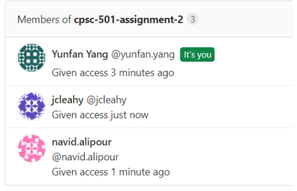
Yunfan Yang 30067857

https://gitlab.cpsc.ucalgary.ca/yunfan.yang/cpsc-501-assignment-2/

The access to this repository has been granted to the TA and the professor.



Baseline Program

Commit: 8f6c17866033a1326ec17acbe77a0a517f328efa

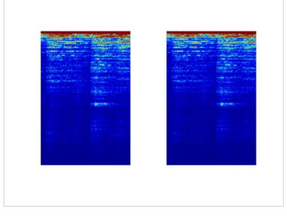
This is the baseline version of the program. It implements: read and write wave file, convolution with O(n^2) time complexity multiplication algorithm.

The full audio takes a really long time to convolute.

Profiling

```
guitar_dry.wav
  %   cumulative   self              self     total
 time   seconds   seconds    calls   s/call   s/call  name
100.00    407.87   407.87        1   407.87   407.87  convolution(WaveFile, WaveFile)
  0.00    407.88     0.01        1     0.01     0.01  WaveFile::writeData()
  0.00    407.88     0.00       12     0.00     0.00  WaveFile::nextIntLSB()
  0.00    407.88     0.00        8     0.00     0.00  WaveFile::nextShortLSB()
  0.00    407.88     0.00        5     0.00     0.00  WaveFile::nextIntLSB(int)
  0.00    407.88     0.00        5     0.00     0.00  WaveFile::~WaveFile()
  0.00    407.88     0.00        4     0.00     0.00  WaveFile::nextShortLSB(short)
  0.00    407.88     0.00        3     0.00     0.00  WaveFile::WaveFile()
  0.00    407.88     0.00        2     0.00     0.00  WaveFile::readHeader()
  0.00    407.88     0.00        2     0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00    407.88     0.00        2     0.00     0.00  WaveFile::readData()
  0.00    407.88     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
  0.00    407.88     0.00        1     0.00     0.00  WaveFile::writeHeader()
  0.00    407.88     0.00        1     0.00     0.01  WaveFile::write(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
```

The similarity between these two audio files is 99.13%.



This is a comparison graph of the provided sample output with the output from my program. The files are almost identical.

Algorithmic Optimization: FFT Program

Commit: c9fc0d2c8a3854b44c11a52c5684a7c9c77c09c6

This is the optimized version of convolution algorithm, using Fast Fourier Transform to achieve O(n log n) time complexity.

The code is partially referenced from: Fast Fourier transform - Rosetta Code.

It implements: find the next closest n^2 number, fft and ifft, updated convolution function adapted fft and ifft.

The time is reduced from 407 seconds to 1.92 seconds, which is really significant.

