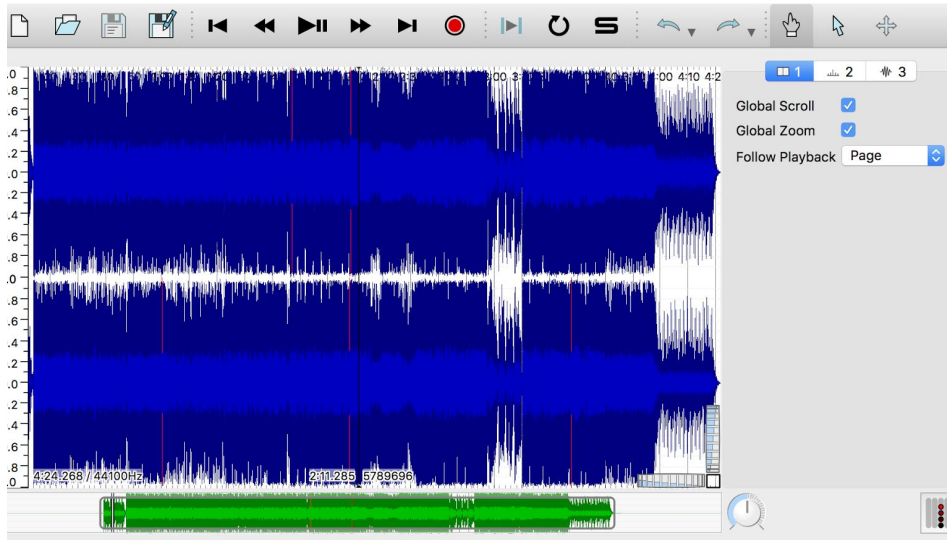


CSC475_Assignment05

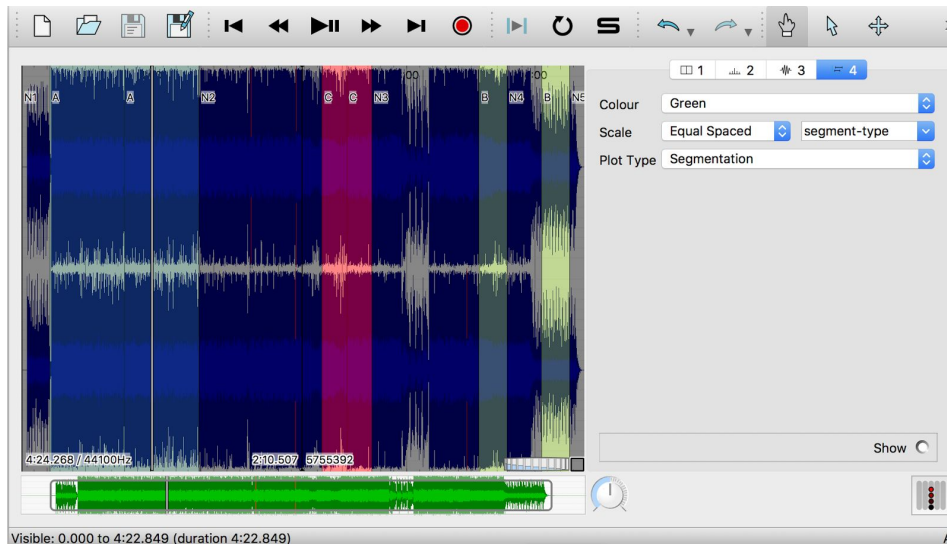
Q1.

a) I choosed a1.mp3 and a2.mp3.

