed.  
2037 997 of 2717 files processed.  
2036 993 of 2717 files processed.  
2034 998 of 2717 files processed.  
2035 991 of 2717 files processed.  
2037 998 of 2717 files processed.  
2034 999 of 2717 files processed.  
2036 994 of 2717 files processed.  
2035 992 of 2717 files processed.  
2037 999 of 2717 files processed.  
2036 995 of 2717 files processed.  
2034 1000 of 2717 files processed.  
2037 1000 of 2717 files processed.  
2035 993 of 2717 files processed.  
2036 996 of 2717 files processed.

2034 1001 of 2717 files processed.  
2035 994 of 2717 files processed.  
2037 1001 of 2717 files processed.  
2036 997 of 2717 files processed.  
2037 1002 of 2717 files processed.  
2035 995 of 2717 files processed.  
2034 1002 of 2717 files processed.  
2036 998 of 2717 files processed.  
2037 1003 of 2717 files processed.  
2035 996 of 2717 files processed.  
2036 999 of 2717 files processed.  
2034 1003 of 2717 files processed.  
2037 1004 of 2717 files processed.  
2035 997 of 2717 files processed.  
2036 1000 of 2717 files processed.  
2034 1004 of 2717 files processed.  
2035 998 of 2717 files processed.  
2037 1005 of 2717 files processed.  
2037 1006 of 2717 files processed.  
2034 1005 of 2717 files processed.  
2036 1001 of 2717 files processed.  
2035 999 of 2717 files processed.  
2037 1007 of 2717 files processed.  
2036 1002 of 2717 files processed.  
2034 1006 of 2717 files processed.  
2037 1008 of 2717 files processed.  
2035 1000 of 2717 files processed.  
2034 1007 of 2717 files processed.  
2036 1003 of 2717 files processed.  
2037 1009 of 2717 files processed.  
2035 1001 of 2717 files processed.  
2034 1008 of 2717 files processed.  
2036 1004 of 2717 files processed.  
2037 1010 of 2717 files processed.  
2035 1002 of 2717 files processed.  
2034 1009 of 2717 files processed.  
2037 1011 of 2717 files processed.  
2035 1003 of 2717 files processed.  
2034 1010 of 2717 files processed.  
2036 1005 of 2717 files processed.  
2037 1012 of 2717 files processed.  
2035 1004 of 2717 files processed.  
2034 1011 of 2717 files processed.  
2035 1005 of 2717 files processed.  
2037 1013 of 2717 files processed.  
2036 1006 of 2717 files processed.  
2034 1012 of 2717 files processed.  
2036 1007 of 2717 files processed.  
2035 1006 of 2717 files processed.  
2037 1014 of 2717 files processed.  
2034 1013 of 2717 files processed.  
2037 1015 of 2717 files processed.  
2036 1008 of 2717 files processed.  
2035 1007 of 2717 files processed.  
2034 1014 of 2717 files processed.  
2036 1009 of 2717 files processed.  
2037 1016 of 2717 files processed.  
2034 1015 of 2717 files processed.  
2035 1008 of 2717 files processed.  
2036 1010 of 2717 files processed.  
2034 1016 of 2717 files processed.  
2037 1017 of 2717 files processed.  
2035 1009 of 2717 files processed.  
2036 1011 of 2717 files processed.  
2034 1017 of 2717 files processed.  
2037 1018 of 2717 files processed.  
2036 1012 of 2717 files processed.  
2035 1010 of 2717 files processed.  
2034 1018 of 2717 files processed.  
2035 1011 of 2717 files processed.  
2036 1013 of 2717 files processed.  
2037 1019 of 2717 files processed.  
2034 1019 of 2717 files processed.  
2036 1014 of 2717 files processed.  
2035 1012 of 2717 files processed.  
2037 1020 of 2717 files processed.  
2034 1020 of 2717 files processed.

2036 1015 of 2717 files processed.  
2037 1021 of 2717 files processed.  
2035 1013 of 2717 files processed.  
2034 1021 of 2717 files processed.  
2035 1014 of 2717 files processed.  
2037 1022 of 2717 files processed.  
2036 1016 of 2717 files processed.  
2034 1022 of 2717 files processed.  
2035 1015 of 2717 files processed.  
2037 1023 of 2717 files processed.  
2034 1023 of 2717 files processed.  
2036 1017 of 2717 files processed.  
2035 1016 of 2717 files processed.  
2037 1024 of 2717 files processed.  
2034 1024 of 2717 files processed.  
2036 1018 of 2717 files processed.  
2037 1025 of 2717 files processed.  
2035 1017 of 2717 files processed.  
2034 1025 of 2717 files processed.  
2036 1019 of 2717 files processed.  
2037 1026 of 2717 files processed.  
2035 1018 of 2717 files processed.  
2034 1026 of 2717 files processed.  
2037 1027 of 2717 files processed.  
2036 1020 of 2717 files processed.  
2035 1019 of 2717 files processed.  
2034 1027 of 2717 files processed.  
2037 1028 of 2717 files processed.  
2036 1021 of 2717 files processed.  
2035 1020 of 2717 files processed.  
2037 1029 of 2717 files processed.  
2034 1028 of 2717 files processed.  
2036 1022 of 2717 files processed.  
2035 1021 of 2717 files processed.  
2037 1030 of 2717 files processed.  
2034 1029 of 2717 files processed.  
2036 1023 of 2717 files processed.  
2035 1022 of 2717 files processed.  
2037 1031 of 2717 files processed.  
2036 1024 of 2717 files processed.  
2037 1032 of 2717 files processed.  
2035 1023 of 2717 files processed.  
2034 1030 of 2717 files processed.  
2035 1024 of 2717 files processed.  
2036 1025 of 2717 files processed.  
2037 1033 of 2717 files processed.  
2034 1031 of 2717 files processed.  
2035 1025 of 2717 files processed.  
2036 1026 of 2717 files processed.  
2037 1034 of 2717 files processed.  
2034 1032 of 2717 files processed.  
2035 1026 of 2717 files processed.  
2037 1035 of 2717 files processed.  
2034 1033 of 2717 files processed.  
2036 1027 of 2717 files processed.  
2035 1027 of 2717 files processed.  
2037 1036 of 2717 files processed.  
2034 1034 of 2717 files processed.  
2036 1028 of 2717 files processed.  
2037 1037 of 2717 files processed.  
2035 1028 of 2717 files processed.  
2036 1029 of 2717 files processed.  
2035 1029 of 2717 files processed.  
2034 1035 of 2717 files processed.  
2037 1038 of 2717 files processed.  
2036 1030 of 2717 files processed.  
2035 1030 of 2717 files processed.  
2034 1036 of 2717 files processed.  
2034 1037 of 2717 files processed.  
2037 1039 of 2717 files processed.  
2035 1031 of 2717 files processed.  
2034 1038 of 2717 files processed.  
2036 1031 of 2717 files processed.  
2037 1040 of 2717 files processed.  
2035 1032 of 2717 files processed.  
2036 1032 of 2717 files processed.  
2034 1039 of 2717 files processed.  
-----

2037 1041 of 2717 files processed.  
2035 1033 of 2717 files processed.  
2036 1033 of 2717 files processed.  
2034 1040 of 2717 files processed.  
2037 1042 of 2717 files processed.  
2035 1034 of 2717 files processed.  
2036 1034 of 2717 files processed.  
2035 1035 of 2717 files processed.  
2034 1041 of 2717 files processed.  
2037 1043 of 2717 files processed.  
2036 1035 of 2717 files processed.  
2037 1044 of 2717 files processed.  
2034 1042 of 2717 files processed.  
2036 1036 of 2717 files processed.  
2035 1036 of 2717 files processed.  
2036 1037 of 2717 files processed.  
2035 1037 of 2717 files processed.  
2034 1043 of 2717 files processed.  
2037 1045 of 2717 files processed.  
2036 1038 of 2717 files processed.  
2034 1044 of 2717 files processed.  
2035 1038 of 2717 files processed.  
2036 1039 of 2717 files processed.  
2034 1045 of 2717 files processed.  
2037 1046 of 2717 files processed.  
2035 1039 of 2717 files processed.  
2036 1040 of 2717 files processed.  
2034 1046 of 2717 files processed.  
2035 1040 of 2717 files processed.  
2037 1047 of 2717 files processed.  
2036 1041 of 2717 files processed.  
2035 1041 of 2717 files processed.  
2034 1047 of 2717 files processed.  
2037 1048 of 2717 files processed.  
2036 1042 of 2717 files processed.  
2035 1042 of 2717 files processed.  
2034 1048 of 2717 files processed.  
2037 1049 of 2717 files processed.  
2036 1043 of 2717 files processed.  
2034 1049 of 2717 files processed.  
2035 1043 of 2717 files processed.  
2036 1044 of 2717 files processed.  
2034 1050 of 2717 files processed.  
2035 1044 of 2717 files processed.  
2037 1050 of 2717 files processed.  
2036 1045 of 2717 files processed.  
2034 1051 of 2717 files processed.  
2035 1045 of 2717 files processed.  
2036 1046 of 2717 files processed.  
2037 1051 of 2717 files processed.  
2034 1052 of 2717 files processed.  
2035 1046 of 2717 files processed.  
2036 1047 of 2717 files processed.  
2034 1053 of 2717 files processed.  
2037 1052 of 2717 files processed.  
2035 1047 of 2717 files processed.  
2036 1048 of 2717 files processed.  
2034 1054 of 2717 files processed.  
2035 1048 of 2717 files processed.  
2037 1053 of 2717 files processed.  
2036 1049 of 2717 files processed.  
2034 1055 of 2717 files processed.  
2035 1049 of 2717 files processed.  
2036 1050 of 2717 files processed.  
2037 1053 of 2717 files processed.  
2036 1049 of 2717 files processed.  
2034 1055 of 2717 files processed.  
2035 1050 of 2717 files processed.  
2036 1051 of 2717 files processed.  
2037 1055 of 2717 files processed.  
2034 1057 of 2717 files processed.  
2036 1052 of 2717 files processed.  
2035 1051 of 2717 files processed.  
2037 1056 of 2717 files processed.  
2034 1058 of 2717 files processed.  
2035 1052 of 2717 files processed.  
2036 1053 of 2717 files processed.  
2034 1059 of 2717 files processed.  
-----

2037 1057 of 2717 files processed.  
2035 1053 of 2717 files processed.  
2036 1054 of 2717 files processed.  
2037 1058 of 2717 files processed.  
2034 1060 of 2717 files processed.  
2035 1054 of 2717 files processed.  
2037 1059 of 2717 files processed.  
2036 1055 of 2717 files processed.  
2035 1055 of 2717 files processed.  
2034 1061 of 2717 files processed.  
2035 1056 of 2717 files processed.  
2036 1056 of 2717 files processed.  
2037 1060 of 2717 files processed.  
2034 1062 of 2717 files processed.  
2035 1057 of 2717 files processed.  
2036 1057 of 2717 files processed.  
2037 1061 of 2717 files processed.  
2034 1063 of 2717 files processed.  
2036 1058 of 2717 files processed.  
2037 1062 of 2717 files processed.  
2035 1058 of 2717 files processed.  
2034 1064 of 2717 files processed.  
2036 1059 of 2717 files processed.  
2037 1063 of 2717 files processed.  
2035 1059 of 2717 files processed.  
2034 1065 of 2717 files processed.  
2036 1060 of 2717 files processed.  
2035 1060 of 2717 files processed.  
2037 1064 of 2717 files processed.  
2034 1066 of 2717 files processed.  
2036 1061 of 2717 files processed.  
2035 1061 of 2717 files processed.  
2037 1065 of 2717 files processed.  
2034 1067 of 2717 files processed.  
2036 1062 of 2717 files processed.  
2037 1066 of 2717 files processed.  
2034 1068 of 2717 files processed.  
2035 1062 of 2717 files processed.  
2036 1063 of 2717 files processed.  
2034 1069 of 2717 files processed.  
2037 1067 of 2717 files processed.  
2036 1064 of 2717 files processed.  
2035 1063 of 2717 files processed.  
2036 1065 of 2717 files processed.  
2034 1070 of 2717 files processed.  
2036 1066 of 2717 files processed.  
2037 1068 of 2717 files processed.  
2035 1064 of 2717 files processed.  
2034 1071 of 2717 files processed.  
2036 1067 of 2717 files processed.  
2037 1069 of 2717 files processed.  
2035 1065 of 2717 files processed.  
2034 1072 of 2717 files processed.  
2035 1066 of 2717 files processed.  
2036 1068 of 2717 files processed.  
2034 1073 of 2717 files processed.  
2037 1070 of 2717 files processed.  
2036 1069 of 2717 files processed.  
2035 1067 of 2717 files processed.  
2034 1074 of 2717 files processed.  
2037 1071 of 2717 files processed.  
2035 1068 of 2717 files processed.  
2034 1075 of 2717 files processed.  
2036 1070 of 2717 files processed.  
2037 1072 of 2717 files processed.  
2035 1069 of 2717 files processed.  
2037 1073 of 2717 files processed.  
2034 1076 of 2717 files processed.  
2036 1071 of 2717 files processed.  
2037 1074 of 2717 files processed.  
2034 1077 of 2717 files processed.  
2035 1070 of 2717 files processed.  
2036 1072 of 2717 files processed.  
2035 1071 of 2717 files processed.  
2034 1078 of 2717 files processed.  
2037 1075 of 2717 files processed.  
2036 1073 of 2717 files processed.





2034 1116 of 2717 files processed.  
2036 1111 of 2717 files processed.  
2035 1113 of 2717 files processed.  
2037 1115 of 2717 files processed.  
2034 1117 of 2717 files processed.  
2036 1112 of 2717 files processed.  
2037 1116 of 2717 files processed.  
2035 1114 of 2717 files processed.  
2036 1113 of 2717 files processed.  
2034 1118 of 2717 files processed.  
2035 1115 of 2717 files processed.  
2037 1117 of 2717 files processed.  
2035 1116 of 2717 files processed.  
2034 1119 of 2717 files processed.  
2037 1118 of 2717 files processed.  
2036 1114 of 2717 files processed.  
2034 1120 of 2717 files processed.  
2036 1115 of 2717 files processed.  
2037 1119 of 2717 files processed.  
2035 1117 of 2717 files processed.  
2034 1121 of 2717 files processed.  
2036 1116 of 2717 files processed.  
2037 1120 of 2717 files processed.  
2035 1118 of 2717 files processed.  
2034 1122 of 2717 files processed.  
2036 1117 of 2717 files processed.  
2034 1123 of 2717 files processed.  
2037 1121 of 2717 files processed.  
2035 1119 of 2717 files processed.  
2034 1124 of 2717 files processed.  
2036 1118 of 2717 files processed.  
2037 1122 of 2717 files processed.  
2035 1120 of 2717 files processed.  
2037 1123 of 2717 files processed.  
2036 1119 of 2717 files processed.  
2034 1125 of 2717 files processed.  
2035 1121 of 2717 files processed.  
2037 1124 of 2717 files processed.  
2036 1120 of 2717 files processed.  
2034 1126 of 2717 files processed.  
2036 1121 of 2717 files processed.  
2035 1122 of 2717 files processed.  
2037 1125 of 2717 files processed.  
2034 1127 of 2717 files processed.  
2035 1123 of 2717 files processed.  
2037 1126 of 2717 files processed.  
2036 1122 of 2717 files processed.  
2034 1128 of 2717 files processed.  
2036 1123 of 2717 files processed.  
2035 1124 of 2717 files processed.  
2037 1127 of 2717 files processed.  
2035 1125 of 2717 files processed.  
2037 1128 of 2717 files processed.  
2034 1129 of 2717 files processed.  
2036 1124 of 2717 files processed.  
2035 1126 of 2717 files processed.  
2034 1130 of 2717 files processed.  
2037 1129 of 2717 files processed.  
2036 1125 of 2717 files processed.  
2035 1127 of 2717 files processed.  
2037 1130 of 2717 files processed.  
2034 1131 of 2717 files processed.  
2036 1126 of 2717 files processed.  
2035 1128 of 2717 files processed.  
2037 1131 of 2717 files processed.  
2034 1132 of 2717 files processed.  
2036 1127 of 2717 files processed.  
2036 1128 of 2717 files processed.  
2037 1132 of 2717 files processed.  
2035 1129 of 2717 files processed.  
2034 1133 of 2717 files processed.  
2037 1133 of 2717 files processed.  
2036 1129 of 2717 files processed.  
2035 1130 of 2717 files processed.  
2034 1134 of 2717 files processed.  
2036 1130 of 2717 files processed.  
2035 1131 of 2717 files processed.











----  
2037 1230 of 2717 files processed.  
2036 1229 of 2717 files processed.  
2035 1227 of 2717 files processed.  
2034 1231 of 2717 files processed.  
2037 1231 of 2717 files processed.  
2035 1228 of 2717 files processed.  
2036 1230 of 2717 files processed.  
2037 1232 of 2717 files processed.  
2034 1232 of 2717 files processed.  
2035 1229 of 2717 files processed.  
2036 1231 of 2717 files processed.  
2034 1233 of 2717 files processed.  
2037 1233 of 2717 files processed.  
2036 1232 of 2717 files processed.  
2035 1230 of 2717 files processed.  
2037 1234 of 2717 files processed.  
2034 1234 of 2717 files processed.  
2035 1231 of 2717 files processed.  
2036 1233 of 2717 files processed.  
2034 1235 of 2717 files processed.  
2037 1235 of 2717 files processed.  
2035 1232 of 2717 files processed.  
2034 1236 of 2717 files processed.  
2036 1234 of 2717 files processed.  
2035 1233 of 2717 files processed.  
2037 1236 of 2717 files processed.  
2034 1237 of 2717 files processed.  
2036 1235 of 2717 files processed.  
2035 1234 of 2717 files processed.  
2037 1237 of 2717 files processed.  
2034 1238 of 2717 files processed.  
2036 1236 of 2717 files processed.  
2035 1235 of 2717 files processed.  
2037 1238 of 2717 files processed.  
2036 1237 of 2717 files processed.  
2034 1239 of 2717 files processed.  
2035 1236 of 2717 files processed.  
2037 1239 of 2717 files processed.  
2036 1238 of 2717 files processed.  
2034 1240 of 2717 files processed.  
2034 1241 of 2717 files processed.  
2037 1240 of 2717 files processed.  
2035 1237 of 2717 files processed.  
2036 1239 of 2717 files processed.  
2034 1242 of 2717 files processed.  
2037 1241 of 2717 files processed.  
2035 1238 of 2717 files processed.  
2036 1240 of 2717 files processed.  
2037 1242 of 2717 files processed.  
2034 1243 of 2717 files processed.  
2035 1239 of 2717 files processed.  
2036 1241 of 2717 files processed.  
2037 1243 of 2717 files processed.  
2034 1244 of 2717 files processed.  
2035 1240 of 2717 files processed.  
2037 1244 of 2717 files processed.  
2034 1245 of 2717 files processed.  
2036 1242 of 2717 files processed.  
2037 1245 of 2717 files processed.  
2035 1241 of 2717 files processed.  
2034 1246 of 2717 files processed.  
2036 1243 of 2717 files processed.  
2037 1246 of 2717 files processed.  
2035 1242 of 2717 files processed.  
2036 1244 of 2717 files processed.  
2034 1247 of 2717 files processed.  
2037 1247 of 2717 files processed.  
2035 1243 of 2717 files processed.  
2034 1248 of 2717 files processed.  
2036 1245 of 2717 files processed.  
2037 1248 of 2717 files processed.  
2037 1249 of 2717 files processed.  
2036 1246 of 2717 files processed.  
2035 1244 of 2717 files processed.  
2034 1249 of 2717 files processed.  
2035 1245 of 2717 files processed.  
2034 1250 of 2717 files processed.



2035 1265 of 2717 files processed.  
2035 1264 of 2717 files processed.  
2036 1266 of 2717 files processed.  
2034 1271 of 2717 files processed.  
2035 1265 of 2717 files processed.  
2036 1267 of 2717 files processed.  
2037 1270 of 2717 files processed.  
2034 1272 of 2717 files processed.  
2035 1266 of 2717 files processed.  
2037 1271 of 2717 files processed.  
2036 1268 of 2717 files processed.  
2034 1273 of 2717 files processed.  
2035 1267 of 2717 files processed.  
2037 1272 of 2717 files processed.  
2036 1269 of 2717 files processed.  
2034 1274 of 2717 files processed.  
2035 1268 of 2717 files processed.  
2037 1273 of 2717 files processed.  
2036 1270 of 2717 files processed.  
2034 1275 of 2717 files processed.  
2035 1269 of 2717 files processed.  
2037 1274 of 2717 files processed.  
2034 1276 of 2717 files processed.  
2036 1271 of 2717 files processed.  
2035 1270 of 2717 files processed.  
2034 1277 of 2717 files processed.  
2037 1275 of 2717 files processed.  
2036 1272 of 2717 files processed.  
2035 1271 of 2717 files processed.  
2037 1276 of 2717 files processed.  
2036 1273 of 2717 files processed.  
2034 1278 of 2717 files processed.  
2035 1272 of 2717 files processed.  
2037 1277 of 2717 files processed.  
2036 1274 of 2717 files processed.  
2034 1279 of 2717 files processed.  
2035 1273 of 2717 files processed.  
2037 1278 of 2717 files processed.  
2036 1275 of 2717 files processed.  
2035 1274 of 2717 files processed.  
2034 1280 of 2717 files processed.  
2037 1279 of 2717 files processed.  
2035 1275 of 2717 files processed.  
2036 1276 of 2717 files processed.  
2034 1281 of 2717 files processed.  
2037 1280 of 2717 files processed.  
2035 1276 of 2717 files processed.  
2036 1277 of 2717 files processed.  
2034 1282 of 2717 files processed.  
2037 1281 of 2717 files processed.  
2035 1277 of 2717 files processed.  
2036 1278 of 2717 files processed.  
2034 1283 of 2717 files processed.  
2037 1282 of 2717 files processed.  
2036 1279 of 2717 files processed.  
2035 1278 of 2717 files processed.  
2034 1284 of 2717 files processed.  
2036 1280 of 2717 files processed.  
2037 1283 of 2717 files processed.  
2035 1279 of 2717 files processed.  
2034 1285 of 2717 files processed.  
2037 1284 of 2717 files processed.  
2036 1281 of 2717 files processed.  
2035 1280 of 2717 files processed.  
2034 1286 of 2717 files processed.  
2037 1285 of 2717 files processed.  
2034 1287 of 2717 files processed.  
2036 1282 of 2717 files processed.  
2035 1281 of 2717 files processed.  
2034 1288 of 2717 files processed.  
2037 1286 of 2717 files processed.  
2035 1282 of 2717 files processed.  
2036 1283 of 2717 files processed.  
2034 1289 of 2717 files processed.  
2035 1283 of 2717 files processed.  
2036 1284 of 2717 files processed.  
2037 1287 of 2717 files processed.  
2034 1290 of 2717 files processed.

2034 1290 of 2717 files processed.  
2036 1285 of 2717 files processed.  
2035 1284 of 2717 files processed.  
2037 1288 of 2717 files processed.  
2034 1291 of 2717 files processed.  
2036 1286 of 2717 files processed.  
2037 1289 of 2717 files processed.  
2035 1285 of 2717 files processed.  
2034 1292 of 2717 files processed.  
2037 1290 of 2717 files processed.  
2036 1287 of 2717 files processed.  
2035 1286 of 2717 files processed.  
2037 1291 of 2717 files processed.  
2035 1287 of 2717 files processed.  
2036 1288 of 2717 files processed.  
2034 1293 of 2717 files processed.  
2037 1292 of 2717 files processed.  
2036 1289 of 2717 files processed.  
2035 1288 of 2717 files processed.  
2034 1294 of 2717 files processed.  
2037 1293 of 2717 files processed.  
2036 1290 of 2717 files processed.  
2035 1289 of 2717 files processed.  
2034 1295 of 2717 files processed.  
2037 1294 of 2717 files processed.  
2035 1290 of 2717 files processed.  
2036 1291 of 2717 files processed.  
2034 1296 of 2717 files processed.  
2035 1291 of 2717 files processed.  
2037 1295 of 2717 files processed.  
2036 1292 of 2717 files processed.  
2034 1297 of 2717 files processed.  
2035 1292 of 2717 files processed.  
2037 1296 of 2717 files processed.  
2036 1293 of 2717 files processed.  
2034 1298 of 2717 files processed.  
2037 1297 of 2717 files processed.  
2035 1293 of 2717 files processed.  
2036 1294 of 2717 files processed.  
2034 1299 of 2717 files processed.  
2037 1298 of 2717 files processed.  
2035 1294 of 2717 files processed.  
2036 1295 of 2717 files processed.  
2034 1300 of 2717 files processed.  
2037 1299 of 2717 files processed.  
2036 1296 of 2717 files processed.  
2034 1301 of 2717 files processed.  
2035 1295 of 2717 files processed.  
2036 1297 of 2717 files processed.  
2037 1300 of 2717 files processed.  
2035 1296 of 2717 files processed.  
2034 1302 of 2717 files processed.  
2037 1301 of 2717 files processed.  
2035 1297 of 2717 files processed.  
2036 1298 of 2717 files processed.  
2034 1303 of 2717 files processed.  
2037 1302 of 2717 files processed.  
2036 1299 of 2717 files processed.  
2035 1298 of 2717 files processed.  
2034 1304 of 2717 files processed.  
2037 1303 of 2717 files processed.  
2035 1299 of 2717 files processed.  
2036 1300 of 2717 files processed.  
2034 1305 of 2717 files processed.  
2037 1304 of 2717 files processed.  
2036 1301 of 2717 files processed.  
2034 1306 of 2717 files processed.  
2035 1300 of 2717 files processed.  
2037 1305 of 2717 files processed.  
2036 1302 of 2717 files processed.  
2034 1307 of 2717 files processed.  
2035 1301 of 2717 files processed.  
2036 1303 of 2717 files processed.  
2037 1306 of 2717 files processed.  
2034 1308 of 2717 files processed.  
2036 1304 of 2717 files processed.  
2035 1302 of 2717 files processed.  
2037 1307 of 2717 files processed

2031 1501 of 2717 files processed.  
2034 1309 of 2717 files processed.  
2036 1305 of 2717 files processed.  
2035 1303 of 2717 files processed.  
2037 1308 of 2717 files processed.  
2034 1310 of 2717 files processed.  
2036 1306 of 2717 files processed.  
2035 1304 of 2717 files processed.  
2037 1309 of 2717 files processed.  
2034 1311 of 2717 files processed.  
2036 1307 of 2717 files processed.  
2037 1310 of 2717 files processed.  
2035 1305 of 2717 files processed.  
2034 1312 of 2717 files processed.  
2037 1311 of 2717 files processed.  
2036 1308 of 2717 files processed.  
2035 1306 of 2717 files processed.  
2034 1313 of 2717 files processed.  
2037 1312 of 2717 files processed.  
2036 1309 of 2717 files processed.  
2035 1307 of 2717 files processed.  
2037 1313 of 2717 files processed.  
2036 1310 of 2717 files processed.  
2035 1308 of 2717 files processed.  
2034 1314 of 2717 files processed.  
2037 1314 of 2717 files processed.  
2035 1309 of 2717 files processed.  
2034 1315 of 2717 files processed.  
2036 1311 of 2717 files processed.  
2037 1315 of 2717 files processed.  
2035 1310 of 2717 files processed.  
2034 1316 of 2717 files processed.  
2036 1312 of 2717 files processed.  
2035 1311 of 2717 files processed.  
2034 1317 of 2717 files processed.  
2037 1316 of 2717 files processed.  
2036 1313 of 2717 files processed.  
2035 1312 of 2717 files processed.  
2036 1314 of 2717 files processed.  
2034 1318 of 2717 files processed.  
2037 1317 of 2717 files processed.  
2035 1313 of 2717 files processed.  
2036 1315 of 2717 files processed.  
2037 1318 of 2717 files processed.  
2034 1319 of 2717 files processed.  
2036 1316 of 2717 files processed.  
2035 1314 of 2717 files processed.  
2037 1319 of 2717 files processed.  
2034 1320 of 2717 files processed.  
2035 1315 of 2717 files processed.  
2036 1317 of 2717 files processed.  
2037 1320 of 2717 files processed.  
2034 1321 of 2717 files processed.  
2035 1316 of 2717 files processed.  
2036 1318 of 2717 files processed.  
2037 1321 of 2717 files processed.  
2034 1322 of 2717 files processed.  
2035 1317 of 2717 files processed.  
2034 1323 of 2717 files processed.  
2036 1319 of 2717 files processed.  
2037 1322 of 2717 files processed.  
2035 1318 of 2717 files processed.  
2034 1324 of 2717 files processed.  
2036 1320 of 2717 files processed.  
2037 1323 of 2717 files processed.  
2035 1319 of 2717 files processed.  
2034 1325 of 2717 files processed.  
2036 1321 of 2717 files processed.  
2037 1324 of 2717 files processed.  
2034 1326 of 2717 files processed.  
2035 1320 of 2717 files processed.  
2036 1322 of 2717 files processed.  
2034 1327 of 2717 files processed.  
2037 1325 of 2717 files processed.  
2035 1321 of 2717 files processed.  
2036 1323 of 2717 files processed.  
2034 1328 of 2717 files processed.

