**CPSC 501 Assignment 4 Report**

1. **Initial Base Program**

**Profiling-**

Flat profile:

Each sample counts as 0.01 seconds.

% cumulative self self total

time seconds seconds calls Ks/call Ks/call name

100.33 1784.78 1784.78 1 1.78 1.78 convolve(float\*, int, float\*, int, float\*, int)

0.00 1784.80 0.02 2 0.00 0.00 wavReader(float\*, char\*, int\*)

0.00 1784.82 0.02 1 0.00 0.00 scale(float\*, int)

0.00 1784.82 0.00 3 0.00 0.00 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode)

0.00 1784.82 0.00 2 0.00 0.00 std::fpos<\_\_mbstate\_t>::fpos(long)

0.00 1784.82 0.00 1 0.00 0.00 \_GLOBAL\_\_sub\_I\_chunkId

0.00 1784.82 0.00 1 0.00 0.00 \_\_static\_initialization\_and\_destruction\_0(int, int)

0.00 1784.82 0.00 1 0.00 0.00 wavWriter(float\*, int, char\*)

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) for 0.00% of 1784.82 seconds

index % time self children called name

<spontaneous>

[1] 100.0 0.00 1784.82 main [1]

1784.78 0.00 1/1 convolve(float\*, int, float\*, int, float\*, int) [2]

0.02 0.00 2/2 wavReader(float\*, char\*, int\*) [3]

0.02 0.00 1/1 scale(float\*, int) [4]

0.00 0.00 1/1 wavWriter(float\*, int, char\*) [15]

-----------------------------------------------

1784.78 0.00 1/1 main [1]

[2] 100.0 1784.78 0.00 1 convolve(float\*, int, float\*, int, float\*, int) [2]

-----------------------------------------------

0.02 0.00 2/2 main [1]

[3] 0.0 0.02 0.00 2 wavReader(float\*, char\*, int\*) [3]

0.00 0.00 2/3 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode) [11]

0.00 0.00 2/2 std::fpos<\_\_mbstate\_t>::fpos(long) [12]

-----------------------------------------------

0.02 0.00 1/1 main [1]

[4] 0.0 0.02 0.00 1 scale(float\*, int) [4]

-----------------------------------------------

0.00 0.00 1/3 wavWriter(float\*, int, char\*) [15]

0.00 0.00 2/3 wavReader(float\*, char\*, int\*) [3]

[11] 0.0 0.00 0.00 3 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode) [11]

-----------------------------------------------

0.00 0.00 2/2 wavReader(float\*, char\*, int\*) [3]

[12] 0.0 0.00 0.00 2 std::fpos<\_\_mbstate\_t>::fpos(long) [12]

-----------------------------------------------

0.00 0.00 1/1 \_\_libc\_csu\_init [21]

[13] 0.0 0.00 0.00 1 \_GLOBAL\_\_sub\_I\_chunkId [13]

0.00 0.00 1/1 \_\_static\_initialization\_and\_destruction\_0(int, int) [14]

-----------------------------------------------

0.00 0.00 1/1 \_GLOBAL\_\_sub\_I\_chunkId [13]

[14] 0.0 0.00 0.00 1 \_\_static\_initialization\_and\_destruction\_0(int, int) [14]

-----------------------------------------------

0.00 0.00 1/1 main [1]

[15] 0.0 0.00 0.00 1 wavWriter(float\*, int, char\*) [15]

0.00 0.00 1/3 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode) [11]

-----------------------------------------------

Copyright (C) 2012-2016 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,

are permitted in any medium without royalty provided the copyright

notice and this notice are preserved.

Index by function name

[13] \_GLOBAL\_\_sub\_I\_chunkId [2] convolve(float\*, int, float\*, int, float\*, int) [12] std::fpos<\_\_mbstate\_t>::fpos(long)

[14] \_\_static\_initialization\_and\_destruction\_0(int, int) [3] wavReader(float\*, char\*, int\*) [11] std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode)

[4] scale(float\*, int) [15] wavWriter(float\*, int, char\*)

**Summary –**

In this change, I implemented the standard DFT convolution. Because of how slow the algorithm is, we see that the total run time is 1784 seconds and 100% of our run time is spent inside of the convolve function.

1. **FFT Optimized Program**

**Profile –**

Flat profile:

Each sample counts as 0.01 seconds.

% cumulative self self total

time seconds seconds calls s/call s/call name

95.98 3.41 3.41 3 1.14 1.14 four1(float\*, int, int)

1.98 3.48 0.07 1 0.07 3.50 convolve(float\*, int, float\*, int, float\*, int)

0.85 3.51 0.03 2 0.02 0.02 wavReader(float\*, char\*, int\*)

0.56 3.53 0.02 1 0.02 0.02 four1Scale(float\*, int)

0.56 3.55 0.02 1 0.02 0.02 scale(float\*, int)

0.28 3.56 0.01 1 0.01 0.01 wavWriter(float\*, int, char\*)

0.00 3.56 0.00 3 0.00 0.00 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode)

0.00 3.56 0.00 2 0.00 0.00 std::fpos<\_\_mbstate\_t>::fpos(long)

0.00 3.56 0.00 1 0.00 0.00 \_GLOBAL\_\_sub\_I\_chunkId

0.00 3.56 0.00 1 0.00 0.00 \_\_static\_initialization\_and\_destruction\_0(int, int)

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) for 0.28% of 3.56 seconds

index % time self children called name

<spontaneous>

[1] 100.0 0.00 3.56 main [1]

0.07 3.43 1/1 convolve(float\*, int, float\*, int, float\*, int) [2]

0.03 0.00 2/2 wavReader(float\*, char\*, int\*) [4]