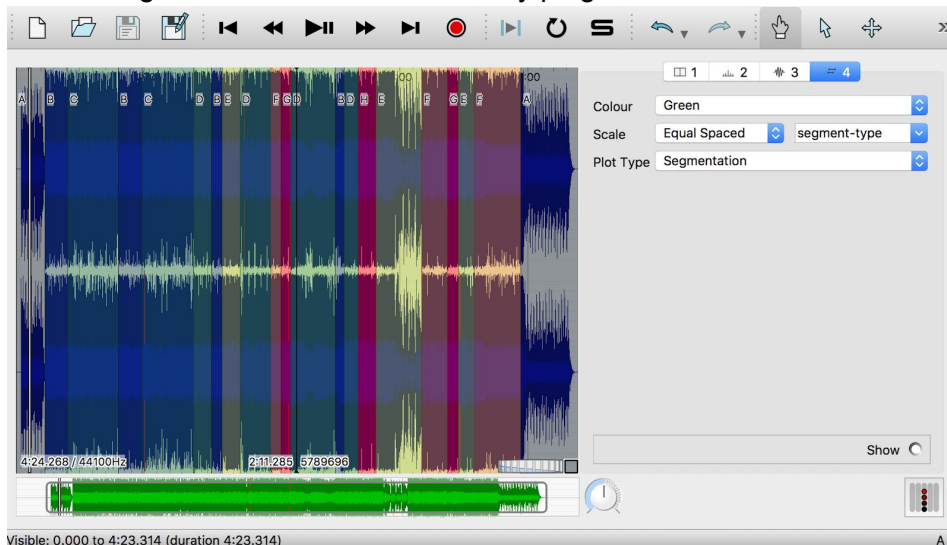
Original a1.mp3:



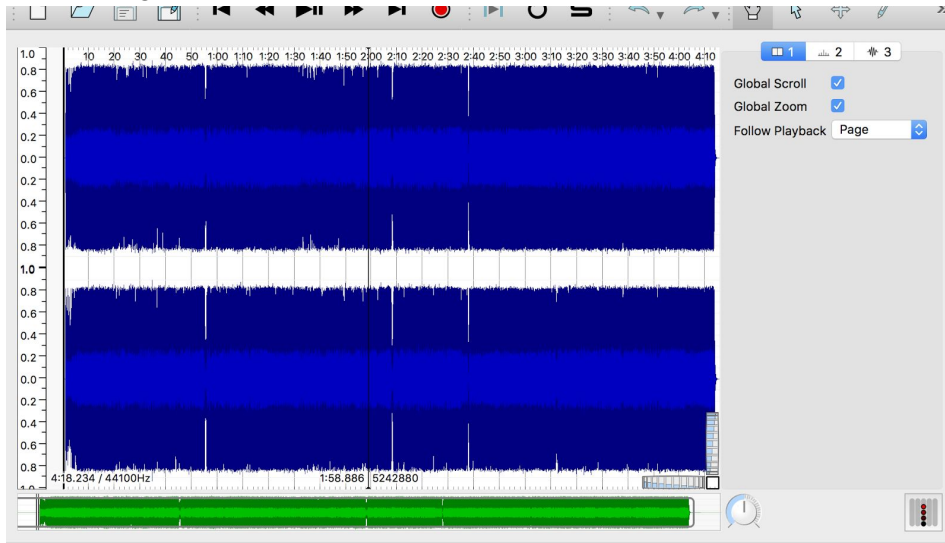
Segmentino plugin:



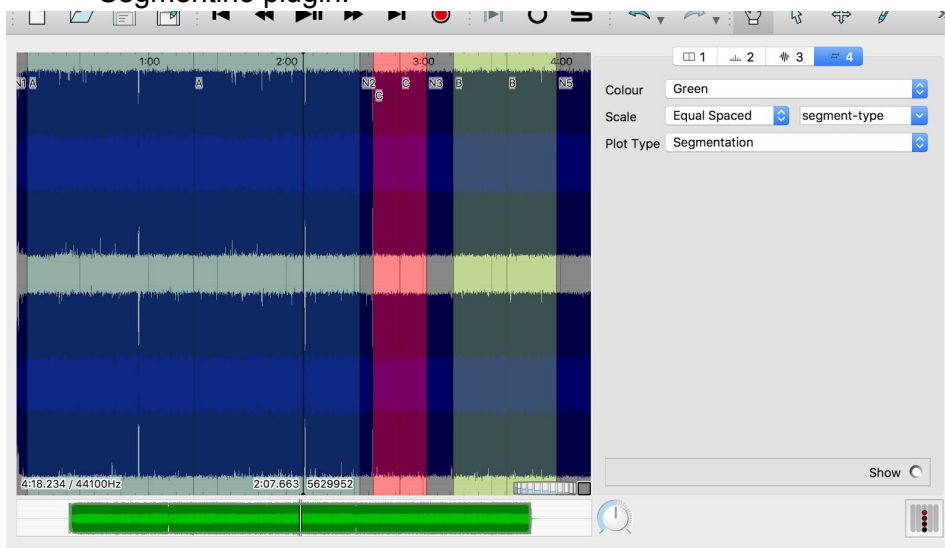
Segmenter from the Queen Mary plugin set:



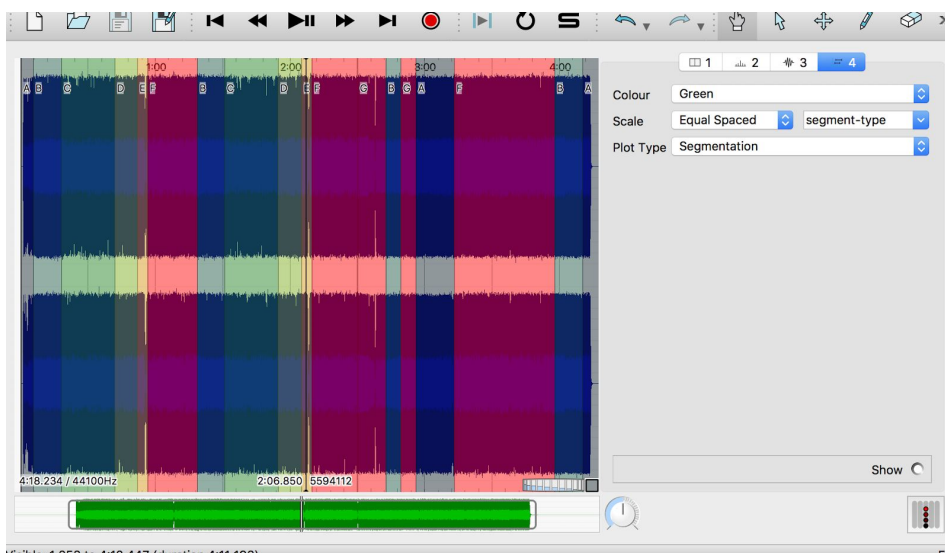
Original a2.mp3



Segmentino plugin:



Segmenter from the Queen Mary plugin set:



b)

Queen Marry for a1.mp3:

```
$ cd /Users/llicht/Documents/sonic-annotator-1.4-osx-amd64
$ ./sonic-annotator -l
$ ./sonic-annotator -s vamp:qm-vamp-plugins:qm-segmenter:segmentation > qmtest1.n3
$ ./sonic-annotator -t qmtest1.n3 a1.mp3 -w csv --csv-stdout
Have audio source: "a1.mp3"
Decoding a1.mp3... Done
File or URL "a1.mp3" opened successfully
Taking default channel count of 2 from audio file
Taking default sample rate of 44100Hz from audio file
(Note: Default may be overridden by transforms)
```

RDFTTransformFactory: NOTE: Transform is:

```
<transform
  id="vamp:qm-vamp-plugins:qm-segmenter:segmentation"
  pluginVersion="3"
  program=""
  stepSize="8820"
  blockSize="26460"
  windowType="hanning"
  startTime="0.000000000"
  duration="0.000000000"
  sampleRate="0">
<parameter name="featureType" value="1"/>
<parameter name="nSegmentTypes" value="10"/>
<parameter name="neighbourhoodLimit" value="4"/>
</transform>
```

NOTE: Transform does not specify a sample rate, using default rate of 44100

NOTE: Loaded and initialised plugin for transform

"vamp:qm-vamp-plugins:qm-segmenter:segmentation" with plugin step size 8820 and block size 26460 (adapter step and block size 1024)

Extracting features for: "a1.mp3"

Audio file "a1.mp3": 2ch at 44100Hz

Extracting and writing features... Done

```
"a1.mp3",0.000000000,13.600000000,1,"A"
,13.600000000,10.600000000,2,"B"
,24.200000000,23.600000000,3,"C"
,47.800000000,11.200000000,4,"D"
,59.000000000,24.200000000,3,"C"
,83.200000000,16.600000000,5,"E"
,99.800000000,6.800000000,6,"F"
,106.600000000,4.200000000,2,"B"
,110.800000000,15.400000000,7,"G"
,126.200000000,26.000000000,5,"E"
```

,152.200000000,14.400000000,7,"G"
,166.600000000,10.400000000,8,"H"
,177.000000000,12.200000000,1,"A"
,189.200000000,29.200000000,8,"H"
,218.400000000,19.400000000,6,"F"
,237.800000000,26.000000000,1,"A"

Segmentino for a1.mp3:

```
$ ./sonic-annotator -s vamp:segmentino:segmentino:segmentation > sgtest1.n3
```

```
$ ./sonic-annotator -t sgtest1.n3 a1.mp3 -w csv --csv-stdout
```