2037 1326 of 2717 files processed.

2031 1326 of 2717 files processed.  
2034 1329 of 2717 files processed.  
2036 1324 of 2717 files processed.  
2035 1322 of 2717 files processed.  
2037 1327 of 2717 files processed.  
2036 1325 of 2717 files processed.  
2034 1330 of 2717 files processed.  
2035 1323 of 2717 files processed.  
2037 1328 of 2717 files processed.  
2036 1326 of 2717 files processed.  
2034 1331 of 2717 files processed.  
2035 1324 of 2717 files processed.  
2037 1329 of 2717 files processed.  
2034 1332 of 2717 files processed.  
2036 1327 of 2717 files processed.  
2037 1330 of 2717 files processed.  
2035 1325 of 2717 files processed.  
2034 1333 of 2717 files processed.  
2036 1328 of 2717 files processed.  
2037 1331 of 2717 files processed.  
2035 1326 of 2717 files processed.  
2034 1334 of 2717 files processed.  
2036 1329 of 2717 files processed.  
2037 1332 of 2717 files processed.  
2035 1327 of 2717 files processed.  
2034 1335 of 2717 files processed.  
2036 1330 of 2717 files processed.  
2037 1333 of 2717 files processed.  
2035 1328 of 2717 files processed.  
2034 1336 of 2717 files processed.  
2036 1331 of 2717 files processed.  
2037 1334 of 2717 files processed.  
2035 1329 of 2717 files processed.  
2034 1337 of 2717 files processed.  
2036 1332 of 2717 files processed.  
2037 1335 of 2717 files processed.  
2035 1330 of 2717 files processed.  
2034 1338 of 2717 files processed.  
2036 1333 of 2717 files processed.  
2037 1336 of 2717 files processed.  
2035 1331 of 2717 files processed.  
2037 1337 of 2717 files processed.  
2034 1339 of 2717 files processed.  
2036 1334 of 2717 files processed.  
2035 1332 of 2717 files processed.  
2034 1340 of 2717 files processed.  
2037 1338 of 2717 files processed.  
2035 1333 of 2717 files processed.  
2036 1335 of 2717 files processed.  
2037 1339 of 2717 files processed.  
2034 1341 of 2717 files processed.  
2036 1336 of 2717 files processed.  
2035 1334 of 2717 files processed.  
2037 1340 of 2717 files processed.  
2034 1342 of 2717 files processed.  
2036 1337 of 2717 files processed.  
2035 1335 of 2717 files processed.  
2037 1341 of 2717 files processed.  
2034 1343 of 2717 files processed.  
2036 1338 of 2717 files processed.  
2037 1342 of 2717 files processed.  
2035 1336 of 2717 files processed.  
2036 1339 of 2717 files processed.  
2034 1344 of 2717 files processed.  
2037 1343 of 2717 files processed.  
2034 1345 of 2717 files processed.  
2036 1340 of 2717 files processed.  
2035 1337 of 2717 files processed.  
2037 1344 of 2717 files processed.  
2036 1341 of 2717 files processed.  
2034 1346 of 2717 files processed.  
2035 1338 of 2717 files processed.  
2037 1345 of 2717 files processed.  
2036 1342 of 2717 files processed.  
2035 1339 of 2717 files processed.  
2034 1347 of 2717 files processed.  
2037 1346 of 2717 files processed.

2036 1343 of 2717 files processed.  
2035 1340 of 2717 files processed.  
2034 1348 of 2717 files processed.  
2037 1347 of 2717 files processed.  
2035 1341 of 2717 files processed.  
2036 1344 of 2717 files processed.  
2034 1349 of 2717 files processed.  
2036 1345 of 2717 files processed.  
2035 1342 of 2717 files processed.  
2037 1348 of 2717 files processed.  
2036 1346 of 2717 files processed.  
2034 1350 of 2717 files processed.  
2035 1343 of 2717 files processed.  
2037 1349 of 2717 files processed.  
2036 1347 of 2717 files processed.  
2035 1344 of 2717 files processed.  
2037 1350 of 2717 files processed.  
2034 1351 of 2717 files processed.  
2035 1345 of 2717 files processed.  
2037 1351 of 2717 files processed.  
2036 1348 of 2717 files processed.  
2034 1352 of 2717 files processed.  
2037 1352 of 2717 files processed.  
2035 1346 of 2717 files processed.  
2034 1353 of 2717 files processed.  
2036 1349 of 2717 files processed.  
2037 1353 of 2717 files processed.  
2034 1354 of 2717 files processed.  
2036 1350 of 2717 files processed.  
2035 1347 of 2717 files processed.  
2036 1351 of 2717 files processed.  
2037 1354 of 2717 files processed.  
2035 1348 of 2717 files processed.  
2034 1355 of 2717 files processed.  
2035 1349 of 2717 files processed.  
2037 1355 of 2717 files processed.  
2036 1352 of 2717 files processed.  
2034 1356 of 2717 files processed.  
2035 1350 of 2717 files processed.  
2036 1353 of 2717 files processed.  
2034 1357 of 2717 files processed.  
2037 1356 of 2717 files processed.  
2035 1351 of 2717 files processed.  
2036 1354 of 2717 files processed.  
2034 1358 of 2717 files processed.  
2037 1357 of 2717 files processed.  
2036 1355 of 2717 files processed.  
2034 1359 of 2717 files processed.  
2035 1352 of 2717 files processed.  
2037 1358 of 2717 files processed.  
2037 1359 of 2717 files processed.  
2034 1360 of 2717 files processed.  
2036 1356 of 2717 files processed.  
2035 1353 of 2717 files processed.  
2037 1360 of 2717 files processed.  
2036 1357 of 2717 files processed.  
2035 1354 of 2717 files processed.  
2034 1361 of 2717 files processed.  
2037 1361 of 2717 files processed.  
2036 1358 of 2717 files processed.  
2035 1355 of 2717 files processed.  
2034 1362 of 2717 files processed.  
2037 1362 of 2717 files processed.  
2036 1359 of 2717 files processed.  
2035 1356 of 2717 files processed.  
2034 1363 of 2717 files processed.  
2037 1363 of 2717 files processed.  
2036 1360 of 2717 files processed.  
2035 1357 of 2717 files processed.  
2037 1364 of 2717 files processed.  
2034 1364 of 2717 files processed.  
2036 1361 of 2717 files processed.  
2035 1358 of 2717 files processed.  
2037 1365 of 2717 files processed.  
2034 1365 of 2717 files processed.  
2036 1362 of 2717 files processed.  
2035 1359 of 2717 files processed.  
-----



2036 1382 of 2717 files processed.  
2034 1385 of 2717 files processed.  
2035 1378 of 2717 files processed.  
2036 1383 of 2717 files processed.  
2037 1387 of 2717 files processed.  
2034 1386 of 2717 files processed.  
2035 1379 of 2717 files processed.  
2036 1384 of 2717 files processed.  
2037 1388 of 2717 files processed.  
2035 1380 of 2717 files processed.  
2034 1387 of 2717 files processed.  
2037 1389 of 2717 files processed.  
2036 1385 of 2717 files processed.  
2037 1390 of 2717 files processed.  
2034 1388 of 2717 files processed.  
2035 1381 of 2717 files processed.  
2036 1386 of 2717 files processed.  
2037 1391 of 2717 files processed.  
2034 1389 of 2717 files processed.  
2035 1382 of 2717 files processed.  
2036 1387 of 2717 files processed.  
2037 1392 of 2717 files processed.  
2035 1383 of 2717 files processed.  
2034 1390 of 2717 files processed.  
2037 1393 of 2717 files processed.  
2036 1388 of 2717 files processed.  
2035 1384 of 2717 files processed.  
2034 1391 of 2717 files processed.  
2036 1389 of 2717 files processed.  
2037 1394 of 2717 files processed.  
2034 1392 of 2717 files processed.  
2035 1385 of 2717 files processed.  
2034 1393 of 2717 files processed.  
2035 1386 of 2717 files processed.  
2036 1390 of 2717 files processed.  
2037 1395 of 2717 files processed.  
2034 1394 of 2717 files processed.  
2037 1396 of 2717 files processed.  
2035 1387 of 2717 files processed.  
2036 1391 of 2717 files processed.  
2034 1395 of 2717 files processed.  
2037 1397 of 2717 files processed.  
2035 1388 of 2717 files processed.  
2036 1392 of 2717 files processed.  
2035 1389 of 2717 files processed.  
2034 1396 of 2717 files processed.  
2037 1398 of 2717 files processed.  
2036 1393 of 2717 files processed.  
2035 1390 of 2717 files processed.  
2034 1397 of 2717 files processed.  
2037 1399 of 2717 files processed.  
2034 1398 of 2717 files processed.  
2036 1394 of 2717 files processed.  
2037 1400 of 2717 files processed.  
2035 1391 of 2717 files processed.  
2037 1401 of 2717 files processed.  
2036 1395 of 2717 files processed.  
2034 1399 of 2717 files processed.  
2035 1392 of 2717 files processed.  
2037 1402 of 2717 files processed.  
2036 1396 of 2717 files processed.  
2035 1393 of 2717 files processed.  
2034 1400 of 2717 files processed.  
2037 1403 of 2717 files processed.  
2036 1397 of 2717 files processed.  
2035 1394 of 2717 files processed.  
2034 1401 of 2717 files processed.  
2037 1404 of 2717 files processed.  
2036 1398 of 2717 files processed.  
2035 1395 of 2717 files processed.  
2034 1402 of 2717 files processed.  
2036 1399 of 2717 files processed.  
2035 1396 of 2717 files processed.  
2037 1405 of 2717 files processed.  
2034 1403 of 2717 files processed.  
2036 1400 of 2717 files processed.  
2035 1397 of 2717 files processed.

2034 1404 of 2717 files processed.  
2037 1406 of 2717 files processed.  
2035 1398 of 2717 files processed.  
2036 1401 of 2717 files processed.  
2034 1405 of 2717 files processed.  
2037 1407 of 2717 files processed.  
2035 1399 of 2717 files processed.  
2036 1402 of 2717 files processed.  
2037 1408 of 2717 files processed.  
2034 1406 of 2717 files processed.  
2035 1400 of 2717 files processed.  
2036 1403 of 2717 files processed.  
2037 1409 of 2717 files processed.  
2034 1407 of 2717 files processed.  
2037 1410 of 2717 files processed.  
2035 1401 of 2717 files processed.  
2036 1404 of 2717 files processed.  
2034 1408 of 2717 files processed.  
2035 1402 of 2717 files processed.  
2036 1405 of 2717 files processed.  
2037 1411 of 2717 files processed.  
2034 1409 of 2717 files processed.  
2035 1403 of 2717 files processed.  
2036 1406 of 2717 files processed.  
2037 1412 of 2717 files processed.  
2034 1410 of 2717 files processed.  
2035 1404 of 2717 files processed.  
2037 1413 of 2717 files processed.  
2034 1411 of 2717 files processed.  
2036 1407 of 2717 files processed.  
2035 1405 of 2717 files processed.  
2037 1414 of 2717 files processed.  
2036 1408 of 2717 files processed.  
2035 1406 of 2717 files processed.  
2037 1415 of 2717 files processed.  
2034 1412 of 2717 files processed.  
2036 1409 of 2717 files processed.  
2035 1407 of 2717 files processed.  
2037 1416 of 2717 files processed.  
2034 1413 of 2717 files processed.  
2036 1410 of 2717 files processed.  
2035 1408 of 2717 files processed.  
2037 1417 of 2717 files processed.  
2036 1411 of 2717 files processed.  
2037 1418 of 2717 files processed.  
2034 1414 of 2717 files processed.  
2035 1409 of 2717 files processed.  
2036 1412 of 2717 files processed.  
2034 1415 of 2717 files processed.  
2037 1419 of 2717 files processed.  
2035 1410 of 2717 files processed.  
2036 1413 of 2717 files processed.  
2034 1416 of 2717 files processed.  
2037 1420 of 2717 files processed.  
2035 1411 of 2717 files processed.  
2034 1417 of 2717 files processed.  
2036 1414 of 2717 files processed.  
2035 1412 of 2717 files processed.  
2037 1421 of 2717 files processed.  
2036 1415 of 2717 files processed.  
2034 1418 of 2717 files processed.  
2035 1413 of 2717 files processed.  
2036 1416 of 2717 files processed.  
2037 1422 of 2717 files processed.  
2034 1419 of 2717 files processed.  
2035 1414 of 2717 files processed.  
2036 1417 of 2717 files processed.  
2037 1423 of 2717 files processed.  
2034 1420 of 2717 files processed.  
2035 1415 of 2717 files processed.  
2037 1424 of 2717 files processed.  
2036 1418 of 2717 files processed.  
2035 1416 of 2717 files processed.  
2034 1421 of 2717 files processed.  
2037 1425 of 2717 files processed.  
2036 1419 of 2717 files processed.  
2034 1422 of 2717 files processed.

2035 1417 of 2717 files processed.  
2037 1426 of 2717 files processed.  
2034 1423 of 2717 files processed.  
2035 1418 of 2717 files processed.  
2036 1420 of 2717 files processed.  
2037 1427 of 2717 files processed.  
2034 1424 of 2717 files processed.  
2035 1419 of 2717 files processed.  
2037 1428 of 2717 files processed.  
2036 1421 of 2717 files processed.  
2034 1425 of 2717 files processed.  
2035 1420 of 2717 files processed.  
2037 1429 of 2717 files processed.  
2036 1422 of 2717 files processed.  
2034 1426 of 2717 files processed.  
2035 1421 of 2717 files processed.  
2037 1430 of 2717 files processed.  
2036 1423 of 2717 files processed.  
2034 1427 of 2717 files processed.  
2035 1422 of 2717 files processed.  
2037 1431 of 2717 files processed.  
2034 1428 of 2717 files processed.  
2036 1424 of 2717 files processed.  
2035 1423 of 2717 files processed.  
2037 1432 of 2717 files processed.  
2034 1429 of 2717 files processed.  
2036 1425 of 2717 files processed.  
2035 1424 of 2717 files processed.  
2037 1433 of 2717 files processed.  
2036 1426 of 2717 files processed.  
2034 1430 of 2717 files processed.  
2035 1425 of 2717 files processed.  
2034 1431 of 2717 files processed.  
2037 1434 of 2717 files processed.  
2035 1426 of 2717 files processed.  
2036 1427 of 2717 files processed.  
2034 1432 of 2717 files processed.  
2037 1435 of 2717 files processed.  
2035 1427 of 2717 files processed.  
2034 1433 of 2717 files processed.  
2036 1428 of 2717 files processed.  
2037 1436 of 2717 files processed.  
2035 1428 of 2717 files processed.  
2037 1437 of 2717 files processed.  
2036 1429 of 2717 files processed.  
2034 1434 of 2717 files processed.  
2035 1429 of 2717 files processed.  
2036 1430 of 2717 files processed.  
2034 1435 of 2717 files processed.  
2037 1438 of 2717 files processed.  
2035 1430 of 2717 files processed.  
2034 1436 of 2717 files processed.  
2037 1439 of 2717 files processed.  
2036 1431 of 2717 files processed.  
2035 1431 of 2717 files processed.  
2037 1440 of 2717 files processed.  
2034 1437 of 2717 files processed.  
2036 1432 of 2717 files processed.  
2035 1432 of 2717 files processed.  
2037 1441 of 2717 files processed.  
2035 1433 of 2717 files processed.  
2036 1433 of 2717 files processed.  
2034 1438 of 2717 files processed.  
2037 1442 of 2717 files processed.  
2035 1434 of 2717 files processed.  
2034 1439 of 2717 files processed.  
2036 1434 of 2717 files processed.  
2035 1435 of 2717 files processed.  
2037 1443 of 2717 files processed.  
2034 1440 of 2717 files processed.  
2036 1435 of 2717 files processed.  
2035 1436 of 2717 files processed.  
2037 1444 of 2717 files processed.  
2036 1436 of 2717 files processed.  
2035 1437 of 2717 files processed.  
2034 1441 of 2717 files processed.  
2037 1445 of 2717 files processed.



2034 1461 of 2717 files processed.  
2037 1464 of 2717 files processed.  
2036 1458 of 2717 files processed.  
2035 1457 of 2717 files processed.  
2034 1462 of 2717 files processed.  
2036 1459 of 2717 files processed.  
2037 1465 of 2717 files processed.  
2034 1463 of 2717 files processed.  
2035 1458 of 2717 files processed.  
2037 1466 of 2717 files processed.  
2036 1460 of 2717 files processed.  
2035 1459 of 2717 files processed.  
2034 1464 of 2717 files processed.  
2037 1467 of 2717 files processed.  
2036 1461 of 2717 files processed.  
2035 1460 of 2717 files processed.  
2037 1468 of 2717 files processed.  
2034 1465 of 2717 files processed.  
2036 1462 of 2717 files processed.  
2035 1461 of 2717 files processed.  
2034 1466 of 2717 files processed.  
2036 1463 of 2717 files processed.  
2037 1469 of 2717 files processed.  
2036 1464 of 2717 files processed.  
2035 1462 of 2717 files processed.  
2034 1467 of 2717 files processed.  
2037 1470 of 2717 files processed.  
2036 1465 of 2717 files processed.  
2034 1468 of 2717 files processed.  
2035 1463 of 2717 files processed.  
2037 1471 of 2717 files processed.  
2036 1466 of 2717 files processed.  
2035 1464 of 2717 files processed.  
2034 1469 of 2717 files processed.  
2037 1472 of 2717 files processed.  
2036 1467 of 2717 files processed.  
2035 1465 of 2717 files processed.  
2034 1470 of 2717 files processed.  
2037 1473 of 2717 files processed.  
2036 1468 of 2717 files processed.  
2034 1471 of 2717 files processed.  
2037 1474 of 2717 files processed.  
2035 1466 of 2717 files processed.  
2036 1469 of 2717 files processed.  
2037 1475 of 2717 files processed.  
2034 1472 of 2717 files processed.  
2035 1467 of 2717 files processed.  
2036 1470 of 2717 files processed.  
2034 1473 of 2717 files processed.  
2037 1476 of 2717 files processed.  
2036 1471 of 2717 files processed.  
2035 1468 of 2717 files processed.  
2034 1474 of 2717 files processed.  
2037 1477 of 2717 files processed.  
2035 1469 of 2717 files processed.  
2036 1472 of 2717 files processed.  
2034 1475 of 2717 files processed.  
2036 1473 of 2717 files processed.  
2037 1478 of 2717 files processed.  
2035 1470 of 2717 files processed.  
2034 1476 of 2717 files processed.  
2037 1479 of 2717 files processed.  
2035 1471 of 2717 files processed.  
2036 1474 of 2717 files processed.  
2034 1477 of 2717 files processed.  
2037 1480 of 2717 files processed.  
2035 1472 of 2717 files processed.  
2036 1475 of 2717 files processed.  
2034 1478 of 2717 files processed.  
2037 1481 of 2717 files processed.  
2035 1473 of 2717 files processed.  
2036 1476 of 2717 files processed.  
2034 1479 of 2717 files processed.  
2037 1482 of 2717 files processed.  
2035 1474 of 2717 files processed.  
2036 1477 of 2717 files processed.  
2034 1480 of 2717 files processed.

2037 1483 of 2717 files processed.  
2035 1475 of 2717 files processed.  
2036 1478 of 2717 files processed.  
2034 1481 of 2717 files processed.  
2037 1484 of 2717 files processed.  
2036 1479 of 2717 files processed.  
2035 1476 of 2717 files processed.  
2037 1485 of 2717 files processed.  
2034 1482 of 2717 files processed.  
2035 1477 of 2717 files processed.  
2036 1480 of 2717 files processed.  
2037 1486 of 2717 files processed.  
2034 1483 of 2717 files processed.  
2036 1481 of 2717 files processed.  
2035 1478 of 2717 files processed.  
2037 1487 of 2717 files processed.  
2034 1484 of 2717 files processed.  
2036 1482 of 2717 files processed.  
2035 1479 of 2717 files processed.  
2037 1488 of 2717 files processed.  
2034 1485 of 2717 files processed.  
2036 1483 of 2717 files processed.  
2035 1480 of 2717 files processed.  
2034 1486 of 2717 files processed.  
2037 1489 of 2717 files processed.  
2036 1484 of 2717 files processed.  
2034 1487 of 2717 files processed.  
2037 1490 of 2717 files processed.  
2036 1485 of 2717 files processed.  
2035 1481 of 2717 files processed.  
2034 1488 of 2717 files processed.  
2037 1491 of 2717 files processed.  
2036 1486 of 2717 files processed.  
2035 1482 of 2717 files processed.  
2034 1489 of 2717 files processed.  
2036 1487 of 2717 files processed.  
2037 1492 of 2717 files processed.  
2035 1483 of 2717 files processed.  
2034 1490 of 2717 files processed.  
2036 1488 of 2717 files processed.  
2037 1493 of 2717 files processed.  
2035 1484 of 2717 files processed.  
2034 1491 of 2717 files processed.  
2036 1489 of 2717 files processed.  
2037 1494 of 2717 files processed.  
2035 1485 of 2717 files processed.  
2034 1492 of 2717 files processed.  
2036 1490 of 2717 files processed.  
2037 1495 of 2717 files processed.  
2035 1486 of 2717 files processed.  
2034 1493 of 2717 files processed.  
2037 1496 of 2717 files processed.  
2036 1491 of 2717 files processed.  
2035 1487 of 2717 files processed.  
2034 1494 of 2717 files processed.  
2035 1488 of 2717 files processed.  
2037 1497 of 2717 files processed.  
2036 1492 of 2717 files processed.  
2034 1495 of 2717 files processed.  
2035 1489 of 2717 files processed.  
2037 1498 of 2717 files processed.  
2036 1493 of 2717 files processed.  
2034 1496 of 2717 files processed.  
2035 1490 of 2717 files processed.  
2037 1499 of 2717 files processed.  
2036 1494 of 2717 files processed.  
2034 1497 of 2717 files processed.  
2035 1491 of 2717 files processed.  
2037 1500 of 2717 files processed.  
2036 1495 of 2717 files processed.  
2034 1498 of 2717 files processed.  
2037 1501 of 2717 files processed.  
2035 1492 of 2717 files processed.  
2036 1496 of 2717 files processed.  
2034 1499 of 2717 files processed.  
2037 1502 of 2717 files processed.  
2035 1493 of 2717 files processed.

2036 1497 of 2717 files processed.  
2034 1500 of 2717 files processed.  
2037 1503 of 2717 files processed.  
2036 1498 of 2717 files processed.  
2034 1501 of 2717 files processed.  
2035 1494 of 2717 files processed.  
2037 1504 of 2717 files processed.  
2034 1502 of 2717 files processed.  
2035 1495 of 2717 files processed.  
2036 1499 of 2717 files processed.  
2037 1505 of 2717 files processed.  
2034 1503 of 2717 files processed.  
2035 1496 of 2717 files processed.  
2036 1500 of 2717 files processed.  
2037 1506 of 2717 files processed.  
2036 1501 of 2717 files processed.  
2035 1497 of 2717 files processed.  
2034 1504 of 2717 files processed.  
2037 1507 of 2717 files processed.  
2036 1502 of 2717 files processed.  
2035 1498 of 2717 files processed.  
2034 1505 of 2717 files processed.  
2037 1508 of 2717 files processed.  
2036 1503 of 2717 files processed.  
2035 1499 of 2717 files processed.  
2034 1506 of 2717 files processed.  
2037 1509 of 2717 files processed.  
2036 1504 of 2717 files processed.  
2035 1500 of 2717 files processed.  
2034 1507 of 2717 files processed.  
2037 1510 of 2717 files processed.  
2036 1505 of 2717 files processed.  
2034 1508 of 2717 files processed.  
2037 1511 of 2717 files processed.  
2035 1501 of 2717 files processed.  
2036 1506 of 2717 files processed.  
2034 1509 of 2717 files processed.  
2037 1512 of 2717 files processed.  
2035 1502 of 2717 files processed.  
2036 1507 of 2717 files processed.  
2037 1513 of 2717 files processed.  
2034 1510 of 2717 files processed.  
2035 1503 of 2717 files processed.  
2036 1508 of 2717 files processed.  
2034 1511 of 2717 files processed.  
2037 1514 of 2717 files processed.  
2036 1509 of 2717 files processed.  
2035 1504 of 2717 files processed.  
2037 1515 of 2717 files processed.  
2034 1512 of 2717 files processed.  
2036 1510 of 2717 files processed.  
2035 1505 of 2717 files processed.  
2034 1513 of 2717 files processed.  
2037 1516 of 2717 files processed.  
2036 1511 of 2717 files processed.  
2035 1506 of 2717 files processed.  
2037 1517 of 2717 files processed.  
2034 1514 of 2717 files processed.  
2035 1507 of 2717 files processed.  
2036 1512 of 2717 files processed.  
2037 1518 of 2717 files processed.  
2034 1515 of 2717 files processed.  
2036 1513 of 2717 files processed.  
2035 1508 of 2717 files processed.  
2037 1519 of 2717 files processed.  
2034 1516 of 2717 files processed.  
2036 1514 of 2717 files processed.  
2035 1509 of 2717 files processed.  
2037 1520 of 2717 files processed.  
2034 1517 of 2717 files processed.  
2036 1515 of 2717 files processed.  
2035 1510 of 2717 files processed.  
2037 1521 of 2717 files processed.  
2036 1516 of 2717 files processed.  
2034 1518 of 2717 files processed.  
2035 1511 of 2717 files processed.  
2037 1522 of 2717 files processed.



