

NAME : SHAYNE MUJURU

UCID : 30029552

**TITLE : CPSC 501 AIV : CODE TUNING, CONVOLUTION, DFT/FFT AND AUDIO
PROCESSING**

BASE CONVOLUTION TIMING

Input file : GuitarDry.wav

IR file : Taj.wav

Output File : TajGuitarDry.wav

```
Writing convolved signal to file: TajGuitarDry.wav
Writing successful

real    28m26.319s
user    28m22.409s
sys     0m0.216s
```

ALGORITHMIC OPTIMIZATION

Input file : GuitarDry.wav

IR file : Taj.wav

Output File : TajGuitarDryFFT.wav

```
Writing convolved signal to file: TajGuitarDryFFT.wav
Writing successful

real    0m8.726s
user    0m8.564s
sys     0m0.093s
[shayne.mujuru@csx3 a4]$
```

MANUAL CODE TUNINGS

```
for(int i = 0; i < maxLengthPow2; i++){
    freqX[i] = 0.0;
    freqH[i] = 0.0;
}
```

CODE JAMMING : initially freqX and freqH were computed in different loops, made more efficient by incorporating into a single loop. Lead to a increase in performance speed.

```
//find next largest power of 2 that's at least as large as maxLength
int pow2 = 1;
while(pow2 < maxLength){
    pow2 *= 2;
}

//double that power of 2 for length of freq arrays X[k],H[k]
//i.e. make sure freq arrays long enough for real and imaginary parts
int maxLengthPow2 = pow2*2;
```

AFTER:

```
int pow2 = 1;
while (pow2 < maxLength) {
    pow2 = pow2 << 1;
}

int maxLengthtoPow2 = pow2 << 1;
```

STRENGTH REDUCTION : using a bitwise left shift to replace the expensive operation multiplication

MINIMIZE ARRAY REFERENCES & REDUCING AMOUNT OF WORK DONE IN LOOPS

```
y[i] = (x[i] * h[i]) - x[i+1] * h[i+1];

//imaginary values
y[i+1] = x[i+1] * h[i] + x[i] * h[i+1];

if((i%100000) == 0)
    printf("Convolving %d...\n", i);
```

Loop had additional work to print at each stage that it was convolving and would try to access info in arrays by direct referencing.

```
for(int i = 0; i < P; i+= 2) {  
  
    //real values  
    double realValX = x[i];  
    double realValH = h[i];  
  
    //imaginary  
    double imaginaryX = x[i+1];  
    double imaginaryH = h[i+1];  
  
    y[i] = (realValX * realValH) - (imaginaryX * imaginaryH);  
  
    //imaginary values  
    y[i+1] = imaginaryX * realValH + realValX * imaginaryH;  
    Shayne Mujuru [58 minutes ago] • Algorithmic Optimization Implemented  
}  
}
```

-