Have audio source: "a1.mp3"

Decoding a1.mp3... Done

File or URL "a1.mp3" opened successfully

Taking default channel count of 2 from audio file

Taking default sample rate of 44100Hz from audio file

(Note: Default may be overridden by transforms)

RDFTTransformFactory: NOTE: Transform is:

<transform

id="vamp:segmentino:segmentino:segmentation"

pluginVersion="2"

program=""

stepSize="512"

blockSize="16384"

windowType="hanning"

startTime="0.000000000"

duration="0.000000000"

sampleRate="0"/>

NOTE: Transform does not specify a sample rate, using default rate of 44100

NOTE: Loaded and initialised plugin for transform

"vamp:segmentino:segmentino:segmentation" with plugin step size 512 and block size 16384 (adapter step and block size 1024)

Extracting features for: "a1.mp3"

Audio file "a1.mp3": 2ch at 44100Hz

Extracting and writing features... Done

"a1.mp3",0.000000000,2.589024943,0,"N1"

,2.589024943,11.575147392,4,"D"

,14.164172335,11.192018141,4,"D"

,25.356190476,11.575147392,1,"A"

,36.931337868,11.563537415,1,"A"

,48.494875283,11.575147392,1,"A"

,60.070022675,11.563537415,1,"A"

,71.633560090,10.123900227,1,"A"

,81.757460317,58.909024943,0,"N8"

,140.666485260,11.563537415,3,"C"

,152.230022675,11.575147393,3,"C"
,163.805170068,52.059138322,0,"N9"
,215.864308390,11.551927437,2,"B"
,227.416235827,11.563537415,2,"B"
,238.979773242,10.123900227,2,"B"
,249.103673469,14.454421769,0,"N12"

Queen Marry for a2.mp3:

```
$ ./sonic-annotator -s vamp:qm-vamp-plugins:qm-segmenter:segmentation  
$ ./sonic-annotator -s vamp:qm-vamp-plugins:qm-segmenter:segmentation > qmtest2.n3  
$ ./sonic-annotator -t qmtest2.n3 a2.mp3 -w csv --csv-stdout  
Have audio source: "a2.mp3"  
Decoding a2.mp3... Done  
File or URL "a2.mp3" opened successfully  
Taking default channel count of 2 from audio file  
Taking default sample rate of 44100Hz from audio file  
(Note: Default may be overridden by transforms)
```

RDFTTransformFactory: NOTE: Transform is:

```
<transform  
  id="vamp:qm-vamp-plugins:qm-segmenter:segmentation"  
  pluginVersion="3"  
  program=""  
  stepSize="8820"  
  blockSize="26460"  
  windowType="hanning"  
  startTime="0.000000000"  
  duration="0.000000000"  
  sampleRate="0">  
<parameter name="featureType" value="1"/>  
<parameter name="nSegmentTypes" value="10"/>  
<parameter name="neighbourhoodLimit" value="4"/>  
</transform>
```

NOTE: Transform does not specify a sample rate, using default rate of 44100

NOTE: Loaded and initialised plugin for transform

"vamp:qm-vamp-plugins:qm-segmenter:segmentation" with plugin step size 8820 and block size 26460 (adapter step and block size 1024)

Extracting features for: "a2.mp3"

Audio file "a2.mp3": 2ch at 44100Hz

Extracting and writing features... Done

"a2.mp3",0.000000000,18.000000000,1,"A"
,18.000000000,22.600000000,2,"B"
,40.600000000,12.400000000,3,"C"
,53.000000000,6.000000000,4,"D"

,59.000000000,7.000000000,5,"E"
,66.000000000,4.600000000,4,"D"
,70.600000000,7.600000000,5,"E"
,78.200000000,12.200000000,1,"A"
,90.400000000,23.200000000,2,"B"
,113.600000000,12.400000000,3,"C"
,126.000000000,5.800000000,4,"D"
,131.800000000,7.000000000,5,"E"
,138.800000000,4.600000000,4,"D"
,143.400000000,6.600000000,5,"E"
,150.000000000,25.800000000,6,"F"
,175.800000000,18.400000000,7,"G"
,194.200000000,8.600000000,5,"E"
,202.800000000,4.600000000,4,"D"
,207.400000000,7.000000000,5,"E"
,214.400000000,4.600000000,4,"D"
,219.000000000,7.000000000,5,"E"
,226.000000000,4.600000000,4,"D"
,230.600000000,7.400000000,5,"E"
,238.000000000,12.600000000,1,"A"
,250.600000000,7.200000000,7,"G"