2037 1542 of 2717 files processed.  
2034 1538 of 2717 files processed.  
2036 1537 of 2717 files processed.  
2035 1531 of 2717 files processed.  
2037 1543 of 2717 files processed.  
2034 1539 of 2717 files processed.  
2036 1538 of 2717 files processed.  
2035 1532 of 2717 files processed.  
2037 1544 of 2717 files processed.  
2035 1533 of 2717 files processed.  
2034 1540 of 2717 files processed.  
2037 1545 of 2717 files processed.  
2036 1539 of 2717 files processed.  
2034 1541 of 2717 files processed.  
2035 1534 of 2717 files processed.  
2036 1540 of 2717 files processed.  
2037 1546 of 2717 files processed.  
2034 1542 of 2717 files processed.  
2035 1535 of 2717 files processed.  
2036 1541 of 2717 files processed.  
2037 1547 of 2717 files processed.  
2034 1543 of 2717 files processed.  
2035 1536 of 2717 files processed.  
2036 1542 of 2717 files processed.  
2037 1548 of 2717 files processed.  
2034 1544 of 2717 files processed.  
2035 1537 of 2717 files processed.  
2036 1543 of 2717 files processed.  
2034 1545 of 2717 files processed.  
2037 1549 of 2717 files processed.  
2035 1538 of 2717 files processed.  
2036 1544 of 2717 files processed.  
2035 1539 of 2717 files processed.  
2034 1546 of 2717 files processed.  
2037 1550 of 2717 files processed.  
2036 1545 of 2717 files processed.  
2035 1540 of 2717 files processed.  
2037 1551 of 2717 files processed.  
2034 1547 of 2717 files processed.  
2036 1546 of 2717 files processed.  
2035 1541 of 2717 files processed.  
2034 1548 of 2717 files processed.  
2037 1552 of 2717 files processed.  
2035 1542 of 2717 files processed.  
2036 1547 of 2717 files processed.  
2037 1553 of 2717 files processed.  
2034 1549 of 2717 files processed.  
2035 1543 of 2717 files processed.  
2036 1548 of 2717 files processed.  
2035 1544 of 2717 files processed.  
2037 1554 of 2717 files processed.  
2034 1550 of 2717 files processed.  
2036 1549 of 2717 files processed.  
2034 1551 of 2717 files processed.  
2037 1555 of 2717 files processed.  
2035 1545 of 2717 files processed.  
2036 1550 of 2717 files processed.  
2037 1556 of 2717 files processed.  
2034 1552 of 2717 files processed.  
2035 1546 of 2717 files processed.  
2036 1551 of 2717 files processed.  
2037 1557 of 2717 files processed.  
2034 1553 of 2717 files processed.  
2035 1547 of 2717 files processed.  
2036 1552 of 2717 files processed.  
2037 1558 of 2717 files processed.  
2034 1554 of 2717 files processed.  
2036 1553 of 2717 files processed.  
2035 1548 of 2717 files processed.  
2037 1559 of 2717 files processed.  
2034 1555 of 2717 files processed.  
2036 1554 of 2717 files processed.  
2035 1549 of 2717 files processed.  
2037 1560 of 2717 files processed.  
2036 1555 of 2717 files processed.  
2034 1556 of 2717 files processed.  
2035 1550 of 2717 files processed.



.....  
2035 1570 of 2717 files processed.  
2034 1576 of 2717 files processed.  
2037 1580 of 2717 files processed.  
2035 1571 of 2717 files processed.  
2036 1576 of 2717 files processed.  
2034 1577 of 2717 files processed.  
2035 1572 of 2717 files processed.  
2037 1581 of 2717 files processed.  
2036 1577 of 2717 files processed.  
2034 1578 of 2717 files processed.  
2035 1573 of 2717 files processed.  
2037 1582 of 2717 files processed.  
2036 1578 of 2717 files processed.  
2034 1579 of 2717 files processed.  
2035 1574 of 2717 files processed.  
2037 1583 of 2717 files processed.  
2036 1579 of 2717 files processed.  
2034 1580 of 2717 files processed.  
2037 1584 of 2717 files processed.  
2036 1580 of 2717 files processed.  
2035 1575 of 2717 files processed.  
2034 1581 of 2717 files processed.  
2036 1581 of 2717 files processed.  
2035 1576 of 2717 files processed.  
2037 1585 of 2717 files processed.  
2034 1582 of 2717 files processed.  
2037 1586 of 2717 files processed.  
2036 1582 of 2717 files processed.  
2035 1577 of 2717 files processed.  
2034 1583 of 2717 files processed.  
2036 1583 of 2717 files processed.  
2035 1578 of 2717 files processed.  
2037 1587 of 2717 files processed.  
2034 1584 of 2717 files processed.  
2036 1584 of 2717 files processed.  
2037 1588 of 2717 files processed.  
2035 1579 of 2717 files processed.  
2034 1585 of 2717 files processed.  
2036 1585 of 2717 files processed.  
2037 1589 of 2717 files processed.  
2035 1580 of 2717 files processed.  
2037 1590 of 2717 files processed.  
2034 1586 of 2717 files processed.  
2035 1581 of 2717 files processed.  
2036 1586 of 2717 files processed.  
2037 1591 of 2717 files processed.  
2034 1587 of 2717 files processed.  
2035 1582 of 2717 files processed.  
2036 1587 of 2717 files processed.  
2037 1592 of 2717 files processed.  
2034 1588 of 2717 files processed.  
2035 1583 of 2717 files processed.  
2037 1593 of 2717 files processed.  
2034 1589 of 2717 files processed.  
2035 1584 of 2717 files processed.  
2036 1588 of 2717 files processed.  
2037 1594 of 2717 files processed.  
2034 1590 of 2717 files processed.  
2035 1585 of 2717 files processed.  
2036 1589 of 2717 files processed.  
2037 1595 of 2717 files processed.  
2035 1586 of 2717 files processed.  
2034 1591 of 2717 files processed.  
2036 1590 of 2717 files processed.  
2037 1596 of 2717 files processed.  
2036 1591 of 2717 files processed.  
2034 1592 of 2717 files processed.  
2035 1587 of 2717 files processed.  
2037 1597 of 2717 files processed.  
2034 1593 of 2717 files processed.  
2036 1592 of 2717 files processed.  
2037 1598 of 2717 files processed.  
2035 1588 of 2717 files processed.  
2034 1594 of 2717 files processed.  
2036 1593 of 2717 files processed.  
2035 1589 of 2717 files processed.  
2034 1595 of 2717 files processed.

2037 1599 of 2717 files processed.  
2036 1594 of 2717 files processed.  
2035 1590 of 2717 files processed.  
2037 1600 of 2717 files processed.  
2034 1596 of 2717 files processed.  
2036 1595 of 2717 files processed.  
2035 1591 of 2717 files processed.  
2037 1601 of 2717 files processed.  
2034 1597 of 2717 files processed.  
2036 1596 of 2717 files processed.  
2035 1592 of 2717 files processed.  
2037 1602 of 2717 files processed.  
2034 1598 of 2717 files processed.  
2036 1597 of 2717 files processed.  
2037 1603 of 2717 files processed.  
2035 1593 of 2717 files processed.  
2034 1599 of 2717 files processed.  
2035 1594 of 2717 files processed.  
2036 1598 of 2717 files processed.  
2037 1604 of 2717 files processed.  
2034 1600 of 2717 files processed.  
2035 1595 of 2717 files processed.  
2036 1599 of 2717 files processed.  
2037 1605 of 2717 files processed.  
2034 1601 of 2717 files processed.  
2035 1596 of 2717 files processed.  
2036 1600 of 2717 files processed.  
2037 1606 of 2717 files processed.  
2034 1602 of 2717 files processed.  
2035 1597 of 2717 files processed.  
2036 1601 of 2717 files processed.  
2037 1607 of 2717 files processed.  
2034 1603 of 2717 files processed.  
2035 1598 of 2717 files processed.  
2036 1602 of 2717 files processed.  
2037 1608 of 2717 files processed.  
2034 1604 of 2717 files processed.  
2035 1599 of 2717 files processed.  
2036 1603 of 2717 files processed.  
2037 1609 of 2717 files processed.  
2034 1605 of 2717 files processed.  
2035 1600 of 2717 files processed.  
2036 1604 of 2717 files processed.  
2037 1610 of 2717 files processed.  
2034 1606 of 2717 files processed.  
2037 1611 of 2717 files processed.  
2036 1605 of 2717 files processed.  
2035 1601 of 2717 files processed.  
2034 1607 of 2717 files processed.  
2037 1612 of 2717 files processed.  
2036 1606 of 2717 files processed.  
2035 1602 of 2717 files processed.  
2034 1608 of 2717 files processed.  
2037 1613 of 2717 files processed.  
2036 1607 of 2717 files processed.  
2035 1603 of 2717 files processed.  
2037 1614 of 2717 files processed.  
2034 1609 of 2717 files processed.  
2036 1608 of 2717 files processed.  
2035 1604 of 2717 files processed.  
2036 1609 of 2717 files processed.  
2037 1615 of 2717 files processed.  
2034 1610 of 2717 files processed.  
2035 1605 of 2717 files processed.  
2037 1616 of 2717 files processed.  
2036 1610 of 2717 files processed.  
2034 1611 of 2717 files processed.  
2035 1606 of 2717 files processed.  
2036 1611 of 2717 files processed.  
2037 1617 of 2717 files processed.  
2034 1612 of 2717 files processed.  
2035 1607 of 2717 files processed.  
2036 1612 of 2717 files processed.  
2037 1618 of 2717 files processed.  
2034 1613 of 2717 files processed.  
2036 1613 of 2717 files processed.  
2035 1608 of 2717 files processed.





















2035 1798 of 2717 files processed.  
2034 1803 of 2717 files processed.  
2037 1815 of 2717 files processed.  
2035 1799 of 2717 files processed.  
2036 1809 of 2717 files processed.  
2034 1804 of 2717 files processed.  
2035 1800 of 2717 files processed.  
2037 1816 of 2717 files processed.  
2036 1810 of 2717 files processed.  
2035 1801 of 2717 files processed.  
2034 1805 of 2717 files processed.  
2037 1817 of 2717 files processed.  
2036 1811 of 2717 files processed.  
2034 1806 of 2717 files processed.  
2035 1802 of 2717 files processed.  
2037 1818 of 2717 files processed.  
2036 1812 of 2717 files processed.  
2037 1819 of 2717 files processed.  
2035 1803 of 2717 files processed.  
2036 1813 of 2717 files processed.  
2034 1807 of 2717 files processed.  
2037 1820 of 2717 files processed.  
2036 1814 of 2717 files processed.  
2034 1808 of 2717 files processed.  
2035 1804 of 2717 files processed.  
2034 1809 of 2717 files processed.  
2037 1821 of 2717 files processed.  
2036 1815 of 2717 files processed.  
2035 1805 of 2717 files processed.  
2036 1816 of 2717 files processed.  
2037 1822 of 2717 files processed.  
2034 1810 of 2717 files processed.  
2035 1806 of 2717 files processed.  
2036 1817 of 2717 files processed.  
2037 1823 of 2717 files processed.  
2034 1811 of 2717 files processed.  
2035 1807 of 2717 files processed.  
2034 1812 of 2717 files processed.  
2036 1818 of 2717 files processed.  
2037 1824 of 2717 files processed.  
2035 1808 of 2717 files processed.  
2034 1813 of 2717 files processed.  
2037 1825 of 2717 files processed.  
2036 1819 of 2717 files processed.  
2035 1809 of 2717 files processed.  
2037 1826 of 2717 files processed.  
2034 1814 of 2717 files processed.  
2036 1820 of 2717 files processed.  
2035 1810 of 2717 files processed.  
2037 1827 of 2717 files processed.  
2034 1815 of 2717 files processed.  
2036 1821 of 2717 files processed.  
2035 1811 of 2717 files processed.  
2036 1822 of 2717 files processed.  
2034 1816 of 2717 files processed.  
2037 1828 of 2717 files processed.  
2035 1812 of 2717 files processed.  
2034 1817 of 2717 files processed.  
2036 1823 of 2717 files processed.  
2035 1813 of 2717 files processed.  
2037 1829 of 2717 files processed.  
2034 1818 of 2717 files processed.  
2036 1824 of 2717 files processed.  
2035 1814 of 2717 files processed.  
2037 1830 of 2717 files processed.  
2034 1819 of 2717 files processed.  
2036 1825 of 2717 files processed.  
2035 1815 of 2717 files processed.  
2037 1831 of 2717 files processed.  
2034 1820 of 2717 files processed.  
2036 1826 of 2717 files processed.  
2035 1816 of 2717 files processed.  
2037 1832 of 2717 files processed.  
2036 1827 of 2717 files processed.  
2034 1821 of 2717 files processed.  
2035 1817 of 2717 files processed.  
2037 1833 of 2717 files processed.



2034 1842 of 2717 files processed.  
2036 1847 of 2717 files processed.  
2035 1837 of 2717 files processed.  
2034 1843 of 2717 files processed.  
2037 1853 of 2717 files processed.  
2036 1848 of 2717 files processed.  
2035 1838 of 2717 files processed.  
2034 1844 of 2717 files processed.  
2037 1854 of 2717 files processed.  
2036 1849 of 2717 files processed.  
2035 1839 of 2717 files processed.  
2034 1845 of 2717 files processed.  
2037 1855 of 2717 files processed.  
2036 1850 of 2717 files processed.  
2035 1840 of 2717 files processed.  
2034 1846 of 2717 files processed.  
2037 1856 of 2717 files processed.  
2036 1851 of 2717 files processed.  
2035 1841 of 2717 files processed.  
2034 1847 of 2717 files processed.  
2037 1857 of 2717 files processed.  
2035 1842 of 2717 files processed.  
2036 1852 of 2717 files processed.  
2034 1848 of 2717 files processed.  
2037 1858 of 2717 files processed.  
2035 1843 of 2717 files processed.  
2036 1853 of 2717 files processed.  
2034 1849 of 2717 files processed.  
2035 1844 of 2717 files processed.  
2036 1854 of 2717 files processed.  
2037 1859 of 2717 files processed.  
2034 1850 of 2717 files processed.  
2036 1855 of 2717 files processed.  
2035 1845 of 2717 files processed.  
2037 1860 of 2717 files processed.  
2034 1851 of 2717 files processed.  
2036 1856 of 2717 files processed.  
2037 1861 of 2717 files processed.  
2035 1846 of 2717 files processed.  
2034 1852 of 2717 files processed.  
2036 1857 of 2717 files processed.  
2035 1847 of 2717 files processed.  
2037 1862 of 2717 files processed.  
2034 1853 of 2717 files processed.  
2036 1858 of 2717 files processed.  
2037 1863 of 2717 files processed.  
2035 1848 of 2717 files processed.  
2034 1854 of 2717 files processed.  
2035 1849 of 2717 files processed.  
2036 1859 of 2717 files processed.  
2037 1864 of 2717 files processed.  
2034 1855 of 2717 files processed.  
2035 1850 of 2717 files processed.  
2034 1856 of 2717 files processed.  
2037 1865 of 2717 files processed.  
2036 1860 of 2717 files processed.  
2035 1851 of 2717 files processed.  
2037 1866 of 2717 files processed.  
2034 1857 of 2717 files processed.  
2036 1861 of 2717 files processed.  
2035 1852 of 2717 files processed.  
2037 1867 of 2717 files processed.  
2034 1858 of 2717 files processed.  
2036 1862 of 2717 files processed.  
2034 1859 of 2717 files processed.  
2035 1853 of 2717 files processed.  
2037 1868 of 2717 files processed.  
2036 1863 of 2717 files processed.  
2034 1860 of 2717 files processed.  
2035 1854 of 2717 files processed.  
2037 1869 of 2717 files processed.  
2036 1864 of 2717 files processed.  
2034 1861 of 2717 files processed.  
2035 1855 of 2717 files processed.  
2037 1870 of 2717 files processed.  
2036 1865 of 2717 files processed.  
2034 1862 of 2717 files processed.

2035 1856 of 2717 files processed.  
2037 1871 of 2717 files processed.  
2036 1866 of 2717 files processed.  
2034 1863 of 2717 files processed.  
2035 1857 of 2717 files processed.  
2037 1872 of 2717 files processed.  
2036 1867 of 2717 files processed.  
2034 1864 of 2717 files processed.  
2037 1873 of 2717 files processed.  
2035 1858 of 2717 files processed.  
2034 1865 of 2717 files processed.  
2036 1868 of 2717 files processed.  
2037 1874 of 2717 files processed.  
2035 1859 of 2717 files processed.  
2034 1866 of 2717 files processed.  
2036 1869 of 2717 files processed.  
2037 1875 of 2717 files processed.  
2035 1860 of 2717 files processed.  
2034 1867 of 2717 files processed.  
2037 1876 of 2717 files processed.  
2036 1870 of 2717 files processed.  
2035 1861 of 2717 files processed.  
2034 1868 of 2717 files processed.  
2037 1877 of 2717 files processed.  
2036 1871 of 2717 files processed.  
2035 1862 of 2717 files processed.  
2034 1869 of 2717 files processed.  
2037 1878 of 2717 files processed.  
2035 1863 of 2717 files processed.  
2036 1872 of 2717 files processed.  
2034 1870 of 2717 files processed.  
2037 1879 of 2717 files processed.  
2035 1864 of 2717 files processed.  
2036 1873 of 2717 files processed.  
2037 1880 of 2717 files processed.  
2034 1871 of 2717 files processed.  
2035 1865 of 2717 files processed.  
2034 1872 of 2717 files processed.  
2036 1874 of 2717 files processed.  
2037 1881 of 2717 files processed.  
2035 1866 of 2717 files processed.  
2034 1873 of 2717 files processed.  
2037 1882 of 2717 files processed.  
2036 1875 of 2717 files processed.  
2035 1867 of 2717 files processed.  
2034 1874 of 2717 files processed.  
2037 1883 of 2717 files processed.  
2036 1876 of 2717 files processed.  
2035 1868 of 2717 files processed.  
2034 1875 of 2717 files processed.  
2037 1884 of 2717 files processed.  
2036 1877 of 2717 files processed.  
2034 1876 of 2717 files processed.  
2035 1869 of 2717 files processed.  
2037 1885 of 2717 files processed.  
2036 1878 of 2717 files processed.  
2034 1877 of 2717 files processed.  
2037 1886 of 2717 files processed.  
2036 1879 of 2717 files processed.  
2035 1870 of 2717 files processed.  
2034 1878 of 2717 files processed.  
2037 1887 of 2717 files processed.  
2035 1871 of 2717 files processed.  
2036 1880 of 2717 files processed.  
2034 1879 of 2717 files processed.  
2035 1872 of 2717 files processed.  
2037 1888 of 2717 files processed.  
2036 1881 of 2717 files processed.  
2034 1880 of 2717 files processed.  
2035 1873 of 2717 files processed.  
2037 1889 of 2717 files processed.  
2036 1882 of 2717 files processed.  
2034 1881 of 2717 files processed.  
2035 1874 of 2717 files processed.  
2037 1890 of 2717 files processed.  
2036 1883 of 2717 files processed.  
2034 1882 of 2717 files processed.



.....  
2035 1895 of 2717 files processed.  
2034 1901 of 2717 files processed.  
2036 1904 of 2717 files processed.  
2037 1910 of 2717 files processed.  
2034 1902 of 2717 files processed.  
2035 1896 of 2717 files processed.  
2036 1905 of 2717 files processed.  
2034 1903 of 2717 files processed.  
2037 1911 of 2717 files processed.  
2035 1897 of 2717 files processed.  
2036 1906 of 2717 files processed.  
2037 1912 of 2717 files processed.  
2034 1904 of 2717 files processed.  
2035 1898 of 2717 files processed.  
2037 1913 of 2717 files processed.  
2034 1905 of 2717 files processed.  
2036 1907 of 2717 files processed.  
2035 1899 of 2717 files processed.  
2037 1914 of 2717 files processed.  
2034 1906 of 2717 files processed.  
2036 1908 of 2717 files processed.  
2035 1900 of 2717 files processed.  
2037 1915 of 2717 files processed.  
2036 1909 of 2717 files processed.  
2034 1907 of 2717 files processed.  
2035 1901 of 2717 files processed.  
2037 1916 of 2717 files processed.  
2036 1910 of 2717 files processed.  
2034 1908 of 2717 files processed.  
2035 1902 of 2717 files processed.  
2037 1917 of 2717 files processed.  
2036 1911 of 2717 files processed.  
2034 1909 of 2717 files processed.  
2036 1912 of 2717 files processed.  
2037 1918 of 2717 files processed.  
2035 1903 of 2717 files processed.  
2034 1910 of 2717 files processed.  
2036 1913 of 2717 files processed.  
2037 1919 of 2717 files processed.  
2035 1904 of 2717 files processed.  
2034 1911 of 2717 files processed.  
2036 1914 of 2717 files processed.  
2035 1905 of 2717 files processed.  
2037 1920 of 2717 files processed.  
2036 1915 of 2717 files processed.  
2035 1906 of 2717 files processed.  
2034 1912 of 2717 files processed.  
2037 1921 of 2717 files processed.  
2036 1916 of 2717 files processed.  
2037 1922 of 2717 files processed.  
2035 1907 of 2717 files processed.  
2034 1913 of 2717 files processed.  
2036 1917 of 2717 files processed.  
2037 1923 of 2717 files processed.  
2034 1914 of 2717 files processed.  
2035 1908 of 2717 files processed.  
2036 1918 of 2717 files processed.  
2037 1924 of 2717 files processed.  
2034 1915 of 2717 files processed.  
2035 1909 of 2717 files processed.  
2036 1919 of 2717 files processed.  
2037 1925 of 2717 files processed.  
2034 1916 of 2717 files processed.  
2036 1920 of 2717 files processed.  
2035 1910 of 2717 files processed.  
2037 1926 of 2717 files processed.  
2034 1917 of 2717 files processed.  
2036 1921 of 2717 files processed.  
2035 1911 of 2717 files processed.  
2037 1927 of 2717 files processed.  
2036 1922 of 2717 files processed.  
2034 1918 of 2717 files processed.  
2035 1912 of 2717 files processed.  
2037 1928 of 2717 files processed.  
2034 1919 of 2717 files processed.  
2036 1923 of 2717 files processed.  
2035 1913 of 2717 files processed

2035 1919 of 2717 files processed.  
2037 1929 of 2717 files processed.  
2034 1920 of 2717 files processed.  
2036 1924 of 2717 files processed.  
2035 1914 of 2717 files processed.  
2037 1930 of 2717 files processed.  
2034 1921 of 2717 files processed.  
2036 1925 of 2717 files processed.  
2035 1915 of 2717 files processed.  
2037 1931 of 2717 files processed.  
2034 1922 of 2717 files processed.  
2036 1926 of 2717 files processed.  
2035 1916 of 2717 files processed.  
2037 1932 of 2717 files processed.  
2034 1923 of 2717 files processed.  
2036 1927 of 2717 files processed.  
2035 1917 of 2717 files processed.  
2037 1933 of 2717 files processed.  
2034 1924 of 2717 files processed.  
2036 1928 of 2717 files processed.  
2037 1934 of 2717 files processed.  
2035 1918 of 2717 files processed.  
2034 1925 of 2717 files processed.  
2036 1929 of 2717 files processed.  
2037 1935 of 2717 files processed.  
2035 1919 of 2717 files processed.  
2036 1930 of 2717 files processed.  
2034 1926 of 2717 files processed.  
2035 1920 of 2717 files processed.  
2037 1936 of 2717 files processed.  
2036 1931 of 2717 files processed.  
2034 1927 of 2717 files processed.  
2035 1921 of 2717 files processed.  
2036 1932 of 2717 files processed.  
2034 1928 of 2717 files processed.  
2037 1937 of 2717 files processed.  
2036 1933 of 2717 files processed.  
2035 1922 of 2717 files processed.  
2034 1929 of 2717 files processed.  
2037 1938 of 2717 files processed.  
2036 1934 of 2717 files processed.  
2035 1923 of 2717 files processed.  
2034 1930 of 2717 files processed.  
2037 1939 of 2717 files processed.  
2036 1935 of 2717 files processed.  
2035 1924 of 2717 files processed.  
2037 1940 of 2717 files processed.  
2034 1931 of 2717 files processed.  
2036 1936 of 2717 files processed.  
2035 1925 of 2717 files processed.  
2037 1941 of 2717 files processed.  
2034 1932 of 2717 files processed.  
2036 1937 of 2717 files processed.  
2035 1926 of 2717 files processed.  
2034 1933 of 2717 files processed.  
2037 1942 of 2717 files processed.  
2035 1927 of 2717 files processed.  
2036 1938 of 2717 files processed.  
2034 1934 of 2717 files processed.  
2037 1943 of 2717 files processed.  
2035 1928 of 2717 files processed.  
2037 1944 of 2717 files processed.  
2036 1939 of 2717 files processed.  
2034 1935 of 2717 files processed.  
2035 1929 of 2717 files processed.  
2037 1945 of 2717 files processed.  
2034 1936 of 2717 files processed.  
2036 1940 of 2717 files processed.  
2035 1930 of 2717 files processed.  
2037 1946 of 2717 files processed.  
2034 1937 of 2717 files processed.  
2036 1941 of 2717 files processed.  
2035 1931 of 2717 files processed.  
2034 1938 of 2717 files processed.  
2036 1942 of 2717 files processed.  
2037 1947 of 2717 files processed.  
2035 1932 of 2717 files processed.  
2034 1939 of 2717 files processed

2036 1943 of 2717 files processed.  
2037 1948 of 2717 files processed.  
2035 1933 of 2717 files processed.  
2034 1940 of 2717 files processed.  
2037 1949 of 2717 files processed.  
2036 1944 of 2717 files processed.  
2035 1934 of 2717 files processed.  
2034 1941 of 2717 files processed.  
2037 1950 of 2717 files processed.  
2036 1945 of 2717 files processed.  
2035 1935 of 2717 files processed.  
2034 1942 of 2717 files processed.  
2037 1951 of 2717 files processed.  
2036 1946 of 2717 files processed.  
2035 1936 of 2717 files processed.  
2034 1943 of 2717 files processed.  
2037 1952 of 2717 files processed.  
2036 1947 of 2717 files processed.  
2035 1937 of 2717 files processed.  
2034 1944 of 2717 files processed.  
2037 1953 of 2717 files processed.  
2036 1948 of 2717 files processed.  
2035 1938 of 2717 files processed.  
2037 1954 of 2717 files processed.  
2034 1945 of 2717 files processed.  
2036 1949 of 2717 files processed.  
2035 1939 of 2717 files processed.  
2034 1946 of 2717 files processed.  
2037 1955 of 2717 files processed.  
2035 1940 of 2717 files processed.  
2036 1950 of 2717 files processed.  
2037 1956 of 2717 files processed.  
2034 1947 of 2717 files processed.  
2035 1941 of 2717 files processed.  
2036 1951 of 2717 files processed.  
2034 1948 of 2717 files processed.  
2037 1957 of 2717 files processed.  
2035 1942 of 2717 files processed.  
2036 1952 of 2717 files processed.  
2035 1943 of 2717 files processed.  
2037 1958 of 2717 files processed.  
2034 1949 of 2717 files processed.  
2036 1953 of 2717 files processed.  
2037 1959 of 2717 files processed.  
2035 1944 of 2717 files processed.  
2034 1950 of 2717 files processed.  
2036 1954 of 2717 files processed.  
2037 1960 of 2717 files processed.  
2035 1945 of 2717 files processed.  
2036 1955 of 2717 files processed.  
2034 1951 of 2717 files processed.  
2037 1961 of 2717 files processed.  
2035 1946 of 2717 files processed.  
2034 1952 of 2717 files processed.  
2036 1956 of 2717 files processed.  
2037 1962 of 2717 files processed.  
2035 1947 of 2717 files processed.  
2034 1953 of 2717 files processed.  
2036 1957 of 2717 files processed.  
2037 1963 of 2717 files processed.  
2035 1948 of 2717 files processed.  
2036 1958 of 2717 files processed.  
2034 1954 of 2717 files processed.  
2035 1949 of 2717 files processed.  
2036 1959 of 2717 files processed.  
2037 1964 of 2717 files processed.  
2034 1955 of 2717 files processed.  
2035 1950 of 2717 files processed.  
2036 1960 of 2717 files processed.  
2037 1965 of 2717 files processed.  
2034 1956 of 2717 files processed.  
2036 1961 of 2717 files processed.  
2037 1966 of 2717 files processed.  
2035 1951 of 2717 files processed.  
2034 1957 of 2717 files processed.  
2037 1967 of 2717 files processed.  
2035 1952 of 2717 files processed.