0.02 0.00 1/1 scale(float\*, int) [6]

0.01 0.00 1/1 wavWriter(float\*, int, char\*) [7]

-----------------------------------------------

0.07 3.43 1/1 main [1]

[2] 98.3 0.07 3.43 1 convolve(float\*, int, float\*, int, float\*, int) [2]

3.41 0.00 3/3 four1(float\*, int, int) [3]

0.02 0.00 1/1 four1Scale(float\*, int) [5]

-----------------------------------------------

3.41 0.00 3/3 convolve(float\*, int, float\*, int, float\*, int) [2]

[3] 95.8 3.41 0.00 3 four1(float\*, int, int) [3]

-----------------------------------------------

0.03 0.00 2/2 main [1]

[4] 0.8 0.03 0.00 2 wavReader(float\*, char\*, int\*) [4]

0.00 0.00 2/3 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode) [14]

0.00 0.00 2/2 std::fpos<\_\_mbstate\_t>::fpos(long) [15]

-----------------------------------------------

0.02 0.00 1/1 convolve(float\*, int, float\*, int, float\*, int) [2]

[5] 0.6 0.02 0.00 1 four1Scale(float\*, int) [5]

-----------------------------------------------

0.02 0.00 1/1 main [1]

[6] 0.6 0.02 0.00 1 scale(float\*, int) [6]

-----------------------------------------------

0.01 0.00 1/1 main [1]

[7] 0.3 0.01 0.00 1 wavWriter(float\*, int, char\*) [7]

0.00 0.00 1/3 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode) [14]

-----------------------------------------------

0.00 0.00 1/3 wavWriter(float\*, int, char\*) [7]

0.00 0.00 2/3 wavReader(float\*, char\*, int\*) [4]

[14] 0.0 0.00 0.00 3 std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode) [14]

-----------------------------------------------

0.00 0.00 2/2 wavReader(float\*, char\*, int\*) [4]

[15] 0.0 0.00 0.00 2 std::fpos<\_\_mbstate\_t>::fpos(long) [15]

-----------------------------------------------

0.00 0.00 1/1 \_\_libc\_csu\_init [23]

[16] 0.0 0.00 0.00 1 \_GLOBAL\_\_sub\_I\_chunkId [16]

0.00 0.00 1/1 \_\_static\_initialization\_and\_destruction\_0(int, int) [17]

-----------------------------------------------

0.00 0.00 1/1 \_GLOBAL\_\_sub\_I\_chunkId [16]

[17] 0.0 0.00 0.00 1 \_\_static\_initialization\_and\_destruction\_0(int, int) [17]

-----------------------------------------------

Copyright (C) 2012-2016 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,

are permitted in any medium without royalty provided the copyright

notice and this notice are preserved.

Index by function name

[16] \_GLOBAL\_\_sub\_I\_chunkId [6] scale(float\*, int) [15] std::fpos<\_\_mbstate\_t>::fpos(long)

[5] four1Scale(float\*, int) [2] convolve(float\*, int, float\*, int, float\*, int) [14] std::operator|(std::\_Ios\_Openmode, std::\_Ios\_Openmode)

[17] \_\_static\_initialization\_and\_destruction\_0(int, int) [4] wavReader(float\*, char\*, int\*)

[3] four1(float\*, int, int) [7] wavWriter(float\*, int, char\*)

**Regression Testing –**

**Summary -**

In this change, I implemented the FFT solution. Because of how much faster that algorithm is, my runtime has been improved to 3.56 seconds. Since the FFT is so much faster, we spent much less time in the convolution stage compared to last time, therefore we can see that 0.28% of our time is spent inside of the wavWriter function.

1. **FFT Optimized Program + Hand Tuning**

**Profile –**

**Regression Testing –**

**Changes –**

1. **Unroll one of the loops in Convolve**

**while (i < newArrSize){**

**newInput[i] = 0;**

**i++;**

**}**

**And**

**while (i < newArrSize){**

**newIR[i] = 0;**

**i++;**

**}**

**Become**

**while (i < newArrSize){**

**newInput[i] = 0;**

**newInput[i+1] = 0;**

**i+=2;**

**}**

**And**

**while (i < newArrSize){**

**newIR[i] = 0;**

**newIR[i+1] = 0;**

**i+=2;**

**}**

1. **Remove power operation**

short sam;

sig = new float[wavSize];

for (int i = 0; i < wavSize; i++){

sam = wavData[i];

sig[i] = (sam\*1.0) / (pow(2.0, 15.0) -1);

if (sig[i] < -1.0){

sig[i] = -1.0;

}

}

**BECOMES**

short sam;

sig = new float[wavSize];

for (int i = 0; i < wavSize; i++){

sam = wavData[i];

sig[i] = (sam\*1.0) / (32767);

if (sig[i] < -1.0){

sig[i] = -1.0;

}

}

1. **ROLL Out four1Scale even further**

**for (i = 0; i < newArrSize; i++) {**

**newOutput[i] = 0;**

**}**

**Becomes**

**for (i = 0; i < newArrSize; i+=2) {**

**newOutput[i] = 0;**

**newOutput[i+1] = 0;**

**}**

// From notes in class

void four1Scale (float signal[], int N)

{

int i;

int j;

for (i = 0, j = 0; i < N; i++, j+=2) {

signal[j] /= (float)N;

signal[j+1] /= (float)N;

}

}

**Becomes**

// From notes in class

void four1Scale (float signal[], int N)

{

int i;

int j;

for (i = 0, j = 0; i < N; i++, j+=4) {

signal[j] /= (float)N;

signal[j+1] /= (float)N;

signal[j+2] /= (float)N;

signal[j+3] /= (float)N;

}

}

**Summary –**