Profiling

```
guitar_dry.wav
  %   cumulative   self              self     total
 time   seconds   seconds    calls  s/call   s/call  name
 33.78     2.00     2.00        3     0.67     1.92  fft(std::valarray<std::complex<double> >&)
  8.78     2.52     0.52 134217728     0.00     0.00  std::complex<double>& std::complex<double>::operator*=<double>(std::complex<double> const&)
  8.45     3.02     0.50 136314884     0.00     0.00  std::complex<double>::complex(double, double)
  7.60     3.47     0.45 132120576     0.00     0.00  std::complex<double> std::polar<double>(double const&, double const&)
  5.24     3.78     0.31 274505830     0.00     0.00  std::complex<double>::real[abi:cxx11]() const
  4.56     4.05     0.27 12582906      0.00     0.00  void std::__valarray_copy_construct<std::complex<double> >(std::complex<double> const*, unsigned int, unsigned int, std::complex<double>*)
  4.05     4.29     0.24 400113869     0.00     0.00  std::valarray<std::complex<double> >::operator[](unsigned int)
  3.89     4.52     0.23 272629760     0.00     0.00  std::complex<double>::imag[abi:cxx11]() const
  3.38     4.72     0.20 66060288      0.00     0.00  std::complex<double> std::operator-<double>(std::complex<double> const&, std::complex<double> const&)
  2.87     4.89     0.17 66060288      0.00     0.00  std::complex<double> std::operator+<double>(std::complex<double> const&, std::complex<double> const&)
  2.70     5.05     0.16 66060288      0.00     0.00  std::complex<double>& std::complex<double>::operator-=<double>(std::complex<double> const&)
  2.53     5.20     0.15 134217728     0.00     0.00  std::complex<double> std::operator*<double>(std::complex<double> const&, std::complex<double> const&)
  1.86     5.31     0.11 12582912      0.00     0.00  void std::__valarray_destroy_elements<std::complex<double> >(std::complex<double>*, std::complex<double>*)
  1.52     5.40     0.09 66060288      0.00     0.00  std::complex<double>& std::complex<double>::operator+=<double>(std::complex<double> const&)
  1.18     5.47     0.07 37748722      0.00     0.00  std::_Array<std::complex<double> >::_Array(std::complex<double>*)
  1.01     5.53     0.06 138412032     0.00     0.00  operator new(unsigned int, void*)
  0.84     5.58     0.05 12582906      0.00     0.00  std::slice_array<std::complex<double> >::slice_array(std::_Array<std::complex<double> >, std::slice const&)
  0.84     5.63     0.05 12582906      0.00     0.00  std::valarray<std::complex<double> >::operator[](std::slice)
  0.51     5.66     0.03 12582913      0.00     0.00  std::valarray<std::complex<double> >::size() const
  0.51     5.69     0.03 12582906      0.00     0.00  std::slice::start() const
  0.34     5.71     0.02 12582912      0.00     0.00  std::__valarray_release_memory(void*)
  0.34     5.73     0.02 12582909      0.00     0.00  std::valarray<std::complex<double> >::~valarray()
  0.34     5.75     0.02 12582909      0.00     0.00  std::complex<double>* restrict std::__valarray_get_storage<std::complex<double> >(unsigned int)
  0.34     5.77     0.02 12582906      0.00     0.00  std::slice::size() const
  0.34     5.79     0.02 12582906      0.00     0.00  std::slice::slice(unsigned int, unsigned int, unsigned int)
  0.34     5.81     0.02 12582906      0.00     0.00  std::valarray<std::complex<double> >::valarray(std::slice_array<std::complex<double> > const&)
  0.34     5.83     0.02  2097152      0.00     0.00  std::complex<double>& std::complex<double>::operator/=<double>(std::complex<double> const&)
  0.34     5.85     0.02        3     0.01     0.01  std::_Array_init_ctor<std::complex<double>, false>::_S_do_it(std::complex<double>*, std::complex<double>)
  0.34     5.87     0.02        2     0.01     0.02  void std::__valarray_copy<std::complex<double>, std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&, unsigned int, std::_Array<std::complex<double> >)
  0.17     5.88     0.01  1876071     0.00     0.00  std::complex<double>::operator=(double)
  0.17     5.89     0.01        1     0.01     0.01  WaveFile::writeData()
  0.17     5.90     0.01        1     0.01     0.02  void std::__valarray_copy<std::complex<double>, std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> > >(std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> > const&, unsigned int,
std::_Array<std::complex<double> >)
  0.17     5.91     0.01                             operator delete(void*)
  0.17     5.92     0.01                             cos
  0.00     5.92     0.00 12582909      0.00     0.00  std::__valarray_get_memory(unsigned int)
  0.00     5.92     0.00 12582906      0.00     0.00  std::slice::stride() const
  0.00     5.92     0.00 12582906      0.00     0.00  std::_Array<std::complex<double> >::begin() const
  0.00     5.92     0.00 12582906      0.00     0.00  void std::__valarray_copy_construct<std::complex<double> >(std::_Array<std::complex<double> >, unsigned int, unsigned int, std::_Array<std::complex<double> >)
  0.00     5.92     0.00  8388608      0.00     0.00  std::valarray<std::complex<double> >::operator[](unsigned int) const
  0.00     5.92     0.00  4194304      0.00     0.00  std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> >::operator[](unsigned int) const
  0.00     5.92     0.00  4194304      0.00     0.00  std::_FunBase<std::valarray<std::complex<double> >, std::complex<double> const&>::operator[](unsigned int) const