2035 1952 of 2717 files processed.  
2036 1962 of 2717 files processed.  
2034 1958 of 2717 files processed.  
2037 1968 of 2717 files processed.  
2035 1953 of 2717 files processed.  
2036 1963 of 2717 files processed.  
2034 1959 of 2717 files processed.  
2037 1969 of 2717 files processed.  
2035 1954 of 2717 files processed.  
2036 1964 of 2717 files processed.  
2034 1960 of 2717 files processed.  
2037 1970 of 2717 files processed.  
2035 1955 of 2717 files processed.  
2036 1965 of 2717 files processed.  
2034 1961 of 2717 files processed.  
2037 1971 of 2717 files processed.  
2035 1956 of 2717 files processed.  
2034 1962 of 2717 files processed.  
2036 1966 of 2717 files processed.  
2037 1972 of 2717 files processed.  
2035 1957 of 2717 files processed.  
2034 1963 of 2717 files processed.  
2036 1967 of 2717 files processed.  
2037 1973 of 2717 files processed.  
2035 1958 of 2717 files processed.  
2034 1964 of 2717 files processed.  
2036 1968 of 2717 files processed.  
2037 1974 of 2717 files processed.  
2034 1965 of 2717 files processed.  
2035 1959 of 2717 files processed.  
2036 1969 of 2717 files processed.  
2037 1975 of 2717 files processed.  
2034 1966 of 2717 files processed.  
2035 1960 of 2717 files processed.  
2036 1970 of 2717 files processed.  
2037 1976 of 2717 files processed.  
2034 1967 of 2717 files processed.  
2035 1961 of 2717 files processed.  
2036 1971 of 2717 files processed.  
2037 1977 of 2717 files processed.  
2034 1968 of 2717 files processed.  
2035 1962 of 2717 files processed.  
2036 1972 of 2717 files processed.  
2037 1978 of 2717 files processed.  
2034 1969 of 2717 files processed.  
2035 1963 of 2717 files processed.  
2037 1979 of 2717 files processed.  
2036 1973 of 2717 files processed.  
2034 1970 of 2717 files processed.  
2035 1964 of 2717 files processed.  
2037 1980 of 2717 files processed.  
2036 1974 of 2717 files processed.  
2034 1971 of 2717 files processed.  
2035 1965 of 2717 files processed.  
2037 1981 of 2717 files processed.  
2034 1972 of 2717 files processed.  
2036 1975 of 2717 files processed.  
2035 1966 of 2717 files processed.  
2037 1982 of 2717 files processed.  
2034 1973 of 2717 files processed.  
2036 1976 of 2717 files processed.  
2037 1983 of 2717 files processed.  
2035 1967 of 2717 files processed.  
2036 1977 of 2717 files processed.  
2037 1984 of 2717 files processed.  
2034 1974 of 2717 files processed.  
2035 1968 of 2717 files processed.  
2036 1978 of 2717 files processed.  
2037 1985 of 2717 files processed.  
2034 1975 of 2717 files processed.  
2035 1969 of 2717 files processed.  
2037 1986 of 2717 files processed.  
2036 1979 of 2717 files processed.  
2034 1976 of 2717 files processed.  
2035 1970 of 2717 files processed.  
2037 1987 of 2717 files processed.  
2034 1977 of 2717 files processed.











2034 2074 of 2717 files processed.  
2036 2076 of 2717 files processed.  
2037 2086 of 2717 files processed.  
2035 2066 of 2717 files processed.  
2034 2075 of 2717 files processed.  
2036 2077 of 2717 files processed.  
2037 2087 of 2717 files processed.  
2035 2067 of 2717 files processed.  
2036 2078 of 2717 files processed.  
2034 2076 of 2717 files processed.  
2037 2088 of 2717 files processed.  
2035 2068 of 2717 files processed.  
2036 2079 of 2717 files processed.  
2034 2077 of 2717 files processed.  
2037 2089 of 2717 files processed.  
2035 2069 of 2717 files processed.  
2036 2080 of 2717 files processed.  
2034 2078 of 2717 files processed.  
2035 2070 of 2717 files processed.  
2037 2090 of 2717 files processed.  
2036 2081 of 2717 files processed.  
2034 2079 of 2717 files processed.  
2035 2071 of 2717 files processed.  
2037 2091 of 2717 files processed.  
2034 2080 of 2717 files processed.  
2036 2082 of 2717 files processed.  
2035 2072 of 2717 files processed.  
2037 2092 of 2717 files processed.  
2034 2081 of 2717 files processed.  
2036 2083 of 2717 files processed.  
2035 2073 of 2717 files processed.  
2037 2093 of 2717 files processed.  
2036 2084 of 2717 files processed.  
2035 2074 of 2717 files processed.  
2034 2082 of 2717 files processed.  
2037 2094 of 2717 files processed.  
2036 2085 of 2717 files processed.  
2034 2083 of 2717 files processed.  
2037 2095 of 2717 files processed.  
2035 2075 of 2717 files processed.  
2036 2086 of 2717 files processed.  
2034 2084 of 2717 files processed.  
2035 2076 of 2717 files processed.  
2037 2096 of 2717 files processed.  
2036 2087 of 2717 files processed.  
2034 2085 of 2717 files processed.  
2037 2097 of 2717 files processed.  
2035 2077 of 2717 files processed.  
2036 2088 of 2717 files processed.  
2034 2086 of 2717 files processed.  
2035 2078 of 2717 files processed.  
2037 2098 of 2717 files processed.  
2036 2089 of 2717 files processed.  
2034 2087 of 2717 files processed.  
2037 2099 of 2717 files processed.  
2035 2079 of 2717 files processed.  
2036 2090 of 2717 files processed.  
2034 2088 of 2717 files processed.  
2037 2100 of 2717 files processed.  
2035 2080 of 2717 files processed.  
2036 2091 of 2717 files processed.  
2034 2089 of 2717 files processed.  
2037 2101 of 2717 files processed.  
2036 2092 of 2717 files processed.  
2035 2081 of 2717 files processed.  
2034 2090 of 2717 files processed.  
2037 2102 of 2717 files processed.  
2034 2091 of 2717 files processed.  
2035 2082 of 2717 files processed.  
2036 2093 of 2717 files processed.  
2037 2103 of 2717 files processed.  
2034 2092 of 2717 files processed.  
2036 2094 of 2717 files processed.  
2037 2104 of 2717 files processed.  
2035 2083 of 2717 files processed.  
2034 2093 of 2717 files processed.  
2036 2095 of 2717 files processed.

























2036 2326 of 2717 files processed.  
2035 2315 of 2717 files processed.  
2034 2323 of 2717 files processed.  
2036 2327 of 2717 files processed.  
2037 2338 of 2717 files processed.  
2035 2316 of 2717 files processed.  
2036 2328 of 2717 files processed.  
2034 2324 of 2717 files processed.  
2037 2339 of 2717 files processed.  
2034 2325 of 2717 files processed.  
2036 2329 of 2717 files processed.  
2037 2340 of 2717 files processed.  
2035 2317 of 2717 files processed.  
2035 2318 of 2717 files processed.  
2034 2326 of 2717 files processed.  
2037 2341 of 2717 files processed.  
2036 2330 of 2717 files processed.  
2035 2319 of 2717 files processed.  
2034 2327 of 2717 files processed.  
2037 2342 of 2717 files processed.  
2036 2331 of 2717 files processed.  
2035 2320 of 2717 files processed.  
2034 2328 of 2717 files processed.  
2036 2332 of 2717 files processed.  
2037 2343 of 2717 files processed.  
2035 2321 of 2717 files processed.  
2034 2329 of 2717 files processed.  
2036 2333 of 2717 files processed.  
2037 2344 of 2717 files processed.  
2034 2330 of 2717 files processed.  
2035 2322 of 2717 files processed.  
2036 2334 of 2717 files processed.  
2037 2345 of 2717 files processed.  
2034 2331 of 2717 files processed.  
2035 2323 of 2717 files processed.  
2036 2335 of 2717 files processed.  
2037 2346 of 2717 files processed.  
2034 2332 of 2717 files processed.  
2036 2336 of 2717 files processed.  
2035 2324 of 2717 files processed.  
2037 2347 of 2717 files processed.  
2034 2333 of 2717 files processed.  
2036 2337 of 2717 files processed.  
2035 2325 of 2717 files processed.  
2037 2348 of 2717 files processed.  
2034 2334 of 2717 files processed.  
2036 2338 of 2717 files processed.  
2035 2326 of 2717 files processed.  
2037 2349 of 2717 files processed.  
2035 2327 of 2717 files processed.  
2034 2335 of 2717 files processed.  
2036 2339 of 2717 files processed.  
2037 2350 of 2717 files processed.  
2035 2328 of 2717 files processed.  
2036 2340 of 2717 files processed.  
2034 2336 of 2717 files processed.  
2037 2351 of 2717 files processed.  
2035 2329 of 2717 files processed.  
2036 2341 of 2717 files processed.  
2034 2337 of 2717 files processed.  
2037 2352 of 2717 files processed.  
2035 2330 of 2717 files processed.  
2036 2342 of 2717 files processed.  
2034 2338 of 2717 files processed.  
2036 2343 of 2717 files processed.  
2037 2353 of 2717 files processed.  
2035 2331 of 2717 files processed.  
2034 2339 of 2717 files processed.  
2037 2354 of 2717 files processed.  
2036 2344 of 2717 files processed.  
2035 2332 of 2717 files processed.  
2034 2340 of 2717 files processed.  
2037 2355 of 2717 files processed.  
2036 2345 of 2717 files processed.  
2035 2333 of 2717 files processed.  
2034 2341 of 2717 files processed.  
2037 2356 of 2717 files processed.  
-----









2036 2423 of 2717 files processed.  
2034 2419 of 2717 files processed.  
2037 2434 of 2717 files processed.  
2035 2411 of 2717 files processed.  
2036 2424 of 2717 files processed.  
2034 2420 of 2717 files processed.  
2037 2435 of 2717 files processed.  
2035 2412 of 2717 files processed.  
2036 2425 of 2717 files processed.  
2037 2436 of 2717 files processed.  
2034 2421 of 2717 files processed.  
2036 2426 of 2717 files processed.  
2035 2413 of 2717 files processed.  
2037 2437 of 2717 files processed.  
2034 2422 of 2717 files processed.  
2036 2427 of 2717 files processed.  
2035 2414 of 2717 files processed.  
2037 2438 of 2717 files processed.  
2034 2423 of 2717 files processed.  
2037 2439 of 2717 files processed.  
2036 2428 of 2717 files processed.  
2034 2424 of 2717 files processed.  
2035 2415 of 2717 files processed.  
2036 2429 of 2717 files processed.  
2035 2416 of 2717 files processed.  
2037 2440 of 2717 files processed.  
2034 2425 of 2717 files processed.  
2036 2430 of 2717 files processed.  
2035 2417 of 2717 files processed.  
2037 2441 of 2717 files processed.  
2034 2426 of 2717 files processed.  
2036 2431 of 2717 files processed.  
2035 2418 of 2717 files processed.  
2034 2427 of 2717 files processed.  
2037 2442 of 2717 files processed.  
2036 2432 of 2717 files processed.  
2035 2419 of 2717 files processed.  
2034 2428 of 2717 files processed.  
2037 2443 of 2717 files processed.  
2036 2433 of 2717 files processed.  
2034 2429 of 2717 files processed.  
2037 2444 of 2717 files processed.  
2035 2420 of 2717 files processed.  
2036 2434 of 2717 files processed.  
2035 2421 of 2717 files processed.  
2037 2445 of 2717 files processed.  
2034 2430 of 2717 files processed.  
2036 2435 of 2717 files processed.  
2035 2422 of 2717 files processed.  
2037 2446 of 2717 files processed.  
2034 2431 of 2717 files processed.  
2036 2436 of 2717 files processed.  
2037 2447 of 2717 files processed.  
2034 2432 of 2717 files processed.  
2035 2423 of 2717 files processed.  
2036 2437 of 2717 files processed.  
2037 2448 of 2717 files processed.  
2034 2433 of 2717 files processed.  
2035 2424 of 2717 files processed.  
2036 2438 of 2717 files processed.  
2037 2449 of 2717 files processed.  
2034 2434 of 2717 files processed.  
2035 2425 of 2717 files processed.  
2037 2450 of 2717 files processed.  
2036 2439 of 2717 files processed.  
2034 2435 of 2717 files processed.  
2035 2426 of 2717 files processed.  
2034 2436 of 2717 files processed.  
2036 2440 of 2717 files processed.  
2037 2451 of 2717 files processed.  
2035 2427 of 2717 files processed.  
2036 2441 of 2717 files processed.  
2037 2452 of 2717 files processed.  
2034 2437 of 2717 files processed.  
2035 2428 of 2717 files processed.  
2034 2438 of 2717 files processed.  
2036 2442 of 2717 files processed.



2034 2457 of 2717 files processed.  
2037 2473 of 2717 files processed.  
2035 2449 of 2717 files processed.  
2036 2462 of 2717 files processed.  
2034 2458 of 2717 files processed.  
2037 2474 of 2717 files processed.  
2036 2463 of 2717 files processed.  
2035 2450 of 2717 files processed.  
2034 2459 of 2717 files processed.  
2037 2475 of 2717 files processed.  
2036 2464 of 2717 files processed.  
2035 2451 of 2717 files processed.  
2034 2460 of 2717 files processed.  
2037 2476 of 2717 files processed.  
2036 2465 of 2717 files processed.  
2035 2452 of 2717 files processed.  
2037 2477 of 2717 files processed.  
2034 2461 of 2717 files processed.  
2036 2466 of 2717 files processed.  
2035 2453 of 2717 files processed.  
2037 2478 of 2717 files processed.  
2034 2462 of 2717 files processed.  
2036 2467 of 2717 files processed.  
2035 2454 of 2717 files processed.  
2037 2479 of 2717 files processed.  
2034 2463 of 2717 files processed.  
2036 2468 of 2717 files processed.  
2035 2455 of 2717 files processed.  
2037 2480 of 2717 files processed.  
2034 2464 of 2717 files processed.  
2035 2456 of 2717 files processed.  
2036 2469 of 2717 files processed.  
2037 2481 of 2717 files processed.  
2035 2457 of 2717 files processed.  
2036 2470 of 2717 files processed.  
2034 2465 of 2717 files processed.  
2037 2482 of 2717 files processed.  
2034 2466 of 2717 files processed.  
2036 2471 of 2717 files processed.  
2035 2458 of 2717 files processed.  
2037 2483 of 2717 files processed.  
2034 2467 of 2717 files processed.  
2036 2472 of 2717 files processed.  
2035 2459 of 2717 files processed.  
2037 2484 of 2717 files processed.  
2034 2468 of 2717 files processed.  
2036 2473 of 2717 files processed.  
2035 2460 of 2717 files processed.  
2037 2485 of 2717 files processed.  
2034 2469 of 2717 files processed.  
2036 2474 of 2717 files processed.  
2035 2461 of 2717 files processed.  
2036 2475 of 2717 files processed.  
2037 2486 of 2717 files processed.  
2034 2470 of 2717 files processed.  
2035 2462 of 2717 files processed.  
2037 2487 of 2717 files processed.  
2036 2476 of 2717 files processed.  
2034 2471 of 2717 files processed.  
2035 2463 of 2717 files processed.  
2037 2488 of 2717 files processed.  
2034 2472 of 2717 files processed.  
2036 2477 of 2717 files processed.  
2035 2464 of 2717 files processed.  
2036 2478 of 2717 files processed.  
2037 2489 of 2717 files processed.  
2034 2473 of 2717 files processed.  
2035 2465 of 2717 files processed.  
2034 2474 of 2717 files processed.  
2035 2466 of 2717 files processed.  
2036 2479 of 2717 files processed.  
2037 2490 of 2717 files processed.  
2034 2475 of 2717 files processed.  
2035 2467 of 2717 files processed.  
2037 2491 of 2717 files processed.  
2036 2480 of 2717 files processed.  
2034 2476 of 2717 files processed.

2037 2492 of 2717 files processed.  
2035 2468 of 2717 files processed.  
2036 2481 of 2717 files processed.  
2034 2477 of 2717 files processed.  
2037 2493 of 2717 files processed.  
2035 2469 of 2717 files processed.  
2036 2482 of 2717 files processed.  
2034 2478 of 2717 files processed.  
2037 2494 of 2717 files processed.  
2035 2470 of 2717 files processed.  
2036 2483 of 2717 files processed.  
2034 2479 of 2717 files processed.  
2037 2495 of 2717 files processed.  
2036 2484 of 2717 files processed.  
2035 2471 of 2717 files processed.  
2037 2496 of 2717 files processed.  
2034 2480 of 2717 files processed.  
2036 2485 of 2717 files processed.  
2035 2472 of 2717 files processed.  
2037 2497 of 2717 files processed.  
2036 2486 of 2717 files processed.  
2034 2481 of 2717 files processed.  
2035 2473 of 2717 files processed.  
2037 2498 of 2717 files processed.  
2036 2487 of 2717 files processed.  
2034 2482 of 2717 files processed.  
2035 2474 of 2717 files processed.  
2036 2488 of 2717 files processed.  
2034 2483 of 2717 files processed.  
2035 2475 of 2717 files processed.  
2037 2499 of 2717 files processed.  
2034 2484 of 2717 files processed.  
2036 2489 of 2717 files processed.  
2037 2500 of 2717 files processed.  
2035 2476 of 2717 files processed.  
2036 2490 of 2717 files processed.  
2034 2485 of 2717 files processed.  
2037 2501 of 2717 files processed.  
2035 2477 of 2717 files processed.  
2037 2502 of 2717 files processed.  
2036 2491 of 2717 files processed.  
2034 2486 of 2717 files processed.  
2035 2478 of 2717 files processed.  
2036 2492 of 2717 files processed.  
2034 2487 of 2717 files processed.  
2035 2479 of 2717 files processed.  
2037 2503 of 2717 files processed.  
2035 2480 of 2717 files processed.  
2037 2504 of 2717 files processed.  
2034 2488 of 2717 files processed.  
2036 2493 of 2717 files processed.  
2037 2505 of 2717 files processed.  
2034 2489 of 2717 files processed.  
2035 2481 of 2717 files processed.  
2036 2494 of 2717 files processed.  
2035 2482 of 2717 files processed.  
2034 2490 of 2717 files processed.  
2037 2506 of 2717 files processed.  
2036 2495 of 2717 files processed.  
2034 2491 of 2717 files processed.  
2035 2483 of 2717 files processed.  
2037 2507 of 2717 files processed.  
2036 2496 of 2717 files processed.  
2034 2492 of 2717 files processed.  
2035 2484 of 2717 files processed.  
2037 2508 of 2717 files processed.  
2036 2497 of 2717 files processed.  
2034 2493 of 2717 files processed.  
2035 2485 of 2717 files processed.  
2037 2509 of 2717 files processed.  
2036 2498 of 2717 files processed.  
2034 2494 of 2717 files processed.  
2035 2486 of 2717 files processed.  
2037 2510 of 2717 files processed.  
2036 2499 of 2717 files processed.  
2034 2495 of 2717 files processed.  
2035 2487 of 2717 files processed.













2031 2621 of 2717 files processed.  
2035 2600 of 2717 files processed.  
2034 2611 of 2717 files processed.  
2037 2628 of 2717 files processed.  
2036 2618 of 2717 files processed.  
2035 2601 of 2717 files processed.  
2034 2612 of 2717 files processed.  
2037 2629 of 2717 files processed.  
2035 2602 of 2717 files processed.  
2036 2619 of 2717 files processed.  
2034 2613 of 2717 files processed.  
2035 2603 of 2717 files processed.  
2037 2630 of 2717 files processed.  
2036 2620 of 2717 files processed.  
2034 2614 of 2717 files processed.  
2035 2604 of 2717 files processed.  
2037 2631 of 2717 files processed.  
2036 2621 of 2717 files processed.  
2034 2615 of 2717 files processed.  
2035 2605 of 2717 files processed.  
2034 2616 of 2717 files processed.  
2037 2632 of 2717 files processed.  
2036 2622 of 2717 files processed.  
2035 2606 of 2717 files processed.  
2037 2633 of 2717 files processed.  
2034 2617 of 2717 files processed.  
2036 2623 of 2717 files processed.  
2037 2634 of 2717 files processed.  
2035 2607 of 2717 files processed.  
2034 2618 of 2717 files processed.  
2036 2624 of 2717 files processed.  
2037 2635 of 2717 files processed.  
2035 2608 of 2717 files processed.  
2034 2619 of 2717 files processed.  
2036 2625 of 2717 files processed.  
2037 2636 of 2717 files processed.  
2035 2609 of 2717 files processed.  
2034 2620 of 2717 files processed.  
2036 2626 of 2717 files processed.  
2037 2637 of 2717 files processed.  
2035 2610 of 2717 files processed.  
2036 2627 of 2717 files processed.  
2034 2621 of 2717 files processed.  
2035 2611 of 2717 files processed.  
2037 2638 of 2717 files processed.  
2036 2628 of 2717 files processed.  
2034 2622 of 2717 files processed.  
2035 2612 of 2717 files processed.  
2037 2639 of 2717 files processed.  
2036 2629 of 2717 files processed.  
2034 2623 of 2717 files processed.  
2035 2613 of 2717 files processed.  
2037 2640 of 2717 files processed.  
2036 2630 of 2717 files processed.  
2034 2624 of 2717 files processed.  
2035 2614 of 2717 files processed.  
2037 2641 of 2717 files processed.  
2036 2631 of 2717 files processed.  
2034 2625 of 2717 files processed.  
2035 2615 of 2717 files processed.  
2036 2632 of 2717 files processed.  
2037 2642 of 2717 files processed.  
2034 2626 of 2717 files processed.  
2035 2616 of 2717 files processed.  
2037 2643 of 2717 files processed.  
2036 2633 of 2717 files processed.  
2034 2627 of 2717 files processed.  
2035 2617 of 2717 files processed.  
2037 2644 of 2717 files processed.  
2036 2634 of 2717 files processed.  
2034 2628 of 2717 files processed.  
2035 2618 of 2717 files processed.  
2037 2645 of 2717 files processed.  
2036 2635 of 2717 files processed.  
2034 2629 of 2717 files processed.  
2035 2619 of 2717 files processed.  
2037 2646 of 2717 files processed.









```
2035 2698 of 2717 files processed.  
2036 2716 of 2717 files processed.  
2034 2709 of 2717 files processed.  
2035 2699 of 2717 files processed.  
2036 2717 of 2717 files processed.  
2034 2710 of 2717 files processed.  
2035 2700 of 2717 files processed.  
2035 2701 of 2717 files processed.  
2034 2711 of 2717 files processed.  
2035 2702 of 2717 files processed.  
2034 2712 of 2717 files processed.  
2035 2703 of 2717 files processed.  
2034 2713 of 2717 files processed.  
2035 2704 of 2717 files processed.  
2034 2714 of 2717 files processed.  
2035 2705 of 2717 files processed.  
2034 2715 of 2717 files processed.  
2035 2706 of 2717 files processed.  
2034 2716 of 2717 files processed.  
2035 2707 of 2717 files processed.  
2034 2717 of 2717 files processed.  
2035 2708 of 2717 files processed.  
2035 2709 of 2717 files processed.  
2035 2710 of 2717 files processed.  
2035 2711 of 2717 files processed.  
2035 2712 of 2717 files processed.  
2035 2713 of 2717 files processed.  
2035 2714 of 2717 files processed.  
2035 2715 of 2717 files processed.  
2035 2716 of 2717 files processed.  
2035 2717 of 2717 files processed.  
Elapsed time: 3.79 hours.
```

In [16]:

```
# Load all generated npz files  
import scipy  
from scipy.sparse import csr_matrix  
from sklearn.preprocessing import normalize  
bytebigram_vect_1 =  
scipy.sparse.load_npz('bytebigram_2034.npz')#normalize(scipy.sparse.load_npz('bytebigram_2034.npz'))  
axis = 0  
bytebigram_vect_2 =  
scipy.sparse.load_npz('bytebigram_2035.npz')#normalize(scipy.sparse.load_npz('bytebigram_2035.npz'))  
axis = 0  
bytebigram_vect_3 =  
scipy.sparse.load_npz('bytebigram_2036.npz')#normalize(scipy.sparse.load_npz('bytebigram_2036.npz'))  
axis = 0  
bytebigram_vect_4 =  
scipy.sparse.load_npz('bytebigram_2037.npz')#normalize(scipy.sparse.load_npz('bytebigram_2037.npz'))  
axis = 0  
# bytebigram_vect_5 = normalize(scipy.sparse.load_npz('bytebigram_8767.npz'), axis = 0)  
# bytebigram_vect_6 = normalize(scipy.sparse.load_npz('bytebigram_8768.npz'), axis = 0)
```