Segmentino for a2.mp3:

```
$ ./sonic-annotator -s vamp:segmentino:segmentino:segmentation > sgtest2.n3
```

```
$ ./sonic-annotator -t sgtest2.n3 a2.mp3 -w csv --csv-stdout
```

Have audio source: "a2.mp3"

Decoding a2.mp3... Done

File or URL "a2.mp3" opened successfully

Taking default channel count of 2 from audio file

Taking default sample rate of 44100Hz from audio file

(Note: Default may be overridden by transforms)

RDFTransformFactory: NOTE: Transform is:

<transform

id="vamp:segmentino:segmentino:segmentation"

pluginVersion="2"

program=""

stepSize="512"

blockSize="16384"

windowType="hanning"

startTime="0.000000000"

duration="0.000000000"

sampleRate="0"/>

NOTE: Transform does not specify a sample rate, using default rate of 44100

NOTE: Loaded and initialised plugin for transform

"vamp:segmentino:segmentino:segmentation" with plugin step size 512 and block size 16384 (adapter step and block size 1024)

Extracting features for: "a2.mp3"

Audio file "a2.mp3": 2ch at 44100Hz

Extracting and writing features... Done

"a2.mp3",0.000000000,7.093696145,0,"N1"

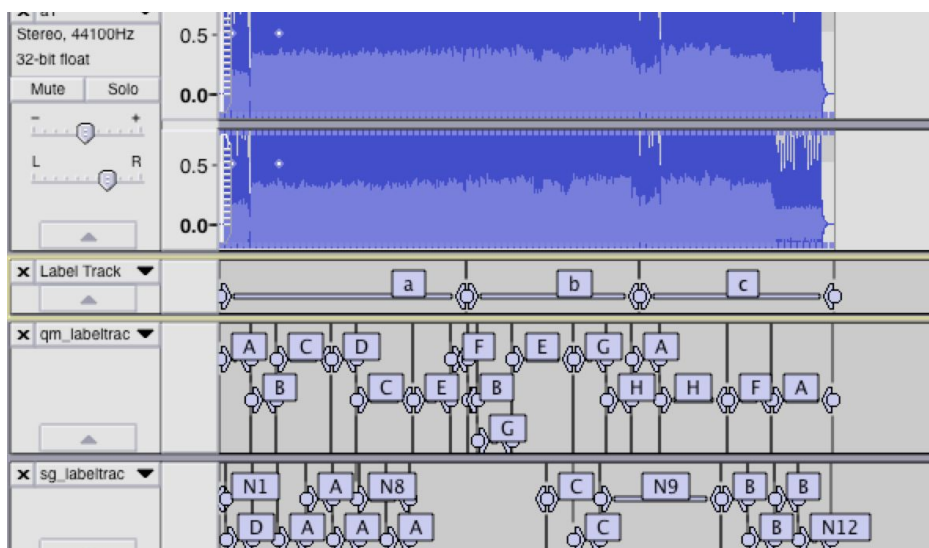
,7.093696145,72.701678004,1,"A"

,79.795374149,71.796099773,1,"A"

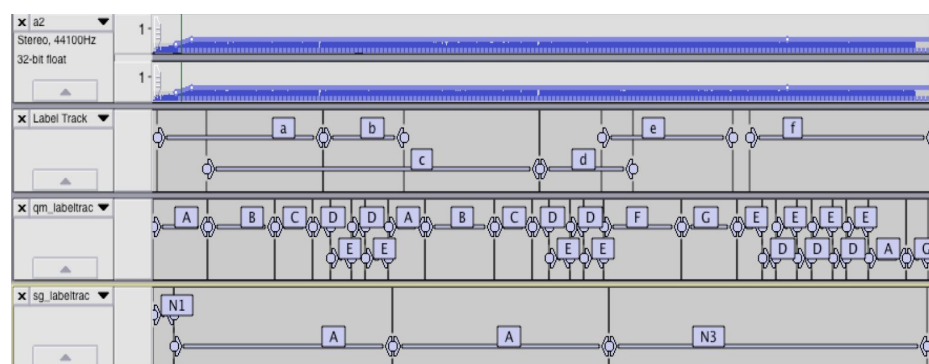
,151.591473922,105.941043084,0,"N3"

c)

three segmentation for a1.mp3:



three segmentation for a2.mp3:



I annotate and named the segmenta randomly. So my manual segmentation does not sound have any regulation. However, the Queen Mary Segmenter seems annotate and named the segmenta by tone, which means the segmentation that have same name have similar tone. And Segmentino annotate and named the segmenta by beat.