  0.00     5.92     0.00  4194304      0.00     0.00  std::complex<double> std::conj<double>(std::complex<double> const&)
  0.00     5.92     0.00  2097152      0.00     0.00  std::complex<double> std::__multiplies::operator()<std::complex<double> >(std::complex<double> const&, std::complex<double> const&) const
  0.00     5.92     0.00  2097152      0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> >::operator[](unsigned int) const
  0.00     5.92     0.00  2097152      0.00     0.00  std::_BinBase<std::__multiplies, std::valarray<std::complex<double> >, std::valarray<std::complex<double> > >::operator[](unsigned int) const
  0.00     5.92     0.00       12      0.00     0.00  WaveFile::nextIntLSB()
  0.00     5.92     0.00        8      0.00     0.00  WaveFile::nextShortLSB()
  0.00     5.92     0.00        5      0.00     0.00  WaveFile::nextIntLSB(int)
  0.00     5.92     0.00        5      0.00     0.00  WaveFile::~WaveFile()
  0.00     5.92     0.00        4      0.00     0.00  WaveFile::nextShortLSB(short)
  0.00     5.92     0.00        3      0.00     0.00  WaveFile::WaveFile()
  0.00     5.92     0.00        3      0.00     0.01  std::valarray<std::complex<double> >::resize(unsigned int, std::complex<double>)
  0.00     5.92     0.00        3      0.00     0.00  std::valarray<std::complex<double> >::valarray()
  0.00     5.92     0.00        3      0.00     0.01  void std::__valarray_fill_construct<std::complex<double> >(std::complex<double>*, std::complex<double>*, std::complex<double>)
  0.00     5.92     0.00        2      0.00     0.00  WaveFile::readHeader()
  0.00     5.92     0.00        2      0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00     5.92     0.00        2      0.00     0.00  WaveFile::readData()
```

```
 0.00      5.92     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
 0.00      5.92     0.00        2     0.00     0.00  std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> >::size() const
 0.00      5.92     0.00        2     0.00     0.00  std::_FunBase<std::valarray<std::complex<double> >, std::complex<double> const&>::size() const
 0.00      5.92     0.00        2     0.00     0.00  std::valarray<std::complex<double> >::apply(std::complex<double> (*)(std::complex<double> const&)) const
 0.00      5.92     0.00        2     0.00     0.00  std::_RefFunClos<std::_ValArray, std::complex<double> >::_RefFunClos(std::valarray<std::complex<double> > const&, std::complex<double> (*)(std::complex<double> const&))
 0.00      5.92     0.00        2     0.00     0.00  std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> >::_Expr(std::_RefFunClos<std::_ValArray, std::complex<double> > const&)
 0.00      5.92     0.00        2     0.00     0.00  std::_FunBase<std::valarray<std::complex<double> >, std::complex<double> const&>::_FunBase(std::valarray<std::complex<double> > const&, std::complex<double> (*)(std::complex<double> const&))
 0.00      5.92     0.00        2     0.00     0.02  std::valarray<std::complex<double> >& std::valarray<std::complex<double> >::operator=<std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&)
 0.00      5.92     0.00        1     0.00     0.00  convolution(WaveFile, WaveFile)
 0.00      5.92     0.00        1     0.00     0.00  upper_power_of_two(unsigned long)
 0.00      5.92     0.00        1     0.00     1.99  ifft(std::valarray<std::complex<double> >&)
 0.00      5.92     0.00        1     0.00     0.00  WaveFile::writeHeader()
 0.00      5.92     0.00        1     0.00     0.01  WaveFile::write(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
 0.00      5.92     0.00        1     0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> >::size() const
 0.00      5.92     0.00        1     0.00     0.00  std::_BinBase<std::__multiplies, std::valarray<std::complex<double> >, std::valarray<std::complex<double> > >::size() const
 0.00      5.92     0.00        1     0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> >::_Expr(std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> > const&)
 0.00      5.92     0.00        1     0.00     0.00  std::_BinBase<std::__multiplies, std::valarray<std::complex<double> >, std::valarray<std::complex<double> > >::_BinBase(std::valarray<std::complex<double> > const&, std::valarray<std::complex<double> > const&)
 0.00      5.92     0.00        1     0.00     0.00  std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >::_BinClos(std::valarray<std::complex<double> > const&, std::valarray<std::complex<double> > const&)
 0.00      5.92     0.00        1     0.00     0.02  std::valarray<std::complex<double> >& std::valarray<std::complex<double> >::operator=<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> > >(std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> > const&)
 0.00      5.92     0.00        1     0.00     0.02  std::valarray<std::complex<double> >::operator/=(std::complex<double> const&)
 0.00      5.92     0.00        1     0.00     0.02  void std::_Array_augmented__divides<std::complex<double> >(std::_Array<std::complex<double> >, unsigned int, std::complex<double> const&)
 0.00      5.92     0.00        1     0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::__fun<std::__multiplies, std::complex<double> >::result_type> std::operator*<std::complex<double> >(std::valarray<std::complex<double> > const&, std::valarray<std::complex<double> > const&)
```