In [17]:

```
# Merge all npz files  
from scipy.sparse import vstack  
bytebigram_vect_7 = vstack((bytebigram_vect_1, bytebigram_vect_2, bytebigram_vect_3,  
bytebigram_vect_4))  
#, bytebigram_vect_5, bytebigram_vect_6  
  
bytebigram_vect_7
```

Out[17]:

```
<10868x66306 sparse matrix of type '<class 'numpy.float64'>'  
with 502109160 stored elements in Compressed Sparse Row format>
```

In [19]:

```
def imp_features(data, features, keep):  
    ...  
    Collect important features using Random Forest Classifier
```

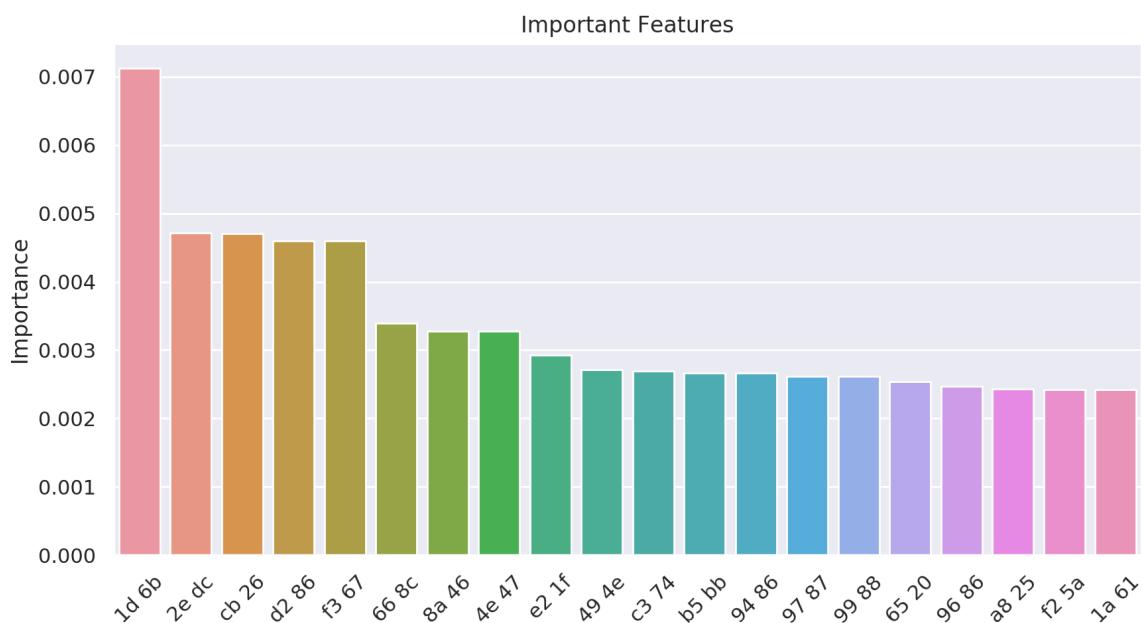
```

rf = RandomForestClassifier(n_estimators = 100, n_jobs = -1)
rf.fit(data, result_y)
imp_feature_indx = np.argsort(rf.feature_importances_)[::-1]
imp_value = np.take(rf.feature_importances_, imp_feature_indx[:20])
imp_feature_name = np.take(features, imp_feature_indx[:20])
sns.set()
plt.figure(figsize = (10, 5))
ax = sns.barplot(x = imp_feature_name, y = imp_value)
ax.set_xticklabels(labels = imp_feature_name, rotation = 45)
sns.set_palette(reversed(sns.color_palette("husl", 10)), 10)
plt.title('Important Features')
plt.xlabel('Feature Names')
plt.ylabel('Importance')
return imp_feature_indx[:keep]

```

In [20]:

```
# collect top 300 features from bigrams
byte_bi_indexes_1 = imp_features(normalize(bytebigram_vect_7, axis=0), all_keys, 300)
```



In [11]:

```
data_y.head()  
result_y = data_y
```

In [21]:

byte bi indexes 1

Out[21]:

```
array([14157, 40691, 17336, 46660, 8825, 4268, 28027, 12130, 21680,
       8953, 23402, 60240, 40185, 57251, 60461, 66125, 28941, 52525,
      11420, 42315, 1194, 52530, 33727, 23362, 38653, 32566, 5260,
     64991, 58535, 35728, 40950, 14695, 28918, 48334, 51662, 15622,
    20587, 13129, 23666, 10273, 12978, 43239, 6174, 51699, 20328,
   10730, 23968, 24865, 29100, 4670, 46546, 57986, 55909, 50088,
   45420, 20401, 14374, 9000, 6345, 38518, 3172, 28171, 35826,
  53280, 61323, 16859, 21166, 26032, 37345, 39969, 3842, 11095,
  16392, 10395, 63477, 63883, 28780, 21058, 12977, 29208, 9070,
  49561, 29935, 52039, 63461, 59348, 16332, 31246, 22790, 37400,
  15607, 25545, 41632, 54259, 63116, 21975, 13175, 43830, 4280,
  50343, 63837, 17033, 50978, 40625, 51685, 30258, 49647, 17990,
  25182, 19723, 42439, 17054, 21857, 51058, 37362, 11483, 54966,
  40121, 51962, 12827, 14293, 21291, 1697, 49129, 54347, 46965,
  55250, 30900, 63464, 60249, 57395, 64040, 38751, 57411, 55023,
```

```
48655, 38375, 8497, 60805, 41133, 48688, 11532, 15447, 14281,  
59816, 35801, 5376, 31260, 26502, 23967, 195, 65467, 20405,  
54663, 5833, 3539, 55511, 16739, 28444, 50815, 24802, 29339,  
6658, 31524, 40996, 56374, 43207, 66184, 31835, 26082, 41514,  
53820, 13307, 17392, 4019, 10927, 23215, 18124, 16169, 32522,  
28977, 31306, 23868, 60693, 2750, 16673, 28529, 31491, 51248,  
26920, 34232, 10409, 47688, 39505, 10089, 55965, 34194, 28953,  
9903, 62172, 53656, 57598, 28603, 18756, 18876, 10725, 185,  
55636, 9957, 36060, 17849, 65135, 9891, 59251, 64398, 39338,  
31351, 33420, 37802, 23257, 37528, 39836, 15681, 33561, 65231,  
18636, 51508, 46730, 39445, 12347, 21445, 26236, 32311, 9668,  
22755, 45469, 52047, 32420, 2785, 29259, 1472, 11015, 34507,  
45665, 57493, 64509, 3154, 42228, 39442, 57594, 14572, 34421,  
54613, 3936, 36977, 14803, 3871, 23382, 40434, 35758, 62469,  
50927, 18393, 24588, 60639, 28632, 20119, 49847, 11640, 2687,  
54923, 25605, 36430, 65573, 13473, 3597, 36518, 62401, 18907,  
46110, 54085, 21236, 5877, 43601, 18066, 14375, 39937, 14102,  
47992, 40069, 6096, 8713, 36692, 1492, 22573, 18440, 40746,  
43090, 23149, 24251])
```

In [22]:

```
# Save generated index file  
np.save('byte_bi_idx', byte_bi_indexes_1)
```

In [23]:

```
byte_bi_indexes = np.load('byte_bi_idx.npy')
```

In [24]:

```
byte_bi_indexes
```

Out[24]:

```
array([14157, 40691, 17336, 46660, 8825, 4268, 28027, 12130, 21680,  
8953, 23402, 60240, 40185, 57251, 60461, 66125, 28941, 52525,  
11420, 42315, 1194, 52530, 33727, 23362, 38653, 32566, 5260,  
64991, 58535, 35728, 40950, 14695, 28918, 48334, 51662, 15622,  
20587, 13129, 23666, 10273, 12978, 43239, 6174, 51699, 20328,  
10730, 23968, 24865, 29100, 4670, 46546, 57986, 55909, 50088,  
45420, 20401, 14374, 9000, 6345, 38518, 3172, 28171, 35826,  
53280, 61323, 16859, 21166, 26032, 37345, 39969, 3842, 11095,  
16392, 10395, 63477, 63883, 28780, 21058, 12977, 29208, 9070,  
49561, 29935, 52039, 63461, 59348, 16332, 31246, 22790, 37400,  
15607, 25545, 41632, 54259, 63116, 21975, 13175, 43830, 4280,  
50343, 63837, 17033, 50978, 40625, 51685, 30258, 49647, 17990,  
25182, 19723, 42439, 17054, 21857, 51058, 37362, 11483, 54966,  
40121, 51962, 12827, 14293, 21291, 1697, 49129, 54347, 46965,  
55250, 30900, 63464, 60249, 57395, 64040, 38751, 57411, 55023,  
48655, 38375, 8497, 60805, 41133, 48688, 11532, 15447, 14281,  
59816, 35801, 5376, 31260, 26502, 23967, 195, 65467, 20405,  
54663, 5833, 3539, 55511, 16739, 28444, 50815, 24802, 29339,  
6658, 31524, 40996, 56374, 43207, 66184, 31835, 26082, 41514,  
53820, 13307, 17392, 4019, 10927, 23215, 18124, 16169, 32522,  
28977, 31306, 23868, 60693, 2750, 16673, 28529, 31491, 51248,  
26920, 34232, 10409, 47688, 39505, 10089, 55965, 34194, 28953,  
9903, 62172, 53656, 57598, 28603, 18756, 18876, 10725, 185,  
55636, 9957, 36060, 17849, 65135, 9891, 59251, 64398, 39338,  
31351, 33420, 37802, 23257, 37528, 39836, 15681, 33561, 65231,  
18636, 51508, 46730, 39445, 12347, 21445, 26236, 32311, 9668,  
22755, 45469, 52047, 32420, 2785, 29259, 1472, 11015, 34507,  
45665, 57493, 64509, 3154, 42228, 39442, 57594, 14572, 34421,  
54613, 3936, 36977, 14803, 3871, 23382, 40434, 35758, 62469,  
50927, 18393, 24588, 60639, 28632, 20119, 49847, 11640, 2687,  
54923, 25605, 36430, 65573, 13473, 3597, 36518, 62401, 18907,  
46110, 54085, 21236, 5877, 43601, 18066, 14375, 39937, 14102,  
47992, 40069, 6096, 8713, 36692, 1492, 22573, 18440, 40746,  
43090, 23149, 24251])
```

In [25]:

```
# Pick top 300 features
```

```

top_byte_bi = np.zeros((10868, 0))
for i in byte_bi_indexes:
    sliced = bytebigram_vect_7[:, i].todense()
    top_byte_bi = np.hstack([top_byte_bi, sliced])

```

In [27]:

```

# Create a DataFrame using top 300 features
byte_bi_df = pd.SparseDataFrame(top_byte_bi, columns = np.take(all_keys, byte_bi_indexes))

```

In [28]:

```

# Save DataFrame into CSV
byte_bi_df.to_dense().to_csv('byte_bi.csv')

```

In [12]:

```

# Read csv and start processing the data
byte_bi_df = pd.read_csv('byte_bi.csv').drop('Unnamed: 0', axis = 1).fillna(0)

```

In [13]:

```
byte_bi_df
```

Out[13]:

	1d 6b	2e dc	cb 26	d2 86	f3 67	66 8c	8a 46	4e 47	e2 1f	49 4e	...	c3 8b	c4 0c	3b 38	3a 5c	3a dc	f0 8b	8b 4b	44 00	00 2d	7f 47
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	...	0.0	3.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	
1	3.0	3.0	5.0	1.0	1.0	8.0	2.0	287.0	1.0	282.0	...	5.0	11.0	1.0	1.0	7.0	7.0	30.0	15.0	3.0	
2	0.0	0.0	2.0	0.0	0.0	1.0	0.0	1.0	1.0	1.0	...	1.0	2.0	2.0	0.0	0.0	3.0	0.0	20.0	9.0	0.0
3	0.0	0.0	0.0	0.0	0.0	4.0	2.0	2.0	0.0	2.0	...	16.0	8.0	2.0	3.0	1.0	31.0	0.0	6.0	12.0	2.0
4	0.0	0.0	0.0	1.0	2.0	0.0	0.0	2.0	1.0	5.0	...	6.0	3.0	0.0	0.0	0.0	6.0	2.0	127.0	152.0	1.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
10863	1.0	1.0	1.0	2.0	0.0	2.0	0.0	3.0	0.0	0.0	...	3.0	5.0	1.0	1.0	1.0	30.0	0.0	9.0	18.0	3.0
10864	2.0	1.0	0.0	1.0	0.0	18.0	41.0	56.0	1.0	56.0	...	121.0	125.0	0.0	12.0	0.0	81.0	24.0	45.0	14.0	9.0
10865	2.0	2.0	1.0	1.0	3.0	14.0	0.0	29.0	0.0	42.0	...	75.0	114.0	6.0	4.0	6.0	42.0	2.0	300.0	299.0	2.0
10866	0.0	0.0	0.0	0.0	0.0	18.0	49.0	1.0	0.0	2.0	...	131.0	48.0	0.0	0.0	0.0	18.0	5.0	38.0	11.0	8.0
10867	0.0	0.0	0.0	0.0	0.0	0.0	1.0	67.0	1.0	85.0	...	0.0	5.0	1.0	1.0	0.0	2.0	0.0	38.0	20.0	0.0

10868 rows × 300 columns

In [14]:

```
byte_bi_df.head()
```

Out[14]:

	1d 6b	2e dc	cb 26	d2 86	f3 67	66 8c	8a 46	4e 47	e2 1f	49 4e	...	c3 8b	c4 0c	3b 38	3a 5c	3a dc	f0 8b	8b 4b	44 00	00 2d	7f 47
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	...	0.0	3.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	
1	3.0	3.0	5.0	1.0	1.0	8.0	2.0	287.0	1.0	282.0	...	5.0	11.0	1.0	1.0	7.0	7.0	30.0	15.0	3.0	
2	0.0	0.0	2.0	0.0	0.0	1.0	0.0	1.0	1.0	1.0	...	1.0	2.0	2.0	0.0	0.0	3.0	0.0	20.0	9.0	0.0
3	0.0	0.0	0.0	0.0	0.0	4.0	2.0	2.0	0.0	2.0	...	16.0	8.0	2.0	3.0	1.0	31.0	0.0	6.0	12.0	2.0
4	0.0	0.0	0.0	1.0	2.0	0.0	0.0	2.0	1.0	5.0	...	6.0	3.0	0.0	0.0	0.0	6.0	2.0	127.0	152.0	1.0

5 rows × 300 columns

In [15]:

```
result_x = byte_bi_df  
result_y = data_y  
result_x.tail()
```

Out[15]:

	1d 6b	2e dc	cb 26	d2 86	f3 67	66 8c	8a 46	4e 47	e2 1f	49 4e	...	c3 8b	c4 0c	3b 38	3a 5c	3a dc	f0 8b	8b 4b	44 00	00 2d	7f 47
10863	1.0	1.0	1.0	2.0	0.0	2.0	0.0	3.0	0.0	0.0	...	3.0	5.0	1.0	1.0	1.0	30.0	0.0	9.0	18.0	3.0
10864	2.0	1.0	0.0	1.0	0.0	18.0	41.0	56.0	1.0	56.0	...	121.0	125.0	0.0	12.0	0.0	81.0	24.0	45.0	14.0	9.0
10865	2.0	2.0	1.0	1.0	3.0	14.0	0.0	29.0	0.0	42.0	...	75.0	114.0	6.0	4.0	6.0	42.0	2.0	300.0	299.0	2.0
10866	0.0	0.0	0.0	0.0	0.0	18.0	49.0	1.0	0.0	2.0	...	131.0	48.0	0.0	0.0	0.0	18.0	5.0	38.0	11.0	8.0
10867	0.0	0.0	0.0	0.0	0.0	0.0	1.0	67.0	1.0	85.0	...	0.0	5.0	1.0	1.0	0.0	2.0	0.0	38.0	20.0	0.0

5 rows × 300 columns

In [16]:

```
result_y.head()
```

Out[16]:

```
0    7  
1    1  
2    6  
3    8  
4    9  
Name: Class, dtype: int64
```

## 1.2 Prepare, Merge and Split data

In [17]:

```
X_train, X_test_merge, y_train, y_test_merge = train_test_split(result_x, result_y, stratify=result_y, test_size=0.20)  
X_train_merge, X_cv_merge, y_train_merge, y_cv_merge = train_test_split(X_train, y_train, stratify=y_train, test_size=0.20)
```

In [18]:

```
X_train.shape, y_train.shape
```

Out[18]:

```
((8694, 300), (8694,))
```

In [19]:

```
result_x.shape
```

Out[19]:

```
(10868, 300)
```

## 1.3 Random Forest Classifier on Bigram of ByteFiles

In [40]:

```
%%time  
plt.close()  
# -----  
# default parameters  
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,  
# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min
```

```

# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features= auto , max_leaf_nodes=None, min_
impurity_decrease=0.0,
# min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-fores
t-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
from sklearn.ensemble import RandomForestClassifier
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=r_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c = 10 is 0.08367721126639815
log_loss for c = 50 is 0.06431163796062295
log_loss for c = 100 is 0.06305145455650585
log_loss for c = 500 is 0.062388746329948244
log_loss for c = 1000 is 0.06318184279924256
log_loss for c = 2000 is 0.06283061462565234
log_loss for c = 3000 is 0.06270454305789515

```

```
CPU times: user 13min 43s, sys: 26.1 s, total: 14min 9s
Wall time: 2min 36s
```

In [41]:

```
%%time
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train_merge,y_train_merge)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

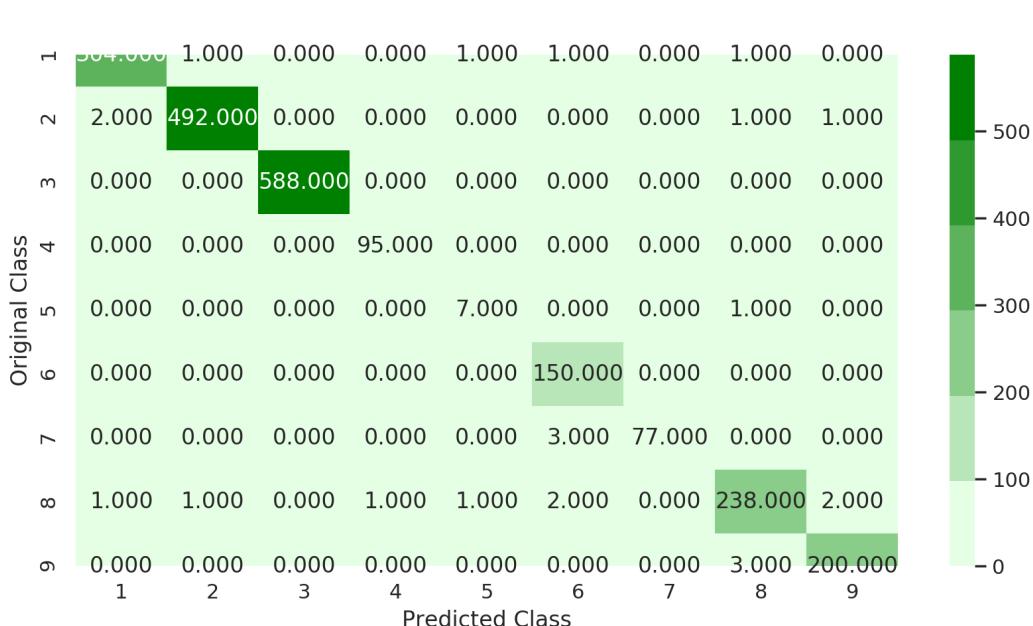
predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_merge, predict_y))
```

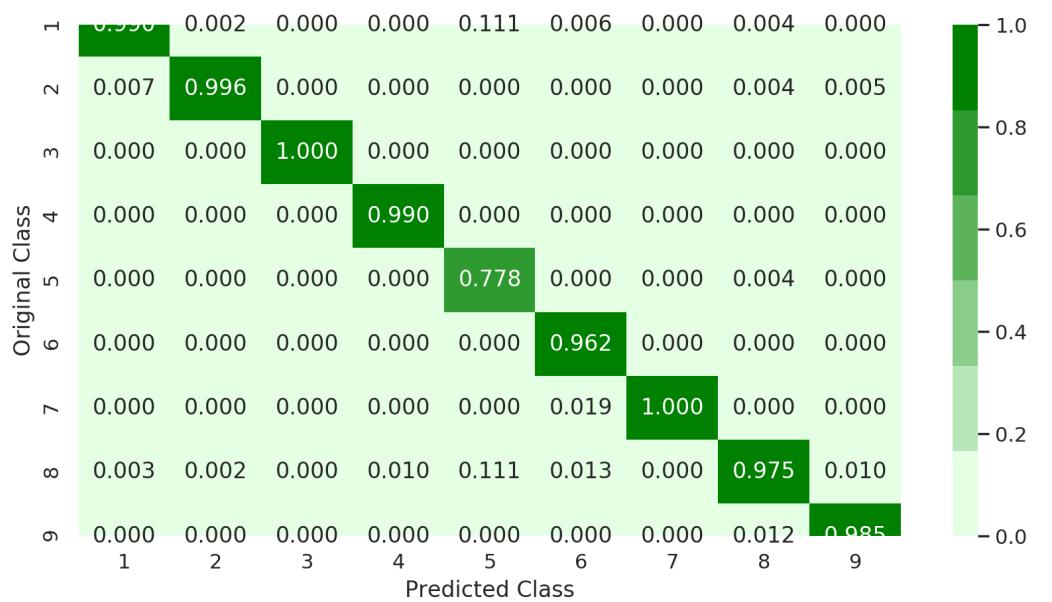
```
For values of best alpha = 500 The train log loss is: 0.017128973414428504
For values of best alpha = 500 The cross validation log loss is: 0.062388746329948244
For values of best alpha = 500 The test log loss is: 0.049864960057818736
CPU times: user 1min 4s, sys: 2.7 s, total: 1min 7s
Wall time: 14 s
```

In [42]:

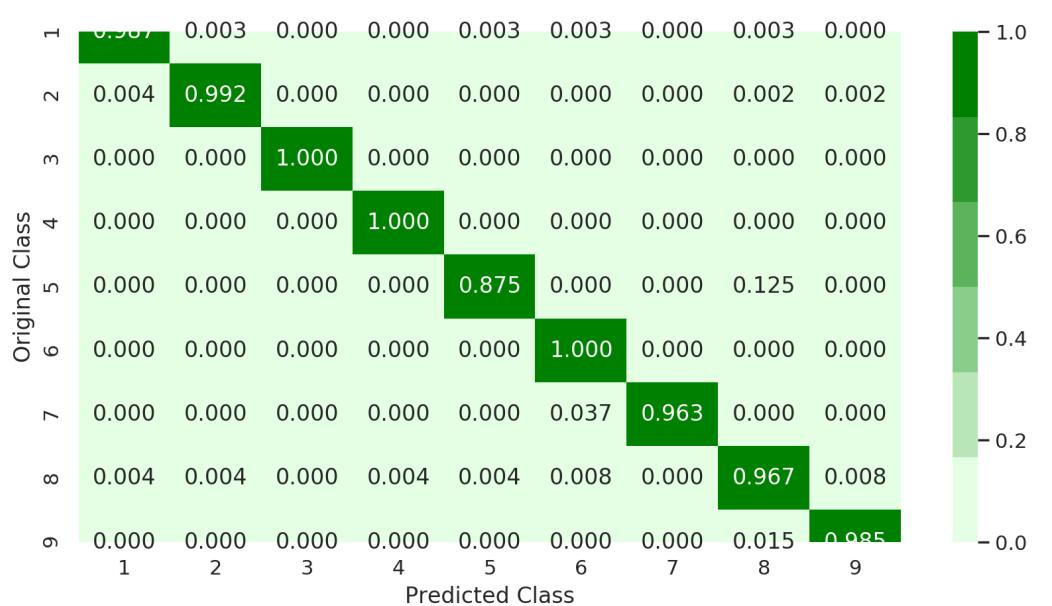
```
plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 1.0579576816927323





Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

## 1.4 Xgboost Classifier on Bigram of ByteFiles

In [20]:

```
%%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
```

```

# ----- #
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# ----- #

alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClassifier(n_estimators=i)
    x_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=x_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

```

```

log_loss for c = 10 is 0.1340123554208798
log_loss for c = 50 is 0.06802316557133047
log_loss for c = 100 is 0.04884714304250573
log_loss for c = 500 is 0.04210767064887841
log_loss for c = 1000 is 0.04168808734618218
log_loss for c = 2000 is 0.04095815860110106
log_loss for c = 3000 is 0.040961270231707134

```

```
CPU times: user 1h 10min 39s, sys: 6.68 s, total: 1h 10min 46s
Wall time: 1h 10min 45s
```

In [21]:

```
%%time
# plt.close()
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_merge, predict_y))

For values of best alpha =  2000 The train log loss is: 0.015158548235132147
For values of best alpha =  2000 The cross validation log loss is: 0.04095815860110106
For values of best alpha =  2000 The test log loss is: 0.05835637248527767
CPU times: user 23min 17s, sys: 5.85 s, total: 23min 23s
Wall time: 23min 23s
```

In [22]:

```
plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 1.0579576816927323

----- Confusion matrix -----  
-----



----- Precision matrix -----

-----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----

-----



```
Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

## 1.5 Xgboost Classifier (Best Hyper Parameter) on Bigram of ByteFiles

In [23]:

```
# https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/
x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
    'subsample':[0.1,0.3,0.5,1]
}
random_cfl=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1,)
random_cfl.fit(X_train_merge, y_train_merge)
```

Fitting 3 folds for each of 10 candidates, totalling 30 fits

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:  3.3min
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:  6.2min
[Parallel(n_jobs=-1)]: Done  19 out of  30 | elapsed:  7.8min remaining:  4.5min
[Parallel(n_jobs=-1)]: Done  23 out of  30 | elapsed:  9.3min remaining:  2.8min
[Parallel(n_jobs=-1)]: Done  27 out of  30 | elapsed: 11.0min remaining:  1.2min
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed: 15.6min finished
```

Out[23]:

```
RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                    estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                           colsample_bylevel=1,
                                           colsample_bynode=1,
                                           colsample_bytree=1, gamma=0,
                                           learning_rate=0.1, max_delta_step=0,
                                           max_depth=3, min_child_weight=1,
                                           missing=None, n_estimators=100,
                                           n_jobs=1, nthread=None,
                                           objective='binary:logistic',
                                           random_state=0, reg_alpha=0.5,
                                           seed=None, silent=None, subsample=1,
                                           verbosity=1),
                    iid='warn', n_iter=10, n_jobs=-1,
                    param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                    pre_dispatch='2*n_jobs', random_state=None, refit=True,
                    return_train_score=False, scoring=None, verbose=10)
```