Q2

a)

```
data, rate = librosa.load('a2.mp3')

segmenter = vamp.collect(data, rate, "qm-vamp-plugins:qm-segmenter")
segmentino = vamp.collect(data, rate, "segmentino:segmentino")

print("segmenter: ")
for i in segmenter["list"]:
    print(i)
print("segmentino: ")
for j in segmentino["list"]:
    print(j)
```

The code is same for a1.

a1.mp3:

segmenter:

```
{'timestamp': 0.000000000, 'duration': 13.600000000, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 13.600000000, 'duration': 11.200000000, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 24.800000000, 'duration': 23.400000000, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 48.200000000, 'duration': 11.400000000, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 59.600000000, 'duration': 23.600000000, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 83.200000000, 'duration': 5.200000000, 'label': 'D', 'values': array([ 4.], dtype=float32)}
{'timestamp': 88.400000000, 'duration': 7.200000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 95.600000000, 'duration': 5.000000000, 'label': 'D', 'values': array([ 4.], dtype=float32)}
{'timestamp': 100.600000000, 'duration': 6.400000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 107.000000000, 'duration': 12.200000000, 'label': 'G', 'values': array([ 7.], dtype=float32)}
{'timestamp': 119.200000000, 'duration': 5.200000000, 'label': 'H', 'values': array([ 8.], dtype=float32)}
{'timestamp': 124.400000000, 'duration': 25.000000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 149.400000000, 'duration': 4.400000000, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 153.800000000, 'duration': 13.000000000, 'label': 'G', 'values': array([ 7.], dtype=float32)}
{'timestamp': 166.800000000, 'duration': 5.400000000, 'label': 'I', 'values': array([ 9.], dtype=float32)}
{'timestamp': 172.200000000, 'duration': 17.600000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 189.800000000, 'duration': 10.200000000, 'label': 'H', 'values': array([ 8.], dtype=float32)}
{'timestamp': 200.000000000, 'duration': 7.600000000, 'label': 'I', 'values': array([ 9.], dtype=float32)}
{'timestamp': 207.600000000, 'duration': 7.800000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 215.400000000, 'duration': 22.600000000, 'label': 'H', 'values': array([ 8.], dtype=float32)}
{'timestamp': 238.000000000, 'duration': 26.200000000, 'label': 'A', 'values': array([ 1.], dtype=float32)}
```

segmentino:

```
{'timestamp': 0.000000000, 'duration': 13.421133786, 'label': 'N1', 'values': array([ 0.], dtype=float32)}
{'timestamp': 13.421133786, 'duration': 34.702222223, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 48.123356009, 'duration': 34.713832199, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 82.837188208, 'duration': 57.817687075, 'label': 'N2', 'values': array([ 0.], dtype=float32)}
{'timestamp': 140.654875283, 'duration': 11.737687075, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 152.392562358, 'duration': 11.435827664, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 163.828390022, 'duration': 50.201541950, 'label': 'N4', 'values': array([ 0.], dtype=float32)}
{'timestamp': 214.029931972, 'duration': 13.014784581, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 227.044716553, 'duration': 15.905668934, 'label': 'N5', 'values': array([ 0.], dtype=float32)}
{'timestamp': 242.950385487, 'duration': 13.003174603, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 255.953560090, 'duration': 7.952834467, 'label': 'N6', 'values': array([ 0.], dtype=float32)}
```

a2.mp3:

segmenter:

```
{'timestamp': 0.000000000, 'duration': 18.000000000, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 18.000000000, 'duration': 22.400000000, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 40.400000000, 'duration': 11.600000000, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 52.000000000, 'duration': 5.800000000, 'label': 'D', 'values': array([ 4.], dtype=float32)}
{'timestamp': 57.800000000, 'duration': 8.200000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
```



```
{'timestamp': 66.000000000, 'duration': 4.800000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 70.800000000, 'duration': 7.200000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 78.000000000, 'duration': 12.800000000, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 90.800000000, 'duration': 22.400000000, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 113.200000000, 'duration': 12.400000000, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 125.600000000, 'duration': 6.200000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 131.800000000, 'duration': 7.000000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 138.800000000, 'duration': 4.600000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 143.400000000, 'duration': 6.000000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 149.400000000, 'duration': 13.000000000, 'label': 'G', 'values': array([ 7.], dtype=float32)}
{'timestamp': 162.400000000, 'duration': 6.600000000, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 169.000000000, 'duration': 6.400000000, 'label': 'G', 'values': array([ 7.], dtype=float32)}
{'timestamp': 175.400000000, 'duration': 19.000000000, 'label': 'D', 'values': array([ 4.], dtype=float32)}
{'timestamp': 194.400000000, 'duration': 8.400000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 202.800000000, 'duration': 4.600000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 207.400000000, 'duration': 7.400000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 214.800000000, 'duration': 4.400000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 219.200000000, 'duration': 6.800000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 226.000000000, 'duration': 4.800000000, 'label': 'F', 'values': array([ 6.], dtype=float32)}
{'timestamp': 230.800000000, 'duration': 7.200000000, 'label': 'E', 'values': array([ 5.], dtype=float32)}
{'timestamp': 238.000000000, 'duration': 12.800000000, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 250.800000000, 'duration': 7.400000000, 'label': 'D', 'values': array([ 4.], dtype=float32)}
segmentino:
{'timestamp': 0.000000000, 'duration': 7.024036281, 'label': 'N1', 'values': array([ 0.], dtype=float32)}
{'timestamp': 7.024036281, 'duration': 72.713287982, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 79.737324263, 'duration': 72.608798185, 'label': 'A', 'values': array([ 1.], dtype=float32)}
{'timestamp': 152.346122448, 'duration': 5.758548753, 'label': 'N2', 'values': array([ 0.], dtype=float32)}
{'timestamp': 158.104671201, 'duration': 11.644807257, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 169.749478458, 'duration': 11.644807256, 'label': 'C', 'values': array([ 3.], dtype=float32)}
{'timestamp': 181.394285714, 'duration': 11.807346939, 'label': 'N3', 'values': array([ 0.], dtype=float32)}
{'timestamp': 193.201632653, 'duration': 23.428934240, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 216.630566893, 'duration': 21.815147392, 'label': 'B', 'values': array([ 2.], dtype=float32)}
{'timestamp': 238.445714285, 'duration': 19.249342404, 'label': 'N5', 'values': array([ 0.], dtype=float32)}
```

b)

```
import vamp
import librosa
import mir_eval

ref_intervals, ref_labels = mir_eval.io.load_labeled_intervals('Label Tracka1.txt')
#print(ref_intervals)
(est_intervals, est_labels) = mir_eval.io.load_labeled_intervals('2_qm_labeltracka1.txt')
#print(est_intervals)
P0S, R0S, F0S = mir_eval.segment.detection(ref_intervals, est_intervals, window=0.5)
P3, R3, F3 = mir_eval.segment.detection(ref_intervals, est_intervals, window=3)
# Ignoring hits for the beginning and end of track
P, R, F = mir_eval.segment.detection(ref_intervals, est_intervals, window=0.5, trim=True)
(ref_intervals, ref_labels) = mir_eval.util.adjust_intervals(ref_intervals, ref_labels, t_min=0)
(est_intervals, est_labels) = mir_eval.util.adjust_intervals(est_intervals, est_labels, t_min=0, t_max=ref_intervals.max())
precision, recall, f = mir_eval.segment.pairwise(ref_intervals, ref_labels, est_intervals, est_labels, frame_size=0.1, beta=1.0)
print(f)
```

replace '2_qm_labeltracka1.txt' by '2_sg_labeltracka1.txt', '2_qm_labeltracka2.txt',
'2_sg_labeltracka2.txt'

combination of Queen Mary segmenter and my manual structure segmentation of a1.mp3:
0.419194327267

combination of Segmentino and my manual structure segmentation of a2.mp3:
0.532480826052

combination of Queen Mary segmenter and my manual structure segmentation of a2.mp3:
0.316064937146

combination of Segmentino and my manual structure segmentation of a2.mp3:
0.630008148739

The pairwise F-measure corresponds to my listening perception of segmentation quality. From Q1.3 I can hear that my segmentation is different from those two automatic segmentation, and in this question, the F-measure also present the difference.