As one can see, the function takes most of the time is fft, thus the following optimizations are focusing on reducing the time of this function.

Regression Testing

```
cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$ cmp output_dry.wav output_dry_base.wav

cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$
```

To testing the correctness, compare the new output file with the original base version output file.

There is nothing prints after the compare command, which indicates that the two files are identical.

Optimization: Compiler

Commit: 0678ecb72ed3a312b64431627424332c50eeace9

In the vscode build task configuration file, the optimization tag -O2 is added to compile. Then run the build task and run the program.

```
5   5            "label": "C/C++: g++.exe build active file",
6   6            "command": "C:\\MinGW\\bin\\g++.exe",
7   7            "args": [
8   +                "-O2",
8   9                "-fdiagnostics-color=always",
9   10               "-pg",
10  11               "-g",
```

Profiling

```
  %   cumulative   self              self     total
 time   seconds   seconds    calls  s/call   s/call  name
 33.05      1.94     1.94        3     0.65     1.90  fft(std::valarray<std::complex<double> >&)
  9.88      2.52     0.58 136314884     0.00     0.00  std::complex<double>::complex(double, double)
  8.69      3.03     0.51 132120576     0.00     0.00  std::complex<double> std::polar<double>(double const&, double const&)
  8.52      3.53     0.50 274505830     0.00     0.00  std::complex<double>::real[abi:cxx11]() const
  6.47      3.91     0.38 134217728     0.00     0.00  std::complex<double>& std::complex<double>::operator*=<double>(std::complex<double> const&)
  5.62      4.24     0.33 400113869     0.00     0.00  std::valarray<std::complex<double> >::operator[](unsigned int)
  4.26      4.49     0.25 12582906      0.00     0.00  void std::__valarray_copy_construct<std::complex<double> >(std::complex<double> const*, unsigned int, unsigned int, std::complex<double>*)
  3.75      4.71     0.22 272629760     0.00     0.00  std::complex<double>::imag[abi:cxx11]() const
  2.90      4.88     0.17 12582912      0.00     0.00  void std::__valarray_destroy_elements<std::complex<double> >(std::complex<double>*, std::complex<double>*)
  2.73      5.04     0.16 66060288      0.00     0.00  std::complex<double> std::operator+<double>(std::complex<double> const&, std::complex<double> const&)
  2.39      5.18     0.14 66060288      0.00     0.00  std::complex<double>& std::complex<double>::operator+=<double>(std::complex<double> const&)
  2.21      5.31     0.13 134217728     0.00     0.00  std::complex<double> std::operator*<double>(std::complex<double> const&, std::complex<double> const&)
  1.87      5.42     0.11 66060288      0.00     0.00  std::complex<double>& std::complex<double>::operator-=<double>(std::complex<double> const&)
  0.85      5.47     0.05 138412032     0.00     0.00  operator new(unsigned int, void*)
  0.85      5.52     0.05 66060288      0.00     0.00  std::complex<double> std::operator-<double>(std::complex<double> const&, std::complex<double> const&)
  0.85      5.57     0.05 12582906      0.00     0.00  std::valarray<std::complex<double> >::valarray(std::slice_array<std::complex<double> > const&)
  0.68      5.61     0.04 12582909      0.00     0.00  std::__valarray_get_memory(unsigned int)
  0.51      5.64     0.03 12582906      0.00     0.00  std::slice::slice(unsigned int, unsigned int, unsigned int)
  0.51      5.67     0.03                             sin
  0.34      5.69     0.02 37748722      0.00     0.00  std::_Array<std::complex<double> >::_Array(std::complex<double>*)
  0.34      5.71     0.02 12582909      0.00     0.00  std::valarray<std::complex<double> >::~valarray()
  0.34      5.73     0.02 12582906      0.00     0.00  std::slice::size() const
  0.34      5.75     0.02 12582906      0.00     0.00  std::slice_array<std::complex<double> >::slice_array(std::_Array<std::complex<double> >, std::slice const&)
  0.34      5.77     0.02 12582906      0.00     0.00  void std::__valarray_copy_construct<std::complex<double> >(std::_Array<std::complex<double> >, unsigned int, unsigned int, std::_Array<std::complex<double> >)
  0.34      5.79     0.02  2097152      0.00     0.00  std::complex<double>& std::complex<double>::operator/=<double>(std::complex<double> const&)
  0.17      5.80     0.01 12582912      0.00     0.00  std::__valarray_release_memory(void*)
  0.17      5.81     0.01 12582909      0.00     0.00  std::complex<double>* restrict std::__valarray_get_storage<std::complex<double> >(unsigned int)
  0.17      5.82     0.01 12582906      0.00     0.00  std::_Array<std::complex<double> >::begin() const
  0.17      5.83     0.01  4194304      0.00     0.00  std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> >::operator[](unsigned int) const
  0.17      5.84     0.01  2097152      0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> >::operator[](unsigned int) const
  0.17      5.85     0.01        1     0.01     0.01  WaveFile::writeData()
  0.17      5.86     0.01                             operator delete(void*)
  0.17      5.87     0.01                             cos
  0.00      5.87     0.00 12582913      0.00     0.00  std::valarray<std::complex<double> >::size() const
  0.00      5.87     0.00 12582906      0.00     0.00  std::slice::start() const
  0.00      5.87     0.00 12582906      0.00     0.00  std::slice::stride() const
  0.00      5.87     0.00 12582906      0.00     0.00  std::valarray<std::complex<double> >::operator[](std::slice)
  0.00      5.87     0.00  8388608      0.00     0.00  std::valarray<std::complex<double> >::operator[](unsigned int) const
  0.00      5.87     0.00  4194304      0.00     0.00  std::_FunBase<std::valarray<std::complex<double> >, std::complex<double> const&>::operator[](unsigned int) const
  0.00      5.87     0.00  4194304      0.00     0.00  std::complex<double> std::conj<double>(std::complex<double> const&)
  0.00      5.87     0.00  2097152      0.00     0.00  std::complex<double> std::__multiplies::operator()<std::complex<double> >(std::complex<double> const&, std::complex<double> const&) const
  0.00      5.87     0.00  2097152      0.00     0.00  std::_BinBase<std::__multiplies, std::valarray<std::complex<double> >, std::valarray<std::complex<double> > >::operator[](unsigned int) const
  0.00      5.87     0.00  1876071      0.00     0.00  std::complex<double>::operator=(double)
  0.00      5.87     0.00       12     0.00     0.00  WaveFile::nextIntLSB()
  0.00      5.87     0.00        8     0.00     0.00  WaveFile::nextShortLSB()
  0.00      5.87     0.00        5     0.00     0.00  WaveFile::nextIntLSB(int)
  0.00      5.87     0.00        5     0.00     0.00  WaveFile::~WaveFile()
  0.00      5.87     0.00        4     0.00     0.00  WaveFile::nextShortLSB(short)
  0.00      5.87     0.00        3     0.00     0.00  WaveFile::WaveFile()
  0.00      5.87     0.00        3     0.00     0.00  std::_Array_init_ctor<std::complex<double>, false>::_S_do_it(std::complex<double>*, std::complex<double>*, std::complex<double>)
  0.00      5.87     0.00        3     0.00     0.00  std::valarray<std::complex<double> >::resize(unsigned int, std::complex<double>)
  0.00      5.87     0.00        3     0.00     0.00  std::valarray<std::complex<double> >::valarray()
  0.00      5.87     0.00        3     0.00     0.00  void std::__valarray_fill_construct<std::complex<double> >(std::complex<double>*, std::complex<double>*, std::complex<double>)
  0.00      5.87     0.00        2     0.00     0.00  WaveFile::readHeader()
  0.00      5.87     0.00        2     0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00      5.87     0.00        2     0.00     0.00  WaveFile::readData()
```

```
  0.00      5.87     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
  0.00      5.87     0.00        2     0.00     0.00  std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> >::size() const
  0.00      5.87     0.00        2     0.00     0.00  std::_FunBase<std::valarray<std::complex<double> >, std::complex<double> const&>::size() const
  0.00      5.87     0.00        2     0.00     0.00  std::valarray<std::complex<double> >::apply(std::complex<double> (*)(std::complex<double> const&)) const
  0.00      5.87     0.00        2     0.00     0.00  std::_RefFunClos<std::_ValArray, std::complex<double> >::_RefFunClos(std::valarray<std::complex<double> > const&, std::complex<double> (*)(std::complex<double> const&))
  0.00      5.87     0.00        2     0.00     0.00  std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> >::_Expr(std::_RefFunClos<std::_ValArray, std::complex<double> > const&)
  0.00      5.87     0.00        2     0.00     0.00  std::_FunBase<std::valarray<std::complex<double> >, std::complex<double> const&>::_FunBase(std::valarray<std::complex<double> > const&, std::complex<double> (*)(std::complex<double> const&))
  0.00      5.87     0.00        2     0.00     0.02  std::valarray<std::complex<double> >& std::valarray<std::complex<double> >::operator=<std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&)
  0.00      5.87     0.00        2     0.00     0.02  void std::__valarray_copy<std::complex<double>, std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&, unsigned int, std::_Array<std::complex<double> >)
  0.00      5.87     0.00        1     0.00     0.00  convolution(WaveFile, WaveFile)
  0.00      5.87     0.00        1     0.00     0.00  upper_power_of_two(unsigned long)
  0.00      5.87     0.00        1     0.00     1.97  ifft(std::valarray<std::complex<double> >&)
  0.00      5.87     0.00        1     0.00     0.00  WaveFile::writeHeader()
  0.00      5.87     0.00        1     0.00     0.01  WaveFile::write(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00      5.87     0.00        1     0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> >::size() const
  0.00      5.87     0.00        1     0.00     0.00  std::_BinBase<std::__multiplies, std::valarray<std::complex<double> >, std::valarray<std::complex<double> > >::size() const
  0.00      5.87     0.00        1     0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> >::_Expr(std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> > const&)
  0.00      5.87     0.00        1     0.00     0.00  std::_BinBase<std::__multiplies, std::valarray<std::complex<double> >, std::valarray<std::complex<double> > >::_BinBase(std::valarray<std::complex<double> > const&, std::valarray<std::complex<double> > const&)
  0.00      5.87     0.00        1     0.00     0.00  std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >::_BinClos(std::valarray<std::complex<double> > const&, std::valarray<std::complex<double> > const&)
  0.00      5.87     0.00        1     0.00     0.02  std::valarray<std::complex<double> >& std::valarray<std::complex<double> >::operator=<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> > >(std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> > const&)
  0.00      5.87     0.00        1     0.00     0.03  std::valarray<std::complex<double> >::operator/=(std::complex<double> const&)
  0.00      5.87     0.00        1     0.00     0.02  void std::__valarray_copy<std::complex<double>, std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> > >(std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::complex<double> > const&, unsigned int,
std::_Array<std::complex<double> >)
  0.00      5.87     0.00        1     0.00     0.03  void std::_Array_augmented___divides<std::complex<double> >(std::_Array<std::complex<double> >, unsigned int, std::complex<double> const&)
  0.00      5.87     0.00        1     0.00     0.00  std::_Expr<std::_BinClos<std::__multiplies, std::_ValArray, std::_ValArray, std::complex<double>, std::complex<double> >, std::__fun<std::__multiplies, std::complex<double> >::result_type> std::operator*<std::complex<double> >(std::valarray<std::complex<double> > const&, std::valarray<std::complex<double> > const&)
```

Regression Testing

```
cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$ cmp output_dry.wav output_dry_base.wav

cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$
```

To testing the correctness, compare the new output file with the original base version output file.

There is nothing prints after the compare command, which indicates that the two files are identical.

Optimization: Jamming

Commit: aa4af5687890fd17b5a094a3d7ab075670878715

This optimization combines the two for-loops from the fft function into one.

```
260 260     for (size_t k = 0; k < n / 2; ++k) {
261 261         Complex t = polar(1.0, -2 * PI * k / n) * odd[k];
262 262         x[k] = even[k] + t;
263     -   }
264     -
265     -   // for (size_t
266     -   for (size_t k = 0; k < n / 2; ++k) {
267     -       Complex t = polar(1.0, -2 * PI * k / n) * odd[k];
268 263         x[k + n / 2] = even[k] - t;
269 264     }
270 265 }
```

Now, the time has reduced from 1.9 seconds to 233.33 miliseconds, which is also a significant improvement.

Profiling

```
  %   cumulative   self              self     total
 time   seconds   seconds    calls  ms/call  ms/call  name
 88.61      0.70     0.70        3   233.33   233.33  fft(std::valarray<std::complex<double> >&)
  5.06      0.74     0.04                             __muldc3
  1.27      0.75     0.01  4194304     0.00     0.00  std::complex<double> std::conj<double>(std::complex<double> const&)
  1.27      0.76     0.01                             _fu1___ZSt4cout
  1.27      0.77     0.01                             _fu21___ZSt4cout
  1.27      0.78     0.01                             _fu45___ZSt4cout
  1.27      0.79     0.01                             _fu9___ZSt4cout
  0.00      0.79     0.00       11     0.00     0.00  __gcc_deregister_frame
  0.00      0.79     0.00        2     0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00      0.79     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
  0.00      0.79     0.00        2     0.00     5.00  void std::__valarray_copy<std::complex<double>, std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&, unsigned int, std::_Array<std::complex<double> >)
  0.00      0.79     0.00        1     0.00     0.00  convolution(WaveFile, WaveFile)
  0.00      0.79     0.00        1     0.00   243.33  ifft(std::valarray<std::complex<double> >&)
```

Regression Testing

```
cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$ cmp output_dry.wav output_dry_base.wav

cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$
```

To testing the correctness, compare the new output file with the original base version output file.

There is nothing prints after the compare command, which indicates that the two files are identical.

Optimization: Minimize Array Reference

Commit: f9735ee1628c4b48bd3e4da1237dc2ea05b226a9

This optimization replace all the even[k] reference with a variable, for fft function. Thus, the access to even[k] would only be once.

```
259 259     // combine
260 260     for (size_t k = 0; k < n / 2; ++k) {
261 261         Complex t = polar(1.0, -2 * PI * k / n) * odd[k];
262         -       x[k] = even[k] + t;
263         -       x[k + n / 2] = even[k] - t;
    262     +       Complex even_k = even[k];
    263     +
    264     +       x[k] = even_k + t;
    265     +       x[k + n / 2] = even_k - t;
264 266     }
265 267 }
266 268
```

Profiling

```
  %   cumulative   self              self     total
 time   seconds   seconds    calls  ms/call  ms/call  name
 95.16      0.59     0.59        3   196.67   196.67  fft(std::valarray<std::complex<doubsle> >&)
  1.61      0.60     0.01                             _fu1___ZSt4cout
  1.61      0.61     0.01                             _fu21___ZSt4cout
  1.61      0.62     0.01                             _fu45___ZSt4cout
  0.00      0.62     0.00  4194304     0.00     0.00  std::complex<double> std::conj<double>(std::complex<double> const&)
  0.00      0.62     0.00       11     0.00     0.00  __gcc_deregister_frame
  0.00      0.62     0.00        2     0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00      0.62     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
  0.00      0.62     0.00        2     0.00     0.00  void std::__valarray_copy<std::complex<double>, std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&, unsigned int, std::_Array<std::complex<double> >)
  0.00      0.62     0.00        1     0.00     0.00  convolution(WaveFile, WaveFile)
  0.00      0.62     0.00        1     0.00   196.67  ifft(std::valarray<std::complex<double> >&)
```

Regression Testing

```
cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$ cmp output_dry.wav output_dry_base.wav

cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$
```

To testing the correctness, compare the new output file with the original base version output file.

There is nothing prints after the compare command, which indicates that the two files are identical.

Optimization: Minimize work

Commit: 6116b4d04ffb1d0cc66d1a1ce48df70e0cb4544d

This optimization replaces all n/2 with a variable, for fft function, so the program would not need to do division every time.

```
248 248     const size_t n = x.size();
249 249     if (n <= 1) return;
250 250
    251 +   int n_over_2 = int(n / 2);
    252 +
251 253     // divide
252     -   ComplexArray even = x[slice(0, n / 2, 2)];
253     -   ComplexArray odd = x[slice(1, n / 2, 2)];
    254 +   ComplexArray even = x[slice(0, n_over_2, 2)];
    255 +   ComplexArray odd = x[slice(1, n_over_2, 2)];
254 256
255 257     // conquer
256 258     fft(even);
257 259     fft(odd);
258 260
259 261     // combine
260     -   for (size_t k = 0; k < n / 2; ++k) {
    262 +   for (size_t k = 0; k < n_over_2; ++k) {
261 263         Complex t = polar(1.0, -2 * PI * k / n) * odd[k];
262 264         Complex even_k = even[k];
263 265
264 266         x[k] = even_k + t;
265     -       x[k + n / 2] = even_k - t;
    267 +       x[k + n_over_2] = even_k - t;
266 268     }
267 269 }
268 270
```

Profiling

```
  %   cumulative   self              self     total
 time   seconds   seconds    calls  ms/call  ms/call  name
 90.38      0.47     0.47        3   156.67   156.67  fft(std::valarray<std::complex<double> >&)
  1.92      0.48     0.01  4194304     0.00     0.00  std::complex<double> std::conj<double>(std::complex<double> const&)
  1.92      0.49     0.01                             _fu1___ZSt4cout
  1.92      0.50     0.01                             _fu45___ZSt4cout
  1.92      0.51     0.01                             _fu9___ZSt4cout
  1.92      0.52     0.01                             sin
  0.00      0.52     0.00       11     0.00     0.00  __gcc_deregister_frame
  0.00      0.52     0.00        2     0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00      0.52     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
  0.00      0.52     0.00        2     0.00     5.00  void std::__valarray_copy<std::complex<double>, std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&, unsigned int, std::_Array<std::complex<double> >)
  0.00      0.52     0.00        1     0.00     0.00  convolution(WaveFile, WaveFile)
  0.00      0.52     0.00        1     0.00   166.67  ifft(std::valarray<std::complex<double> >&)
```

Regression Testing

```
cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$ cmp output_dry.wav output_dry_base.wav

cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$
```

To testing the correctness, compare the new output file with the original base version output file.

There is nothing prints after the compare command, which indicates that the two files are identical.

Optimization: Constant

Commit: 6006b3dcff99b3d7aa885a9a7c7d8a7ceea4963a

This optimization replace -2 * PI with a defined constant, for fft function, so the program would not need to dynamically compute the value for each iteration of the for-loop in the runtime.

```
13  13
14  14  // Constants
15  15  #define PI                  3.14159265358979
    16  +#define NEG_DOUBLE_PI       -2 * PI
16  17  #define TONE_FREQUENCY      440 // Frequency of tone to be created (A = 440 Hz)
17  18  #define SAMPLE_RATE         44100.0 //  Standard sample rate in Hz
18  19  #define BITS_PER_SAMPLE       16 //  Standard sample size in bits
    @@ -260,7 +261,7 @@ void fft(ComplexArray& x) {
260 261
261 262     // combine
262 263     for (size_t k = 0; k < n_over_2; ++k) {
263     -       Complex t = polar(1.0, -2 * PI * k / n) * odd[k];
    264 +       Complex t = polar(1.0, NEG_DOUBLE_PI * k / n) * odd[k];
264 265         Complex even_k = even[k];
265 266
266 267         x[k] = even_k + t;
```

Profiling

```
  %   cumulative   self              self     total
 time   seconds   seconds    calls  ms/call  ms/call  name
 87.23      0.41     0.41        3   136.67   136.67  fft(std::valarray<std::complex<double> >&)
  4.26      0.43     0.02                             __muldc3
  2.13      0.44     0.01                             _fu1___ZSt4cout
  2.13      0.45     0.01                             _fu21___ZSt4cout
  2.13      0.46     0.01                             _fu45___ZSt4cout
  2.13      0.47     0.01                             _fu9___ZSt4cout
  0.00      0.47     0.00  4194304     0.00     0.00  std::complex<double> std::conj<double>(std::complex<double> const&)
  0.00      0.47     0.00       11     0.00     0.00  __gcc_deregister_frame
  0.00      0.47     0.00        2     0.00     0.00  WaveFile::read(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >)
  0.00      0.47     0.00        2     0.00     0.00  WaveFile::WaveFile(WaveFile const&)
  0.00      0.47     0.00        2     0.00     0.00  void std::__valarray_copy<std::complex<double>, std::_RefFunClos<std::_ValArray, std::complex<double> > >(std::_Expr<std::_RefFunClos<std::_ValArray, std::complex<double> >, std::complex<double> > const&, unsigned int, std::_Array<std::complex<double> >)
  0.00      0.47     0.00        1     0.00     0.00  convolution(WaveFile, WaveFile)
  0.00      0.47     0.00        1     0.00   136.67  ifft(std::valarray<std::complex<double> >&)
```

Regression Testing

```
cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$ cmp output_dry.wav output_dry_base.wav

cloud@CloudyYoungOmen15 MINGW64 ~/OneDrive/Desktop/cpsc-501-assignment-2 (main)
$
```

To testing the correctness, compare the new output file with the original base version output file.

There is nothing prints after the compare command, which indicates that the two files are identical.