In [25]:

```
print(random_cfl.best_params_)
```

```
{'subsample': 0.3, 'n_estimators': 2000, 'max_depth': 3, 'learning_rate': 0.03,
 'colsample_bytree': 0.3}
```

In [26]:

```
%%time
# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----
x_cfl=XGBClassifier(n_estimators=2000,max_depth=3,learning_rate=0.03,colsample_bytree=0.3,subsample=0.3,nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)
```

CPU times: user 12min 5s, sys: 1.44 s, total: 12min 7s  
Wall time: 12min 7s

Out[26]:

```
CalibratedClassifierCV(base_estimator=XGBClassifier(base_score=0.5,
booster='gbtree',
colsample_bylevel=1,
colsample_bynode=1,
colsample_bytree=0.3,
gamma=0, learning_rate=0.03,
max_delta_step=0,
max_depth=3,
min_child_weight=1,
missing=None,
n_estimators=2000, n_jobs=1,
nthread=-1,
objective='multi:softmax',
random_state=0, reg_alpha=0,
reg_lambda=1,
scale_pos_weight=1,
seed=None, silent=None,
subsample=0.3,
verbosity=1),
cv='warn', method='sigmoid')
```

In [27]:

```
predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))
```

For values of best alpha = 2000 The train log loss is: 0.01423811620716136  
For values of best alpha = 2000 The cross validation log loss is: 0.04110612337750246  
For values of best alpha = 2000 The test log loss is: 0.05387802812381134

In [28]:

```
plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 0.8279668813247469

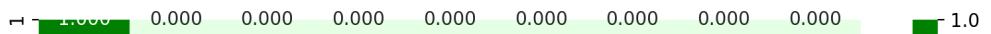


#### Precision matrix -



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]

#### Recall matrix





```
Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

In [31]:

```
from prettytable import PrettyTable

table = PrettyTable()
table.field_names = ['Model', 'Best Hyper Parameter', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']

table.add_row(['Random Forest Classifier', 500, 0.017128973414428504, 0.062388746329948244, 0.049864960057818736, 1.0579576816927323])
table.add_row(['XgBoost Classifier', 2000, 0.015158548235132147, 0.04095815860110106, 0.05835637248527767, 1.0579576816927323])
table.add_row(['XgBoost Classifier with Best Hyper Parameter', 2000, 0.01423811620716136, 0.04110612337750246, 0.05387802812381134, 0.8279668813247469])

print(table)
```

Model		Best Hyper Parameter	Train Log Loss	CV Log Loss	Test Log Loss
Log Loss	Test Log Loss	Number of Misclassified Points			
Random Forest Classifier	500	0.017128973414428504	0.062388746329948244	0.049864960057818736	1.0579576816927323
XgBoost Classifier	2000	0.015158548235132147	0.04095815860110106	0.05835637248527767	1.0579576816927323
XgBoost Classifier with Best Hyper Parameter	2000	0.01423811620716136	0.04110612337750246	0.05387802812381134	0.8279668813247469

## Assignment Part 2

- 2 Use features from link and reduce log loss <https://github.com/dchad/malware-detection>

### 2.1 Image Based Features Extraction from asm files

In [3]:

```
from multiprocessing import Pool
import os
from csv import writer
import numpy as np
import math
```

```
import scipy.misc
import array
import time as tm

import numpy as np
import scipy as sp
import pandas as pd
import sklearn as skl
import matplotlib.pyplot as plt
from sklearn.feature_selection import SelectKBest, SelectPercentile
from sklearn.feature_selection import chi2
from sklearn.metrics import log_loss, confusion_matrix, accuracy_score
from sklearn.ensemble import RandomForestClassifier, ExtraTreesClassifier
from sklearn.model_selection import cross_val_score, KFold
```

In [28]:

```
def entropy(p,n):
    """
    Calculate entropy of the file
    """
    p_ratio = float(p)/(p+n)
    n_ratio = float(n)/(p+n)
    return -p_ratio*math.log(p_ratio) - n_ratio * math.log(n_ratio)
```

In [29]:

```
def info_gain(p0,n0,p1,n1,p,n):
    """
    Calculate information gain
    """
    return entropy(p,n) - float(p0+n0)/(p+n)*entropy(p0,n0) - float(p1+n1)/(p+n)*entropy(p1,n1)
```

In [30]:

```
def read_image(filename):
    """
    Read image data
    """
    f = open(filename, 'rb')
    ln = os.path.getsize(filename) # length of file in bytes
    width = 256
    rem = ln%width
    a = array.array("B") # uint8 array
    a.fromfile(f,ln-rem)
    f.close()
    g = np.reshape(a,(int(len(a)/width), width))
    g = np.uint8(g)
    g = np.resize(g, (1000,))
    return list(g)
```

In [31]:

```
def extract_asm_image_features(tfiles):
    """
    Extract image features from the asm files
    """
    asm_files = [i for i in tfiles if '.asm' in i]
    ftot = len(asm_files)

    # Generate feature file csv
    pid = os.getpid()
    feature_file = str(pid) + '-image-features-asm.csv'

    outrows = []
    with open(feature_file,'w') as f:
        fw = writer(f)
        column_names = ['filename'] + [("ASM_{:s}".format(str(x))) for x in range(1000)]
        fw.writerow(column_names)
        for idx, fname in enumerate(asm_files):
            file_id = fname.split('.')[0]
            image_data = read_image('asmFiles/' + fname)
            outrows.append([file_id] + image_data)
```

```

# Print progress
if (idx+1) % 100 == 0:
    print(pid, idx + 1, 'of', ftot, 'files processed.')
    fw.writerows(outrows)
    outrows = []

# Write remaining files
if len(outrows) > 0:
    fw.writerows(outrows)
    outrows = []

```

In [14]:

```

# Now divide the train files into four groups for multiprocessing
start_time = tm.time()
tfiles = os.listdir('asmFiles')
quart = int(len(tfiles)/4)
# print(quart)
train1 = tfiles[:quart]
train2 = tfiles[quart:(2*quart)]
train3 = tfiles[(2*quart):(3*quart)]
train4 = tfiles[(3*quart):]
print(len(tfiles), quart, (len(train1)+len(train2)+len(train3)+len(train4)))
trains = [train1, train2, train3, train4]
p = Pool(4)
p.map(extract_asm_image_features, trains)
print("Elapsed time: {:.2f} hours.".format((tm.time() - start_time)/3600.0))

```

```

10868 2717 10868
2320 100 of 2717 files processed.
2319 100 of 2717 files processed.
2317 100 of 2717 files processed.
2318 100 of 2717 files processed.
2320 200 of 2717 files processed.
2319 200 of 2717 files processed.
2317 200 of 2717 files processed.
2318 200 of 2717 files processed.
2319 300 of 2717 files processed.
2320 300 of 2717 files processed.
2317 300 of 2717 files processed.
2318 300 of 2717 files processed.
2319 400 of 2717 files processed.
2320 400 of 2717 files processed.
2317 400 of 2717 files processed.
2318 400 of 2717 files processed.
2319 500 of 2717 files processed.
2320 500 of 2717 files processed.
2317 500 of 2717 files processed.
2319 600 of 2717 files processed.
2318 500 of 2717 files processed.
2320 600 of 2717 files processed.
2318 600 of 2717 files processed.
2317 600 of 2717 files processed.
2319 700 of 2717 files processed.
2320 700 of 2717 files processed.
2319 800 of 2717 files processed.
2317 700 of 2717 files processed.
2318 700 of 2717 files processed.
2320 800 of 2717 files processed.
2319 900 of 2717 files processed.
2317 800 of 2717 files processed.
2318 800 of 2717 files processed.
2320 900 of 2717 files processed.
2317 900 of 2717 files processed.
2318 900 of 2717 files processed.
2319 1000 of 2717 files processed.
2320 1000 of 2717 files processed.
2318 1000 of 2717 files processed.
2319 1100 of 2717 files processed.
2317 1000 of 2717 files processed.
2318 1100 of 2717 files processed.
2320 1100 of 2717 files processed.
2319 1200 of 2717 files processed.
2317 1100 of 2717 files processed.

```

```
2318 1200 of 2717 files processed.  
2320 1200 of 2717 files processed.  
2319 1300 of 2717 files processed.  
2317 1200 of 2717 files processed.  
2318 1300 of 2717 files processed.  
2320 1300 of 2717 files processed.  
2319 1400 of 2717 files processed.  
2318 1400 of 2717 files processed.  
2320 1400 of 2717 files processed.  
2317 1300 of 2717 files processed.  
2319 1500 of 2717 files processed.  
2320 1500 of 2717 files processed.  
2318 1500 of 2717 files processed.  
2317 1400 of 2717 files processed.  
2319 1600 of 2717 files processed.  
2320 1600 of 2717 files processed.  
2318 1600 of 2717 files processed.  
2319 1700 of 2717 files processed.  
2317 1500 of 2717 files processed.  
2320 1700 of 2717 files processed.  
2319 1800 of 2717 files processed.  
2318 1700 of 2717 files processed.  
2317 1600 of 2717 files processed.  
2319 1900 of 2717 files processed.  
2320 1800 of 2717 files processed.  
2318 1800 of 2717 files processed.  
2317 1700 of 2717 files processed.  
2319 2000 of 2717 files processed.  
2320 1900 of 2717 files processed.  
2317 1800 of 2717 files processed.  
2318 1900 of 2717 files processed.  
2319 2100 of 2717 files processed.  
2320 2000 of 2717 files processed.  
2318 2000 of 2717 files processed.  
2317 1900 of 2717 files processed.  
2319 2200 of 2717 files processed.  
2320 2100 of 2717 files processed.  
2317 2000 of 2717 files processed.  
2318 2100 of 2717 files processed.  
2319 2300 of 2717 files processed.  
2320 2200 of 2717 files processed.  
2317 2100 of 2717 files processed.  
2318 2200 of 2717 files processed.  
2319 2400 of 2717 files processed.  
2320 2300 of 2717 files processed.  
2317 2200 of 2717 files processed.  
2318 2300 of 2717 files processed.  
2319 2500 of 2717 files processed.  
2320 2400 of 2717 files processed.  
2317 2300 of 2717 files processed.  
2318 2400 of 2717 files processed.  
2320 2500 of 2717 files processed.  
2319 2600 of 2717 files processed.  
2318 2500 of 2717 files processed.  
2317 2400 of 2717 files processed.  
2319 2700 of 2717 files processed.  
2320 2600 of 2717 files processed.  
2318 2600 of 2717 files processed.  
2317 2500 of 2717 files processed.  
2320 2700 of 2717 files processed.  
2318 2700 of 2717 files processed.  
2317 2600 of 2717 files processed.  
2317 2700 of 2717 files processed.  
Elapsed time: 0.39 hours.
```

```
Process ForkPoolWorker-13:  
Traceback (most recent call last):  
  File "/home/mayankgupta/anaconda3/lib/python3.7/multiprocessing/process.py", line 297, in  
_bootstrap  
    self._run()  
  File "/home/mayankgupta/anaconda3/lib/python3.7/multiprocessing/process.py", line 99, in run  
    self._target(*self._args, **self._kwargs)  
  File "/home/mayankgupta/anaconda3/lib/python3.7/multiprocessing/pool.py", line 110, in worker  
    task = get()  
  File "/home/mayankgupta/anaconda3/lib/python3.7/multiprocessing/queues.py", line 351, in get  
    with self._rlock:  
  File "/home/mayankgupta/anaconda3/lib/python3.7/multiprocessing/synchronize.py", line 95, in _
```

```
file "/home/mayankgupta/anaconda3/lib/python3.7/multiprocessing/synchronize.py", line 95, in __enter__
    return self._semlock.__enter__()
KeyboardInterrupt
```

In [32]:

```
#merging all generated csv files

labels = pd.read_csv('trainLabels.csv')
d1 = pd.read_csv('2317-image-features-asm.csv')
d2 = pd.read_csv('2318-image-features-asm.csv')
d3 = pd.read_csv('2319-image-features-asm.csv')
d4 = pd.read_csv('2320-image-features-asm.csv')
d4.shape
```

Out[32]:

```
(2717, 1001)
```

In [33]:

```
data = pd.concat([d1, d2, d3, d4])
data.shape
```

Out[33]:

```
(10868, 1001)
```

In [34]:

```
data.reset_index(drop=True, inplace=True)
```

In [35]:

```
labels.head()
```

Out[35]:

	Id	Class
0	01kcPWA9K2B0xQeS5Rju	1
1	04EjldbPV5e1XroFOpiN	1
2	05EeG39MTRrl6VY21DPd	1
3	05rJTUWYAKNegBk2wE8X	1
4	0AnoOZDNbPXlr2MRBSCJ	1

In [36]:

```
sorted_train_data = data.sort_values(by='filename', axis=0, ascending=True, inplace=False)
sorted_train_labels = labels.sort_values(by='Id', axis=0, ascending=True, inplace=False)
X = sorted_train_data.iloc[:,1:]
y = np.array(sorted_train_labels.iloc[:,1:])
```

In [37]:

```
X.shape, y.shape
```

Out[37]:

```
((10868, 1000), (10868,))
```

## 2.1.1 Selecting top 50% variance features

In [38]:

```
# find the top 50 percent variance features, from 1000 -> 500 features
fsp = SelectPercentile(chi2, 50)
X_new_50 = fsp.fit_transform(X,y)
X_new_50.shape
```

Out[38]:

(10868, 500)

In [39]:

```
selected_names = fsp.get_support(indices=True)
selected_names = selected_names + 1
selected_names
```

Out[39]:

```
array([ 2,  4,  5, 15, 21, 22, 24, 25, 26, 27, 29, 30, 32,
       33, 34, 35, 41, 42, 43, 44, 48, 50, 125, 126, 135, 136,
      138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 151, 152,
      154, 155, 156, 157, 158, 160, 161, 162, 163, 164, 165, 167, 169,
      173, 174, 179, 186, 188, 190, 198, 201, 202, 205, 215, 216, 217,
      219, 220, 221, 222, 223, 224, 226, 227, 229, 236, 240, 241, 242,
      243, 244, 245, 246, 247, 248, 249, 252, 253, 260, 261, 262, 263,
      264, 265, 266, 267, 268, 269, 271, 272, 273, 282, 287, 291, 292,
      293, 294, 295, 296, 297, 307, 308, 310, 311, 312, 313, 314, 315,
      316, 317, 318, 319, 321, 323, 326, 327, 328, 330, 334, 337, 338,
      339, 340, 341, 343, 344, 345, 346, 349, 350, 351, 352, 353, 354,
      356, 357, 358, 359, 366, 367, 368, 370, 371, 372, 373, 374, 375,
      376, 378, 379, 380, 381, 384, 385, 386, 387, 388, 390, 391, 392,
      399, 400, 401, 402, 403, 404, 405, 408, 409, 410, 412, 413, 414,
      415, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431,
      436, 437, 439, 440, 441, 442, 443, 445, 446, 447, 448, 449, 450,
      451, 452, 453, 457, 458, 459, 460, 461, 464, 465, 466, 467, 477,
      478, 479, 480, 481, 482, 538, 539, 555, 556, 557, 558, 559, 560,
      561, 563, 564, 567, 568, 571, 572, 573, 580, 581, 582, 583, 584,
      585, 586, 587, 588, 589, 590, 597, 598, 600, 601, 602, 603, 606,
      607, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624,
      627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 640, 641,
      642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654,
      655, 656, 657, 658, 659, 662, 664, 670, 671, 672, 673, 674, 675,
      676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688,
      689, 691, 692, 693, 694, 695, 696, 701, 702, 703, 704, 708, 709,
      711, 712, 713, 714, 715, 717, 718, 719, 720, 721, 722, 723, 724,
      725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 738,
      739, 740, 743, 744, 751, 752, 753, 754, 755, 756, 757, 758, 759,
      760, 761, 762, 763, 765, 774, 775, 776, 777, 778, 779, 780, 781,
      782, 784, 785, 786, 787, 788, 789, 793, 798, 801, 802, 813, 814,
      818, 819, 820, 830, 831, 835, 836, 837, 838, 840, 841, 847, 848,
      849, 850, 851, 852, 853, 855, 856, 857, 866, 867, 868, 869, 870,
      873, 874, 875, 876, 877, 878, 879, 882, 898, 899, 904, 907, 908,
      919, 920, 923, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939,
      940, 941, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957,
      958, 959, 960, 961, 962, 963, 965, 966, 967, 968, 973, 974, 975,
      976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 989, 990, 991,
      992, 995, 996, 997, 998, 999])
```

In [40]:

```
data_trimmed = sorted_train_data.iloc[:,selected_names]
data_fnames = pd.DataFrame(sorted_train_data['filename'])
data_reduced = data_fnames.join(data_trimmed)
data_reduced.head()
```

Out[40]:

	filename	ASM_1	ASM_3	ASM_4	ASM_14	ASM_20	ASM_21	ASM_23	ASM_24	ASM_25	...	ASM_984	ASM
7551	01lsoiSMh5gxyDYTE4CB	116	120	116	9	32	32	32	32	32	...	54	
6521	01SuzwMJEIXsK7A8dQbl	69	68	69	48	9	9	13	10	72	...	32	
10798	01azqd4lnC7m9JpocGv5	69	68	69	48	9	9	13	10	72	...	32	
640	01iLmYSAUuGpDpH	69	68	69	48	9	9	13	10	72	...	32	

```

b10      01jsnpXSAlgwbareuxru
filename  ASM_1   ASM_3   ASM_4   ASM_14  ASM_20  ASM_21  ASM_23  ASM_24  ASM_25 ... ASM_984  ASM
---6715  01kcPWA9K2B0xQeS5Rju  69    68    69    48    9     9     13    10    72    ... 83

```

5 rows × 501 columns

In [20]:

```
data_reduced.to_csv('sorted-features-asm-50percent.csv', index=False)
```

In [41]:

```
data_reduced = pd.read_csv('sorted-features-asm-50percent.csv')
data_reduced.shape
```

Out[41]:

```
(10868, 501)
```

In [42]:

```
data_reduced.rename(columns={'filename': 'ID'}, inplace=True)
```

In [43]:

```
data_reduced.shape
```

Out[43]:

```
(10868, 501)
```

In [44]:

```
data_reduced.head()
```

Out[44]:

	ID	ASM_1	ASM_3	ASM_4	ASM_14	ASM_20	ASM_21	ASM_23	ASM_24	ASM_25	...	ASM_984	ASM_985
0	01lsoiSMh5gxyDYTI4CB	116	120	116	9	32	32	32	32	32	...	54	4
1	01SuzwMJEIXsK7A8dQbI	69	68	69	48	9	9	13	10	72	...	32	10
2	01azqd4InC7m9JpocGv5	69	68	69	48	9	9	13	10	72	...	32	10
3	01jsnpXSAlgwbareuxru	69	68	69	48	9	9	13	10	72	...	32	10
4	01kcPWA9K2B0xQeS5Rju	69	68	69	48	9	9	13	10	72	...	83	8

5 rows × 501 columns

## 2.2 Implement asm image features + bytes uni-gram features

### 2.2.1 Merge asm image features + bytes uni-gram features

In [46]:

```
result_x = pd.merge(result.drop('size', axis=1), data_reduced, on='ID', how='left')
result_y = result_x['Class']
# result_x = result_x.drop(['ID', 'rtn', '.BSS:', '.CODE', 'Class'], axis=1)
result_x = result_x.drop(['ID', 'Class'], axis=1)
result_x.head()
```

Out[46]:

0	1	2	3	4	5	6	7	8	9	...	ASM_984	ASM_988	ASM_989
0	0.002525	0.000082	0.000013	0.000017	0.000016	0.000012	0.000004	0.000003	0.000099	0.000004	...	32	32

1	0.010740	0.001771	0.000418	0.000489	0.000886	0.000328	0.000338	0.000662	0.000996	0.001819	...	ASM_984	ASM_988	ASM_992
2	0.005374	0.000624	0.000130	0.000249	0.000129	0.000100	0.000105	0.000168	0.000174	0.000170	...	32	70	1
3	0.008818	0.000957	0.000176	0.000247	0.000174	0.000207	0.000123	0.000221	0.000222	0.000236	...	32	10	1
4	0.037465	0.000991	0.000251	0.000246	0.000315	0.000366	0.000252	0.000369	0.000447	0.000460	...	32	10	1

5 rows × 758 columns

In [47]:

```
X_train, X_test_merge, y_train, y_test_merge = train_test_split(result_x, result_y,stratify=result_y,test_size=0.20)
X_train_merge, X_cv_merge, y_train_merge, y_cv_merge = train_test_split(X_train, y_train,stratify=y_train,test_size=0.20)
```

## 2.2.2 Random Forest Classifier asm image features + bytes uni-gram features

In [48]:

```
%%time
plt.close()
# -----
# default parameters
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,
# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
# min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
# verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forest-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
from sklearn.ensemble import RandomForestClassifier
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=r_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

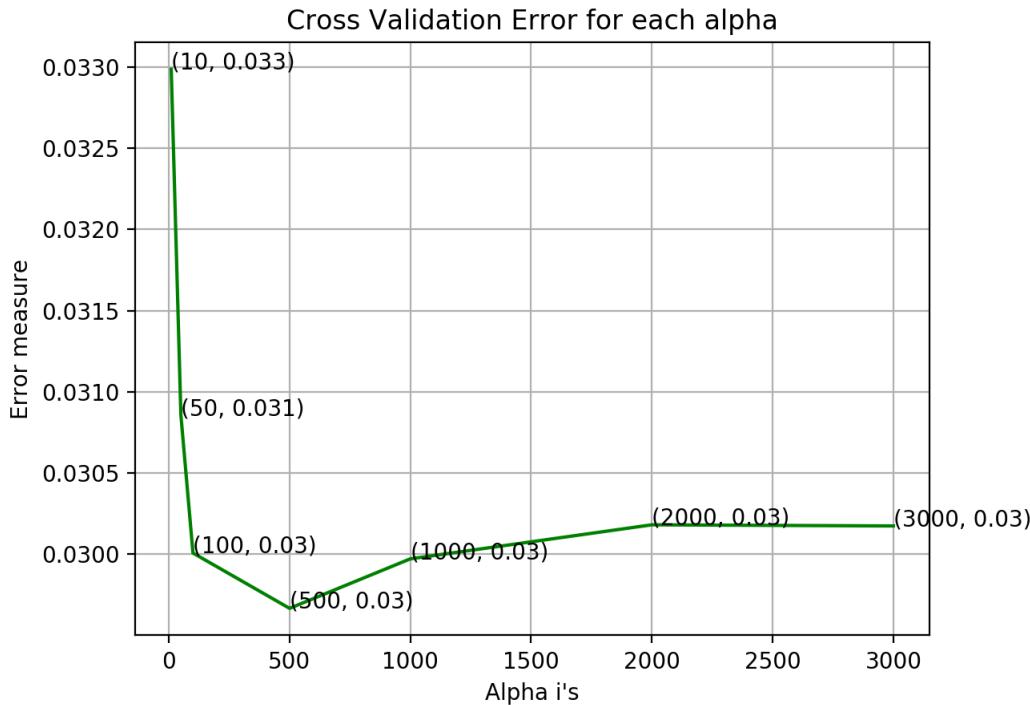
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```

```

log_loss for c = 10 is 0.03298443094752871
log_loss for c = 50 is 0.03085568636919456
log_loss for c = 100 is 0.030007763918641725
log_loss for c = 500 is 0.02966720993128468
log_loss for c = 1000 is 0.029971997588127103
log_loss for c = 2000 is 0.030181377953265455
log_loss for c = 3000 is 0.0301742816771141

```



```

CPU times: user 18min 20s, sys: 32 s, total: 18min 52s
Wall time: 3min 11s

```

## 2.2.3 Random Forest Classifier (Best Hyper Parameters) asm image features + bytes uni-gram features

In [49]:

```

%%time
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train_merge,y_train_merge)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))

```

```

For values of best alpha = 500 The train log loss is: 0.012780736170043525
For values of best alpha = 500 The cross validation log loss is: 0.02966720993128468
For values of best alpha = 500 The test log loss is: 0.034478799054374044
CPU times: user 1min 27s, sys: 4.1 s, total: 1min 31s
Wall time: 16.7 s

```

In [50]:

```
plt.close()
```

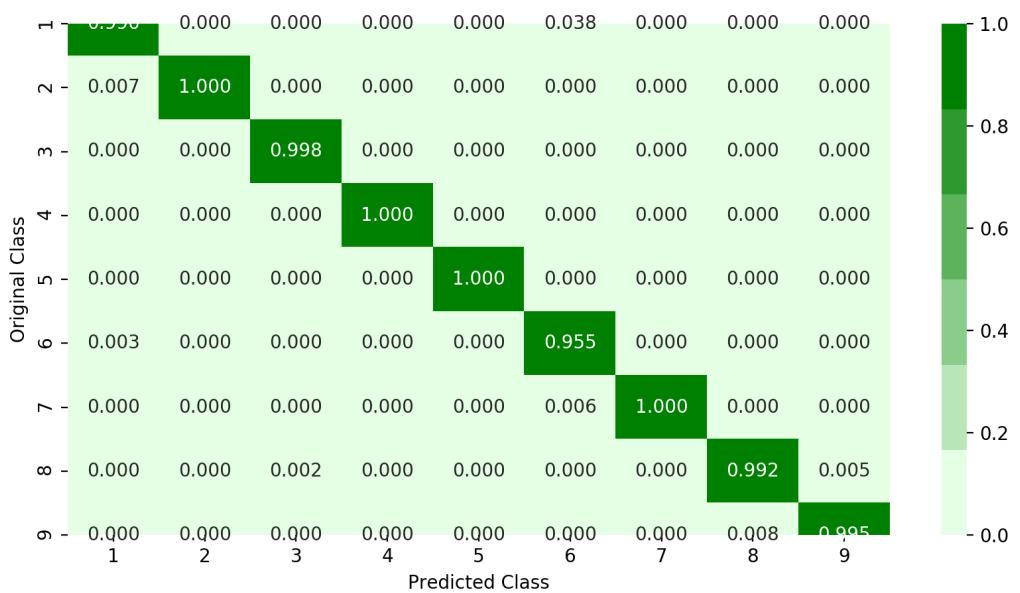
```
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 0.6439742410303588

### Confusion matrix -----

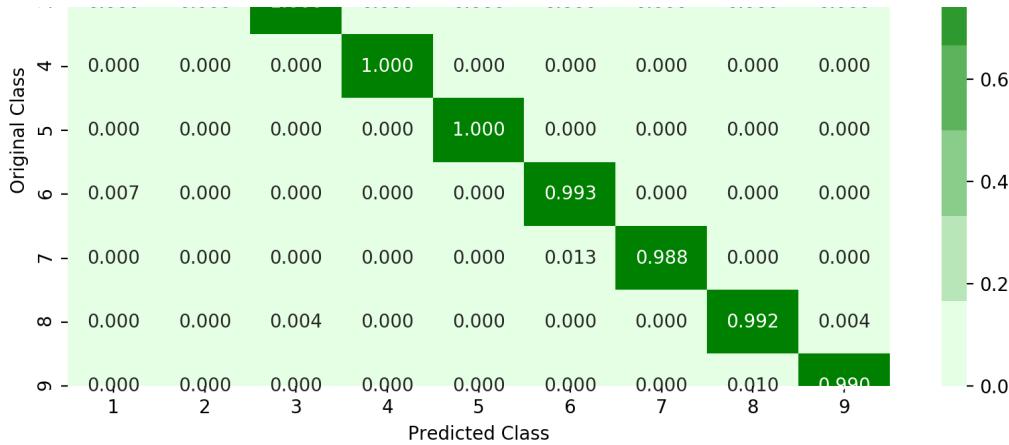


### Precision matrix ----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

## Recall matrix



```
Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

## 2.2.4 Xgboost Classifier asm image features + bytes uni-gram features

In [51]:

```
%%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClassifier(n_estimators=i)
    x_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=x_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i], 'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
#-----
```

```

plt.ylabel("error measure")
plt.show()

log_loss for c = 10 is 0.0792497193661065
log_loss for c = 50 is 0.031980906132399546
log_loss for c = 100 is 0.025713754198451427
log_loss for c = 500 is 0.024323002938319807
log_loss for c = 1000 is 0.02476490339663419
log_loss for c = 2000 is 0.02516493136273984
log_loss for c = 3000 is 0.025065229722220864

```

CPU times: user 3h 13min 26s, sys: 613 ms, total: 3h 13min 26s  
Wall time: 3h 13min 25s

In [52]:

```

%%time
plt.close()
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_merge, predict_y))

For values of best alpha = 500 The train log loss is: 0.01098127054981705
For values of best alpha = 500 The cross validation log loss is: 0.025065229722220864
For values of best alpha = 500 The test log loss is: 0.033166108866613246
CPU times: user 1h 18min 17s, sys: 0 ns, total: 1h 18min 17s
Wall time: 1h 18min 17s

```

## 2.2.5 Xgboost Classifier (Best Hyper Parameters) asm image features + bytes uni-gram features

In [53]:

```
x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
    'subsample':[0.1,0.3,0.5,1]
}
random_cfl=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1,)
random_cfl.fit(X_train_merge, y_train_merge)
```

Fitting 3 folds for each of 10 candidates, totalling 30 fits

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:  1.9min
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:  3.8min
[Parallel(n_jobs=-1)]: Done  19 out of  30 | elapsed:  6.5min remaining:  3.7min
[Parallel(n_jobs=-1)]: Done  23 out of  30 | elapsed: 11.3min remaining:  3.4min
[Parallel(n_jobs=-1)]: Done  27 out of  30 | elapsed: 14.0min remaining:  1.6min
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed: 16.0min finished
```

Out[53]:

```
RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                    estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                            colsample_bylevel=1,
                                            colsample_bynode=1,
                                            colsample_bytree=1, gamma=0,
                                            learning_rate=0.1, max_delta_step=0,
                                            max_depth=3, min_child_weight=1,
                                            missing=None, n_estimators=100,
                                            n_jobs=1, nthread=None,
                                            objective='binary:logistic',
                                            random_state=0, reg_alpha=0,
                                            seed=None, silent=None, subsample=1,
                                            verbosity=1),
                    iid='warn', n_iter=10, n_jobs=-1,
                    param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                    pre_dispatch='2*n_jobs', random_state=None, refit=True,
                    return_train_score=False, scoring=None, verbose=10)
```

In [54]:

```
print (random_cfl.best_params_)

{'subsample': 0.3, 'n_estimators': 1000, 'max_depth': 3, 'learning_rate': 0.15,
'colsample_bytree': 0.3}
```

In [57]:

```
%%time
# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# # max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# # scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit() v. sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None,
```

```

# fit(x, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----
```

```

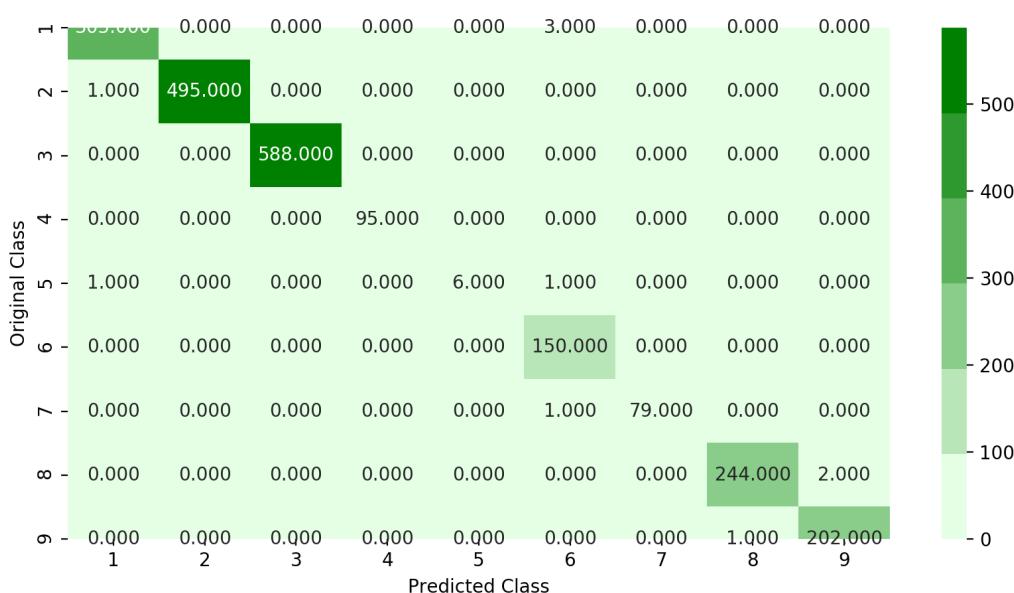
x_cfl=XGBClassifier(n_estimators=1000,max_depth=3,learning_rate=0.15,colsample_bytree=0.3,subsample=0.3,nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))
```

In [56]:

```
plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 0.45998160073597055





Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

## 2.2.6 Conclusion and Model Comparision (asm image features + bytes uni-gram features)

In [79]:

```
table = PrettyTable()
table.field_names = ['Model', 'Best HyperParameter', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']

table.add_row(['Random Forest Classifier', 500, 0.012780736170043525, 0.02966720993128468, 0.034478799054374044, 0.6439742410303588])
table.add_row(['Xgboost Classifier', 500, 0.011284904939918825, 0.024527899714724164, 0.028035316596927817, 0.45998160073597055])

print(table)
```

+-----+-----+-----+-----+-----+

Model	Best HyperParameter	Train Log Loss	CV Log Loss	
Test Log Loss	Number of Misclassified Points			
Random Forest Classifier   0.034478799054374044	500   0.6439742410303588	0.012780736170043525	0.02966720993128468	0.
Xgboost Classifier   0.028035316596927817	500   0.45998160073597055	0.011284904939918825	0.024527899714724164	0.

## 2.3 Implement asm unigram + asm extracted image features

### 2.3.1 Merge asm unigram + asm extracted image features

In [80]:

```
print(data_reduced.shape)
print(result_asm.shape)
```

(10868, 501)  
(10868, 54)

In [81]:

```
result_asm.columns
```

Out[81]:

```
Index(['ID', 'HEADER:', '.text:', '.Pav:', '.idata:', '.data:', '.bss:',
      '.rdata:', '.edata:', '.rsrc:', '.tls:', '.reloc:', '.BSS:', '.CODE',
      'jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc',
      'dec', 'add', 'imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror',
      'rol', 'jnb', 'jz', 'rtn', 'lea', 'movzx', '.dll', 'std::', ':dword',
      'edx', 'esi', 'eax', 'ebx', 'ecx', 'edi', 'ebp', 'esp', 'eip', 'Class',
      'size'],
      dtype='object')
```

In [82]:

```
data_reduced.columns
```

Out[82]:

```
Index(['ID', 'ASM_1', 'ASM_3', 'ASM_4', 'ASM_14', 'ASM_20', 'ASM_21', 'ASM_23',
      'ASM_24', 'ASM_25',
      ...,
      'ASM_984', 'ASM_988', 'ASM_989', 'ASM_990', 'ASM_991', 'ASM_994',
      'ASM_995', 'ASM_996', 'ASM_997', 'ASM_998'],
      dtype='object', length=501)
```

In [83]:

```
result_x = pd.merge(result_asm, data_reduced, on='ID', how='left')
result_y = result_x['Class']
result_x = result_x.drop(['ID', 'rtn', '.BSS:', '.CODE', 'Class'], axis=1)
# result_x = result_x.drop(['ID', 'Class'], axis=1)
result_x.head()
```

Out[83]:

	HEADER:	.text:	.Pav:	.idata:	.data:	.bss:	.rdata:	.edata:	.rsrc:	.tls:	...	ASM_984	ASM_988	ASM_989	ASM_991
0	0.107345	0.001092	0.0	0.000761	0.000023	0.0	0.000084	0.0	0.000072	0.0	...	83	83	69	7
1	0.096045	0.001230	0.0	0.000617	0.000019	0.0	0.000000	0.0	0.000072	0.0	...	114	32	32	32
2	0.096045	0.000627	0.0	0.000300	0.000017	0.0	0.000038	0.0	0.000072	0.0	...	32	32	32	59

```
3 0.096045 0.000333 .text: .Pav: 0.0 0.000258 0.000008 .idata: .data: .bss: 0.0 0.000000 .rdata: .edata: 0.0 0.000072 .rsrc: .tls: ... :: ASM_984 114 ASM_988 32 ASM_989 32 ASM_991 3
```

5 rows × 549 columns

In [84]:

```
result_x.columns
```

Out[84]:

```
Index(['HEADER:', '.text:', '.Pav:', '.idata:', '.data:', '.bss:', '.rdata:', '.edata:', '.rsrc:', '.tls:', '', 'ASM_984', 'ASM_988', 'ASM_989', 'ASM_990', 'ASM_991', 'ASM_994', 'ASM_995', 'ASM_996', 'ASM_997', 'ASM_998'], dtype='object', length=549)
```

In [85]:

```
print(result_x.shape)
```

(10868, 549)

In [86]:

```
X_train, X_test_merge, y_train, y_test_merge = train_test_split(result_x, result_y, stratify=result_y, test_size=0.20)
X_train_merge, X_cv_merge, y_train_merge, y_cv_merge = train_test_split(X_train, y_train, stratify=y_train, test_size=0.20)
```

## 2.3.2 Random Forest Classifier with asm unigram + asm extracted image features

In [86]:

```
%%time
plt.close()
# -----
# default parameters
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,
# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
# min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
# verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forest-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
from sklearn.ensemble import RandomForestClassifier
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
```

```

cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=r_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

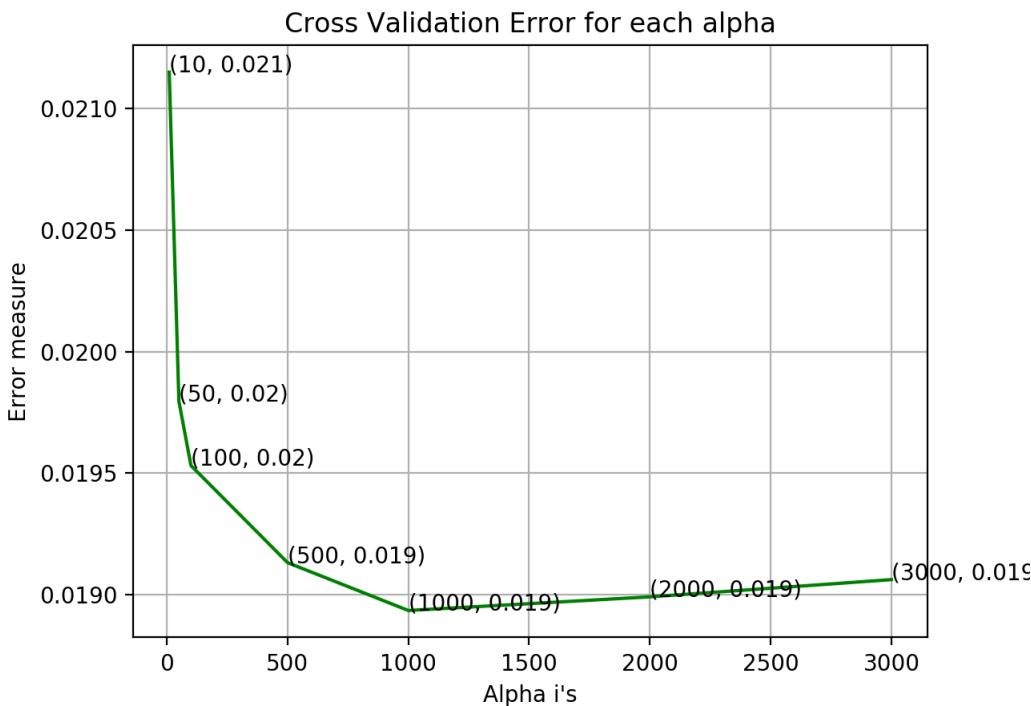
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

```

```

log_loss for c = 10 is 0.021147757013626152
log_loss for c = 50 is 0.019795586264623286
log_loss for c = 100 is 0.019531514435224526
log_loss for c = 500 is 0.019132586421543085
log_loss for c = 1000 is 0.01893597948023988
log_loss for c = 2000 is 0.018992036016380148
log_loss for c = 3000 is 0.01906281256167374

```



```

CPU times: user 7min 37s, sys: 35.9 s, total: 8min 12s
Wall time: 2min 9s

```

In [87]:

```

%%time
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train_merge,y_train_merge)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ',alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ',alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)

```

```

predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:", log_loss(y_test_merge, predict_y))

```

For values of best alpha = 1000 The train log loss is: 0.008896043075915374  
 For values of best alpha = 1000 The cross validation log loss is: 0.01893597948023988  
 For values of best alpha = 1000 The test log loss is: 0.02220239178075276  
 CPU times: user 1min 16s, sys: 6.4 s, total: 1min 22s  
 Wall time: 22.9 s

In [88]:

```

plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))

```

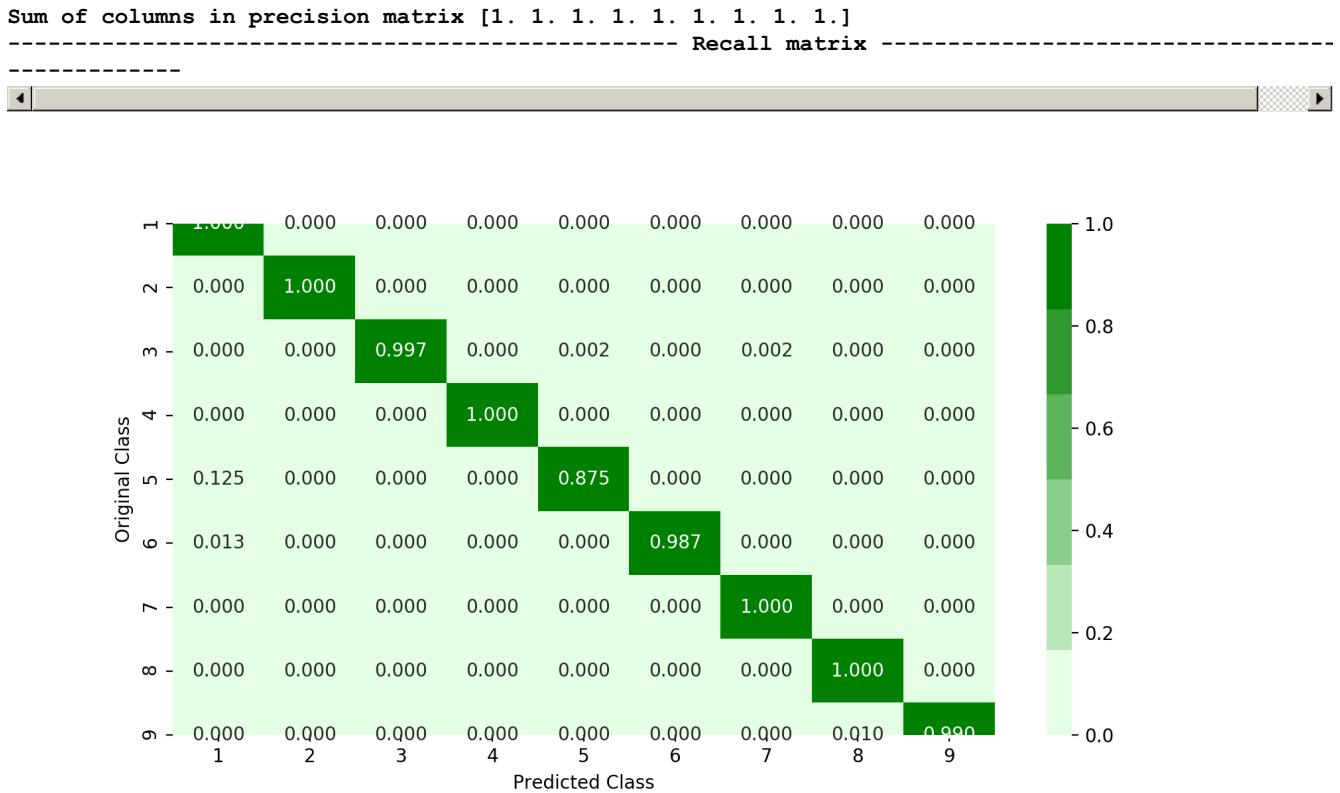
Number of misclassified points 0.3219871205151794

----- Confusion matrix -----



----- Precision matrix -----





Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

### 2.3.3 Xgboost implementation with asm unigram + asm extracted image features

In [87]:

```
%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClassifier(n_estimators=i)
    x_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=x_cfl.classes_, eps=1e-15))
```

```

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

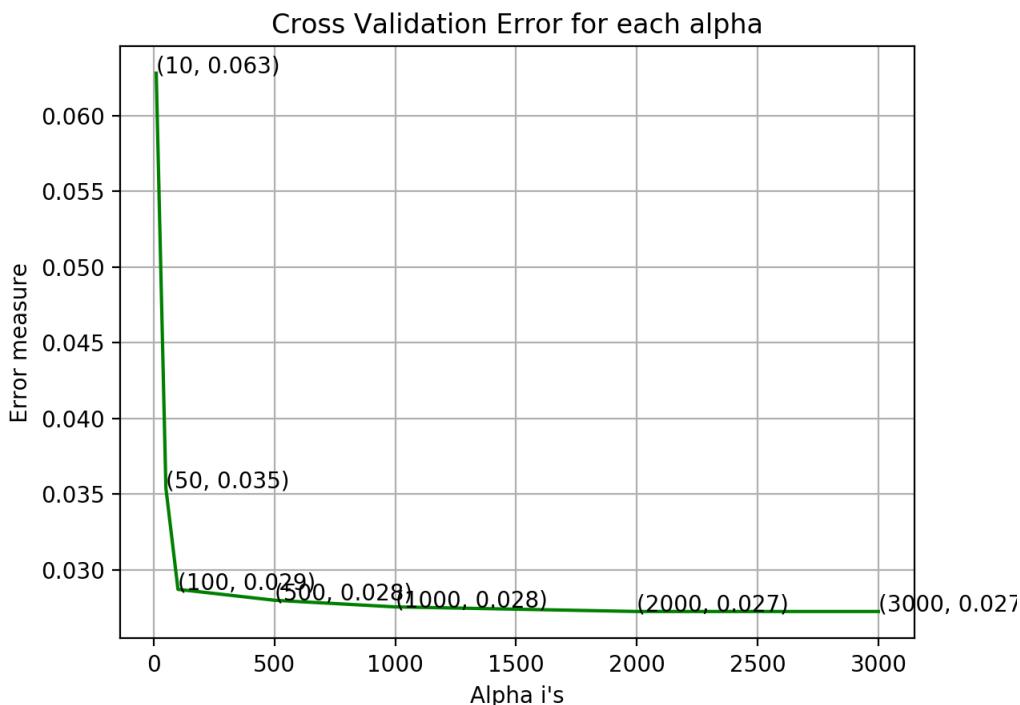
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

```

```

log_loss for c = 10 is 0.06278899176536516
log_loss for c = 50 is 0.035408785513269654
log_loss for c = 100 is 0.02871795575938164
log_loss for c = 500 is 0.027998979315273183
log_loss for c = 1000 is 0.027561393234936435
log_loss for c = 2000 is 0.02725772002665156
log_loss for c = 3000 is 0.02725750007288518

```



```

CPU times: user 1h 48min 55s, sys: 363 ms, total: 1h 48min 55s
Wall time: 1h 48min 55s

```

In [88]:

```

%%time
plt.close()
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha]. "The test log loss"

```

```

is:",log_loss(y_test_merge, predict_y))

For values of best alpha = 3000 The train log loss is: 0.008320609361993012
For values of best alpha = 3000 The cross validation log loss is: 0.02725750007288518
For values of best alpha = 3000 The test log loss is: 0.02159222942854402
CPU times: user 45min, sys: 67.8 ms, total: 45min
Wall time: 45min

```

## 2.3.4 Xgboost implementation (Best HyperParameter) with asm unigram + asm extracted image features

In [89]:

```

x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
    'subsample':[0.1,0.3,0.5,1]
}
random_cfl=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1,)
random_cfl.fit(X_train_merge, y_train_merge)

```

Fitting 3 folds for each of 10 candidates, totalling 30 fits

```

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:   36.2s
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:   2.9min
[Parallel(n_jobs=-1)]: Done  19 out of  30 | elapsed: 11.5min remaining:  6.6min
[Parallel(n_jobs=-1)]: Done  23 out of  30 | elapsed: 12.8min remaining:  3.9min
[Parallel(n_jobs=-1)]: Done  27 out of  30 | elapsed: 14.0min remaining:  1.6min
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed: 22.9min finished

```

Out[89]:

```

RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                    estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                           colsample_bylevel=1,
                                           colsample_bynode=1,
                                           colsample_bytree=1, gamma=0,
                                           learning_rate=0.1, max_delta_step=0,
                                           max_depth=3, min_child_weight=1,
                                           missing=None, n_estimators=100,
                                           n_jobs=1, nthread=None,
                                           objective='binary:logistic',
                                           random_state=0, reg_alpha=0.5,
                                           seed=None, silent=None, subsample=1,
                                           verbosity=1),
                    iid='warn', n_iter=10, n_jobs=-1,
                    param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                    pre_dispatch='2*n_jobs', random_state=None, refit=True,
                    return_train_score=False, scoring=None, verbose=10)

```

In [90]:

```
print (random_cfl.best_params_)
```

```
{'subsample': 0.5, 'n_estimators': 500, 'max_depth': 3, 'learning_rate': 0.2, 'colsample_bytree': 0.5}
```

In [91]:

```

%%time
# find more about XGBClассifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClассifier
# -----
# default parameters
# class xgboost.XGBClассifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----
x_cfl=XGBClассifier(n_estimators=500,max_depth=3,learning_rate=0.2,colsample_bytree=0.5,subsample=0.5,nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_loss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_loss(y_test_merge, predict_y))

```

```

For values of best alpha = 3000 The train log loss is: 0.007914584198355933
For values of best alpha = 3000 The cross validation log loss is: 0.027262640685997187
For values of best alpha = 3000 The test log loss is: 0.02043044592140415
CPU times: user 6min 33s, sys: 28 ms, total: 6min 33s
Wall time: 6min 33s

```

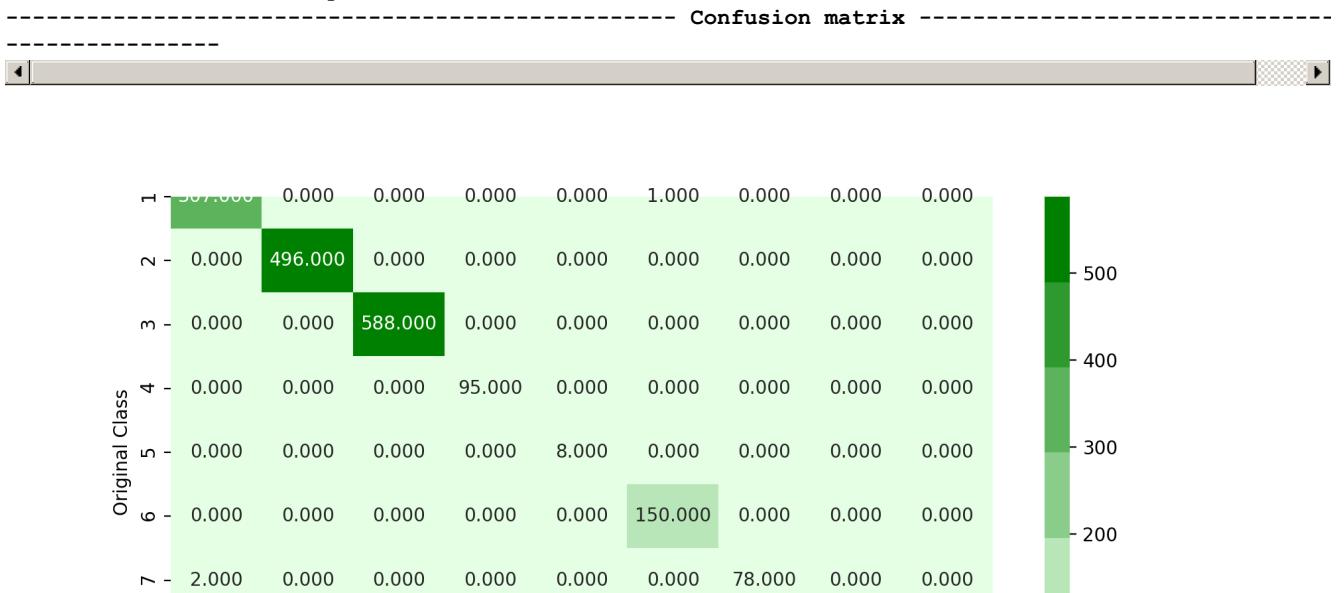
In [92]:

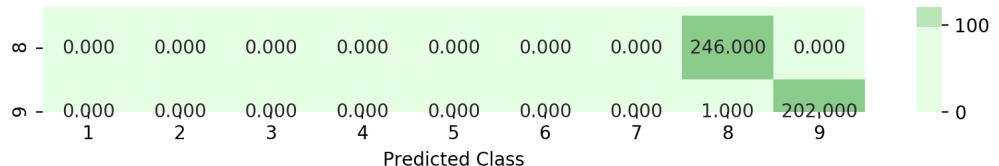
```

plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))

```

Number of misclassified points 0.18399264029438822



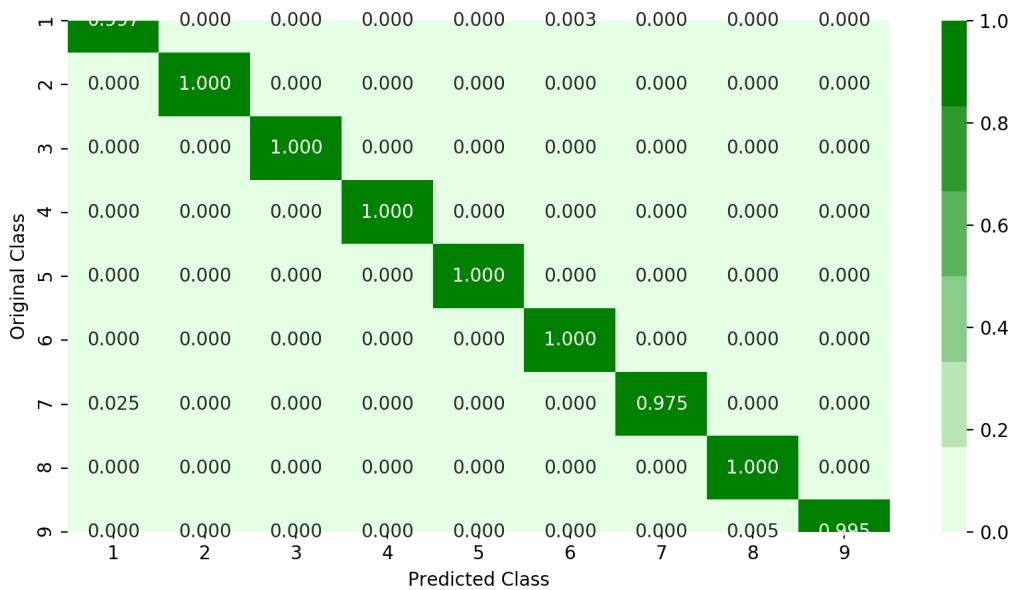


----- Precision matrix -----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

### 2.3.5 Conclusion and Model Comparison (asm unigram + asm extracted image features)

In [93]:

```

table = PrettyTable()
table.field_names = ['Model', 'Best HyperParameter', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']

table.add_row(['Random Forest Classifier', 1000, 0.008896043075915374, 0.01893597948023988, 0.02220239178075276, 0.3219871205151794])
table.add_row(['Xgboost Classifier', 3000, 0.007914584198355933, 0.027262640685997187, 0.02043044592140415, 0.18399264029438822])

print(table)

```

Model	Best HyperParameter	Train Log Loss	CV Log Loss	Test Log Loss	Number of Misclassified Points
Random Forest Classifier	1000	0.008896043075915374	0.01893597948023988	0.02220239178075276	0.3219871205151794
Xgboost Classifier	3000	0.007914584198355933	0.027262640685997187	0.02043044592140415	0.18399264029438822

## 2.4 Implemented ASM unigram + ASM image features + ByteFile unigram

### 2.4.1 Prepare, Merge and Split (ASM unigram + ASM image features + ByteFile unigram) data

In [52]:

```

print(data_reduced.shape)
print(result_asm.shape)
print(result.shape)

```

```
(10868, 501)
(10868, 54)
(10868, 261)
```

In [53]:

```
data_reduced.columns
```

Out[53]:

```

Index(['ID', 'ASM_1', 'ASM_3', 'ASM_4', 'ASM_14', 'ASM_20', 'ASM_21', 'ASM_23',
       'ASM_24', 'ASM_25',
       ...
       'ASM_984', 'ASM_988', 'ASM_989', 'ASM_990', 'ASM_991', 'ASM_994',
       'ASM_995', 'ASM_996', 'ASM_997', 'ASM_998'],
       dtype='object', length=501)

```

In [54]:

```
result_asm.columns
```

Out[54]:

```

Index(['ID', 'HEADER:', '.text:', '.Pav:', '.idata:', '.data:', '.bss:',
       '.rdata:', '.edata:', '.rsrc:', '.tls:', '.reloc:', '.BSS:', '.CODE',
       'jmp', 'mov', 'retf', 'push', 'pop', 'xor', 'retn', 'nop', 'sub', 'inc',
       'dec', 'add', 'imul', 'xchg', 'or', 'shr', 'cmp', 'call', 'shl', 'ror',
       'rol', 'jnb', 'jz', 'rtn', 'lea', 'movzx', '.dll', 'std::', ':dword',
       'edx', 'esi', 'eax', 'ebx', 'ecx', 'edi', 'ebp', 'esp', 'eip', 'Class',
       'size'],
       dtype='object')

```

In [55]:

```
result.columns
```

Out[55]:

```
Index(['ID', '0', '1', '2', '3', '4', '5', '6', '7', '8',
       ...
       'fa', 'fb', 'fc', 'fd', 'fe', 'ff', '??', 'Unnamed: 0', 'size',
       'Class'],
      dtype='object', length=261)
```

In [56]:

```
result_y
```

Out[56]:

```
0      1
1      1
2      1
3      1
4      1
..
10863   2
10864   2
10865   2
10866   2
10867   2
Name: Class, Length: 10868, dtype: int64
```

In [57]:

```
result_x = pd.merge(result_asm, data_reduced, on='ID', how='left')
result_x = pd.merge(result_x, result.drop('size', axis=1), on='ID', how='left')
# result_y = result_x['Class']
result_x = result_x.drop(['ID', 'rtn', '.BSS:', '.CODE'], axis=1)
# # result_x = result_x.drop(['ID', 'Class'], axis=1)
result_x.head()
```

Out[57]:

	HEADER:	.text:	.Pav:	.idata:	.data:	.bss:	.rdata:	.edata:	.rsrc:	.tls:	...	f9	fa	fb	fc
0	0.107345	0.001092	0.0	0.000761	0.000023	0.0	0.000084	0.0	0.000072	0.0	...	0.002121	0.001886	0.002272	0.013032
1	0.096045	0.001230	0.0	0.000617	0.000019	0.0	0.000000	0.0	0.000072	0.0	...	0.001264	0.000972	0.001255	0.004003
2	0.096045	0.000627	0.0	0.000300	0.000017	0.0	0.000038	0.0	0.000072	0.0	...	0.001194	0.001094	0.003702	0.004244
3	0.096045	0.000333	0.0	0.000258	0.000008	0.0	0.000000	0.0	0.000072	0.0	...	0.001259	0.000963	0.001250	0.003853
4	0.096045	0.000590	0.0	0.000353	0.000068	0.0	0.000000	0.0	0.000072	0.0	...	0.001294	0.001061	0.001105	0.004509

5 rows × 809 columns

In [58]:

```
result_x.shape
```

Out[58]:

```
(10868, 809)
```

In [59]:

```
X_train, X_test_merge, y_train, y_test_merge = train_test_split(result_x, result_y,stratify=result_y,test_size=0.20)
X_train_merge, X_cv_merge, y_train_merge, y_cv_merge = train_test_split(X_train, y_train,stratify=y_train,test_size=0.20)
```

## 2.4.2 Random Forest Classifier ASM unigram + ASM image features + ByteFile unigram

In [104]:

```
%%time
plt.close()
# -----
# default parameters
# sklearn.ensemble.RandomForestClassifier(n_estimators=10, criterion='gini', max_depth=None, min_samples_split=2,
# min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
# min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=1, random_state=None,
# verbose=0, warm_start=False,
# class_weight=None)

# Some of methods of RandomForestClassifier()
# fit(X, y, [sample_weight]) Fit the SVM model according to the given training data.
# predict(X) Perform classification on samples in X.
# predict_proba (X) Perform classification on samples in X.

# some of attributes of RandomForestClassifier()
# feature_importances_ : array of shape = [n_features]
# The feature importances (the higher, the more important the feature).

# -----
# video link: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forest-and-their-construction-2/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
from sklearn.ensemble import RandomForestClassifier
for i in alpha:
    r_cfl=RandomForestClassifier(n_estimators=i,random_state=42,n_jobs=-1)
    r_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=r_cfl.classes_, eps=1e-15))

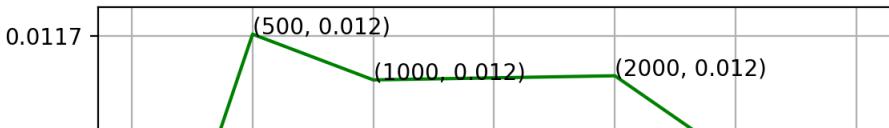
for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

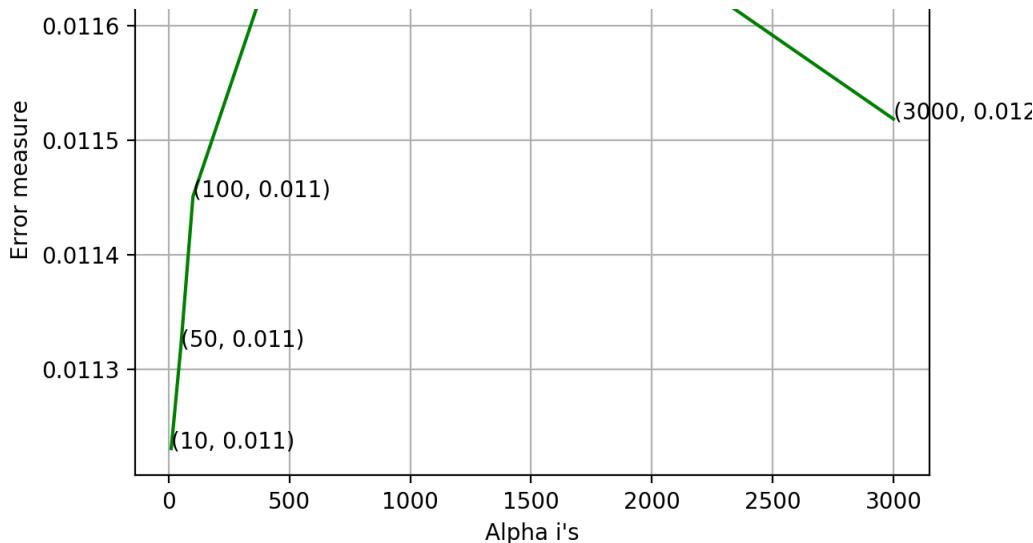
best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c = 10 is 0.01123057551993269
log_loss for c = 50 is 0.011319907320875825
log_loss for c = 100 is 0.011450850380533962
log_loss for c = 500 is 0.011701246562309798
log_loss for c = 1000 is 0.011661285365579324
log_loss for c = 2000 is 0.011664893517260147
log_loss for c = 3000 is 0.011518735957461427
```

Cross Validation Error for each alpha





CPU times: user 16min 36s, sys: 23.8 s, total: 17min  
Wall time: 2min 55s

In [105]:

```
%time
plt.close()
r_cfl=RandomForestClassifier(n_estimators=alpha[best_alpha],random_state=42,n_jobs=-1)
r_cfl.fit(X_train_merge,y_train_merge)
sig_clf = CalibratedClassifierCV(r_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_merge, predict_y))
```

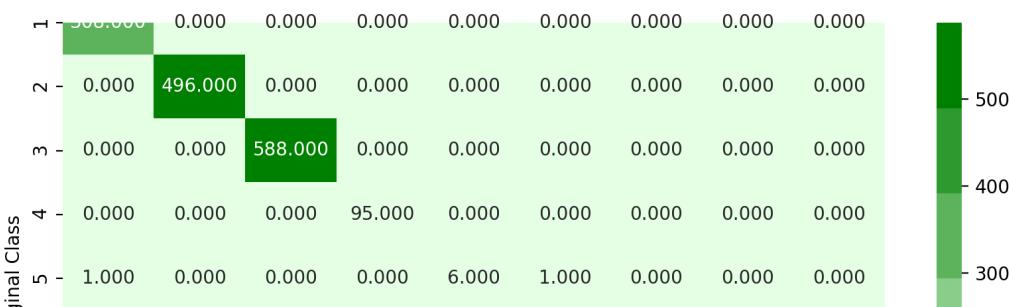
For values of best alpha = 10 The train log loss is: 0.009189865009960088  
For values of best alpha = 10 The cross validation log loss is: 0.01123057551993269  
For values of best alpha = 10 The test log loss is: 0.012817156034718172  
CPU times: user 2.08 s, sys: 105 ms, total: 2.19 s  
Wall time: 2.2 s

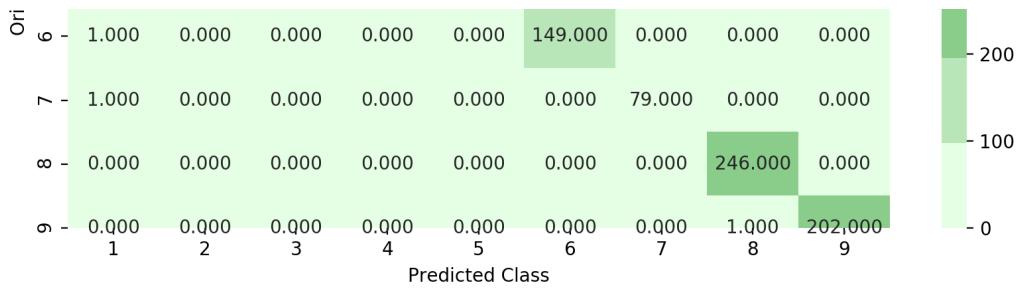
In [106]:

```
plt.close()
plot_confusion_matrix(y_test_merge,sig_clf.predict(X_test_merge))
```

Number of misclassified points 0.22999080036798528

----- Confusion matrix -----  
-----





Precision matrix



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

Recall matrix



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

## 2.4.3 XgBoost Classifier ASM unigram + ASM image features + ByteFile unigram

In [107]:

```
%%time
plt.close()
# Training a hyper-parameter tuned Xg-Boost regressor on our train data

# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
min_child_weight=1,
# max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
reg_lambda=1,
# scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----


alpha=[10,50,100,500,1000,2000,3000]
cv_log_error_array=[]
for i in alpha:
    x_cfl=XGBClassifier(n_estimators=i)
    x_cfl.fit(X_train_merge,y_train_merge)
    sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
    sig_clf.fit(X_train_merge, y_train_merge)
    predict_y = sig_clf.predict_proba(X_cv_merge)
    cv_log_error_array.append(log_loss(y_cv_merge, predict_y, labels=x_cfl.classes_, eps=1e-15))

for i in range(len(cv_log_error_array)):
    print ('log_loss for c = ',alpha[i],'is',cv_log_error_array[i])

best_alpha = np.argmin(cv_log_error_array)

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],np.round(txt,3)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()

log_loss for c = 10 is 0.020072573934995657
log_loss for c = 50 is 0.005998591407924537
log_loss for c = 100 is 0.005086113372305249
log_loss for c = 500 is 0.005086670978063485
log_loss for c = 1000 is 0.005086547485919617
log_loss for c = 2000 is 0.005086592539420503
log_loss for c = 3000 is 0.0050865362831616495
```

```
CPU times: user 2h 43min 59s, sys: 574 ms, total: 2h 44min
Wall time: 2h 44min
```

In [111]:

```
%%time
plt.close()
x_cfl=XGBClassifier(n_estimators=alpha[best_alpha],nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)

predict_y = sig_clf.predict_proba(X_train_merge)
print ('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_merge, predict_y))
```

```
For values of best alpha = 100 The train log loss is: 0.0052644779291518866
For values of best alpha = 100 The cross validation log loss is: 0.005086113372305249
For values of best alpha = 100 The test log loss is: 0.005282303659991528
CPU times: user 5min 18s, sys: 110 ms, total: 5min 18s
Wall time: 5min 18s
```

## 2.4.5 XgBoost Classifier (Best HyperParameter) ASM unigram + ASM image features + ByteFile unigram

In [60]:

```
x_cfl=XGBClassifier()

prams={
    'learning_rate':[0.01,0.03,0.05,0.1,0.15,0.2],
    'n_estimators':[100,200,500,1000,2000],
    'max_depth':[3,5,10],
    'colsample_bytree':[0.1,0.3,0.5,1],
    'subsample':[0.1,0.3,0.5,1]
}
random_cfl=RandomizedSearchCV(x_cfl,param_distributions=prams,verbose=10,n_jobs=-1,)
random_cfl.fit(X_train_merge, y_train_merge)
```

```
Fitting 3 folds for each of 10 candidates, totalling 30 fits
```

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 8 concurrent workers.
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:   24.7s
[Parallel(n_jobs=-1)]: Done   2 tasks      | elapsed:   4.5min
```

```
[Parallel(n_jobs=-1)]: Done  9 tasks   | elapsed:  4.5min
[Parallel(n_jobs=-1)]: Done 19 out of 30 | elapsed:  9.1min remaining:  5.3min
[Parallel(n_jobs=-1)]: Done 23 out of 30 | elapsed: 10.0min remaining:  3.0min
[Parallel(n_jobs=-1)]: Done 27 out of 30 | elapsed: 10.6min remaining:  1.2min
[Parallel(n_jobs=-1)]: Done 30 out of 30 | elapsed: 11.1min finished
```

Out[60]:

```
RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
                    estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                            colsample_bylevel=1,
                                            colsample_bynode=1,
                                            colsample_bytree=1, gamma=0,
                                            learning_rate=0.1, max_delta_step=0,
                                            max_depth=3, min_child_weight=1,
                                            missing=None, n_estimators=100,
                                            n_jobs=1, nthread=None,
                                            objective='binary:logistic',
                                            random_state=0, reg_alpha=0.5,
                                            seed=None, silent=None, subsample=1,
                                            verbosity=1),
                    iid='warn', n_iter=10, n_jobs=-1,
                    param_distributions={'colsample_bytree': [0.1, 0.3, 0.5, 1],
                                         'learning_rate': [0.01, 0.03, 0.05, 0.1,
                                                          0.15, 0.2],
                                         'max_depth': [3, 5, 10],
                                         'n_estimators': [100, 200, 500, 1000,
                                                          2000],
                                         'subsample': [0.1, 0.3, 0.5, 1]},
                    pre_dispatch='2*n_jobs', random_state=None, refit=True,
                    return_train_score=False, scoring=None, verbose=10)
```

In [61]:

```
print (random_cfl.best_params_)
```

```
{'subsample': 1, 'n_estimators': 1000, 'max_depth': 3, 'learning_rate': 0.03, 'colsample_bytree': 0.5}
```

In [71]:

```
%%time
# find more about XGBClassifier function here
http://xgboost.readthedocs.io/en/latest/python/python_api.html?#xgboost.XGBClassifier
# -----
# default parameters
# class xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100, silent=True,
# # objective='binary:logistic', booster='gbtree', n_jobs=1, nthread=None, gamma=0,
# min_child_weight=1,
# # max_delta_step=0, subsample=1, colsample_bytree=1, colsample_bylevel=1, reg_alpha=0,
# reg_lambda=1,
# # scale_pos_weight=1, base_score=0.5, random_state=0, seed=None, missing=None, **kwargs)

# some of methods of RandomForestRegressor()
# fit(X, y, sample_weight=None, eval_set=None, eval_metric=None, early_stopping_rounds=None, verbose=True, xgb_model=None)
# get_params([deep]) Get parameters for this estimator.
# predict(data, output_margin=False, ntree_limit=0) : Predict with data. NOTE: This function is not thread safe.
# get_score(importance_type='weight') -> get the feature importance
# -----
# video link2: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/
# -----

x_cfl=XGBClassifier(n_estimators=1000,max_depth=3,learning_rate=0.03,colsample_bytree=0.5,subsample=1,nthread=-1)
x_cfl.fit(X_train_merge,y_train_merge,verbose=True)
sig_clf = CalibratedClassifierCV(x_cfl, method="sigmoid")
sig_clf.fit(X_train_merge, y_train_merge)
```

CPU times: user 18min 13s, sys: 260 ms, total: 18min 13s
Wall time: 18min 13s

Out[71]:

```
CalibratedClassifierCV(base_estimator=XGBClassifier(base_score=0.5,
                                                    booster='gbtree',
                                                    colsample_bylevel=1,
                                                    colsample_bynode=1,
                                                    colsample_bytree=0.5,
                                                    gamma=0, learning_rate=0.03,
                                                    max_delta_step=0,
                                                    max_depth=3,
                                                    min_child_weight=1,
                                                    missing=None,
                                                    n_estimators=1000, n_jobs=1,
                                                    nthread=-1,
                                                    objective='multi:softprob',
                                                    random_state=0, reg_alpha=0,
                                                    reg_lambda=1,
                                                    scale_pos_weight=1,
                                                    seed=None, silent=None,
                                                    subsample=1, verbosity=1),
                      cv='warn', method='sigmoid')
```

In [72]:

```

alpha=[10,50,100,500,1000,2000,3000]
best_alpha = 2
predict_y = sig_clf.predict_proba(X_train_merge)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss
is:",log_loss(y_train_merge, predict_y))
predict_y = sig_clf.predict_proba(X_cv_merge)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_lo
ss(y_cv_merge, predict_y))
predict_y = sig_clf.predict_proba(X_test_merge)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss
is:",log_loss(y_test_merge, predict_y))

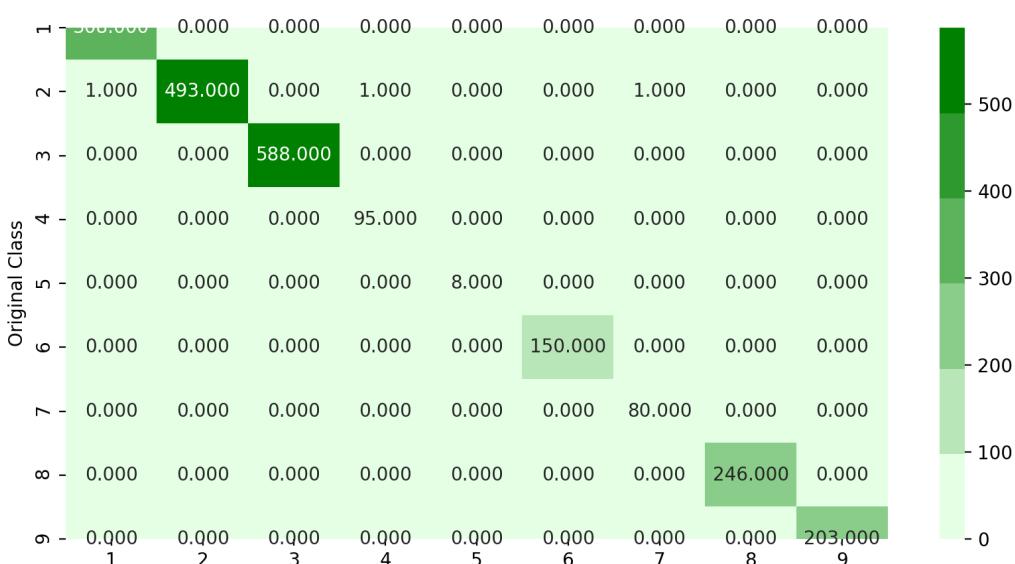
```

```
For values of best alpha = 100 The train log loss is: 0.004408321855502199  
For values of best alpha = 100 The cross validation log loss is: 0.007235101  
For values of best alpha = 100 The test log loss is: 0.01083806227335543
```

In [73]:

```
plt.close()  
plot confusion matrix(y test merge,sig clf.predict(X test merge))
```

Number of misclassified points 0.13799448022079117



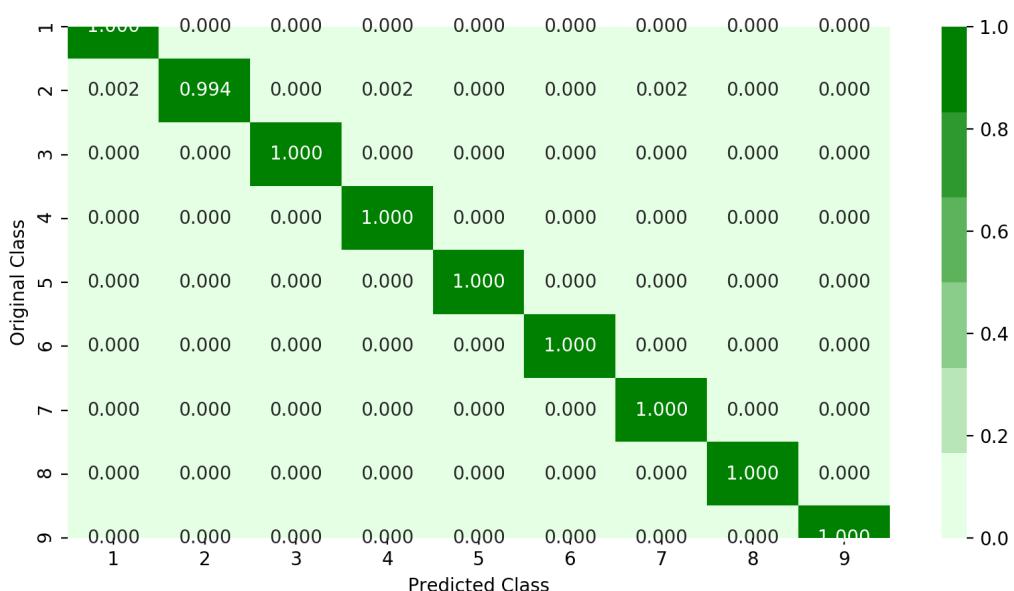
Predicted Class

----- Precision matrix -----



Sum of columns in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

----- Recall matrix -----



Sum of rows in precision matrix [1. 1. 1. 1. 1. 1. 1. 1. 1.]

## 2.4.6 Conclusion and Model Comparision (ASM unigram + ASM image features + ByteFile unigram)

In [94]:

```
from prettytable import PrettyTable
table = PrettyTable()
table.field_names = ['Model', 'Best HyperParameter', 'Train Log Loss', 'CV Log Loss', 'Test Log Loss', 'Number of Misclassified Points']
```

```

ss , 'Number of Misclassified Points']

table.add_row(['Random Forest Classifier', 10, 0.009189865009960088, 0.01123057551993269,
0.012817156034718172, 0.22999080036798528])
table.add_row(['XgBoost Classifier', 100, 0.004408321855502199, 0.007235101907919579,
0.01083806227335543, 0.13799448022079117])
table.add_row(['XgBoost Classifier With Best HyperParameter', 100, 0.0052644779291518866,
0.005086113372305249, 0.005282303659991528, 0.12799448022079117])

print(table)

```

Log Loss	Model	Best HyperParameter	Train Log Loss	C
Test Log Loss		Number of Misclassified Points		
3057551993269	Random Forest Classifier	10	0.009189865009960088	0.01123057551993269
0.012817156034718172		0.22999080036798528		
0.007235101907919579	XgBoost Classifier	100	0.004408321855502199	
0.01083806227335543		0.13799448022079117		
0.005086113372305249	XgBoost Classifier With Best HyperParameter	100	0.0052644779291518866	0.005
0.005282303659991528		0.12799448022079116		

- Conclusion

- I followed following feature engineering approach:
- Tried Bigram of ByteFiles
- ASM image features + bytes uni-gram features
- ASM unigram + ASM extracted image features
- ASM unigram + ASM image features + ByteFile unigram
- ASM unigram + ASM image features + ByteFile unigram (XgBoost Classifier With Best HyperParameter) reduced the log-loss < 0.01 (0.005282303659991528)

In [ ]: