BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÁO CÁO THỰC TẬP CƠ SỞ**

**TÌM HIỂU CẤU TRÚC FILE ÂM THANH (WAV), THỂ HIỆN ĐỒ HỌA DẠNG SÓNG VÀ PHỔ VÀ CÁC THÔNG TIN HEADER**

**Giảng viên hướng dẫn: Nguyễn Thị Hương Lý**

**Sinh viên thực hiện: Mạnh Văn Hiệp**

**Mã số sinh viên: 61133631**

Khánh Hòa – 2022

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÁO CÁO THỰC TẬP CƠ SỞ**

**TÌM HIỂU CẤU TRÚC FILE ÂM THANH (WAV), THỂ HIỆN ĐỒ HỌA DẠNG SÓNG VÀ PHỔ VÀ CÁC THÔNG TIN HEADER**

GVHD: Nguyễn Thị Hương Lý

SVTH: Mạnh Văn Hiệp

MSSV: 61133631

Khánh Hòa – Tháng 01/2022

# LỜI CẢM ƠN

Để hoàn thành bài báo cáo thực tập cơ sở này, trước hết em xin gửi đến quý thầy, cô Khoa Công nghệ Thông tin - Trường Đại học Nha Trang lời cảm ơn chân thành.

Em xin gửi đến cô Nguyễn Thị Hương Lý, người đã tận tình hướng dẫn, giúp đỡ em hoàn thành bài báo cáo này lời cảm ơn sâu sắc nhất.

Trong quá trình nghiên cứu thực hiện đề tài, cũng như là trong quá trình làm bài báo cáo thực tập cơ sở, khó tránh khỏi sai sót, rất mong các thầy, cô bỏ qua. Đồng thời do kiến thức cũng như kinh nghiệm thực tiễn của bản thân còn hạn chế nên đề tài, bài báo cáo cơ sở này khó thể không tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp từ quý thầy, cô để em học hỏi thêm được nhiều kinh nghiệm, cũng như kỹ năng cần thiết.

Em xin chân thành cảm ơn!

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc92358214)

[MỤC LỤC ii](#_Toc92358215)

[Chương 1. TỔNG QUAN VỀ VẤN ĐỀ NGHIÊN CỨU 1](#_Toc92358216)

[1.1 CƠ SỞ LÝ THUYẾT 1](#_Toc92358217)

[1.1.1 Tổng quan về file âm thanh (wav) 1](#_Toc92358218)

[1.1.1.1 Mô tả 1](#_Toc92358219)

[1.1.1.2 Chi tiết kỹ thuật 2](#_Toc92358220)

[1.1.1.3 Cách mở file WAV, WAVE 3](#_Toc92358221)

[1.1.2 Biến đổi Fourier 4](#_Toc92358222)

[1.1.2.1 Ứng dụng 4](#_Toc92358223)

[1.1.2.2 Các dạng của biến đổi Fourier 5](#_Toc92358224)

[1.2 CÔNG CỤ VÀ NGÔN NGỮ LẬP TRÌNH ĐƯỢC SỬ DỤNG 7](#_Toc92358225)

[1.2.1 Tổng quan về VISUAL STUDIO 2019 7](#_Toc92358226)

[1.2.2 Ngôn ngữ C++ 8](#_Toc92358227)

[1.2.3 Ngôn ngữ C# 9](#_Toc92358228)

[Chương 2. Đặc tả bài toán 10](#_Toc92358229)

[2.1 BÀI TOÁN 10](#_Toc92358230)

[Chương 3. CÀI ĐẶT CHƯƠNG TRÌNH 10](#_Toc92358231)

[3.1 Code chương trình 10](#_Toc92358232)

[3.1.1 Mức 1 10](#_Toc92358233)

[3.1.1.1 Đọc thông tin header file wav 10](#_Toc92358234)

[3.1.1.2 Hiển thị dạng sóng 13](#_Toc92358235)

# PHẦN MỞ ĐẦU

1. **Lý do chọn đề tài:**

**Công nghệ thông tin** là một trong những ngành phát triển vượt bậc trong những năm gần đây. Ngày nay với sự phát triển nhanh chóng của xã hội thì công nghệ thông tin được ứng dụng rộng rãi ở hầu hết tất cả các lĩnh vực và ngày càng đóng vai trò quan trọng, trở thành một phần thiết yếu trong đời sống hằng ngày. Công nghệ thông tin là một ngành đòi hỏi ở người học một nền tảng kiến thức vững chắc, sự tư duy logic cao, hiểu biết sâu rộng trên nhiều lĩnh vực. Với chúng em hiện đang là những sinh viên công nghệ thông tin cần phải có sự đầu tư, không ngừng học hỏi để nâng cao kiến thức. Do đó để củng cố lại kiến thức đã học, đề tài mà em chọn để thực hiện là: **Tìm hiểu cấu trúc file âm thanh (wav), thể hiện đồ họa dạng sóng và phổ và các thông tin header.**

1. **Mục tiêu của đề tài.**
   * Củng cố lại kiến thức đã học về các cấu trúc điều khiển, ôn tập lại kiến thức về phần mảng, chuỗi cũng như kiểu dữ liệu cấu trúc...
   * Rèn luyện kỹ năng lập trình trên ngôn ngữ C++, C#.
   * Ứng dụng lý thuyết đã học giải quyết bài toán ứng dụng cụ thể.
2. **Phạm vi nghiên cứu.**
   * Đọc thông tin header file wav, hiện thị dạng song.
   * Lý thuyết về biến đổi Fourier, sử dụng Fourier để phân tích phổ
3. **Phương pháp nghiên cứu.**
   * Tìm kiếm và nghiên cứu trên mạng Internet.

# TỔNG QUAN VỀ VẤN ĐỀ NGHIÊN CỨU

## CƠ SỞ LÝ THUYẾT

### Tổng quan về file âm thanh (wav)

File đuôi có dạng .WAV hoặc .WAVE là các file Waveform Audio được phát triển bởi sự hợp tác của Microsoft và IBM vào những năm 1991.

Mặc dù các file WAV thường không được nén nhưng chúng ta vẫn có thể nén chúng nếu muốn, phù hợp để sử dụng trên các hệ thống Windows, đối với MacOS vẫn có thể sử dụngnhưng không phổ biến bằng.

WAV thường được dùng cho công việc chỉnh sửa âm nhạc, trò chơi điện tử, không được dùng trong chia sẻ vì kích thước tệp khá lớn. Về bản chất, WAV khá giống với các tệp AIFF và 8SVX, cả hai đều được sử dụng phổ biến.



WAV là định dạng file chứa âm thanh của Windows

#### ****Mô tả****

WAV chứa các tác phẩm âm nhạc, bản ghi âm giọng nói và các hiệu ứng âm thanh. Có thể phát được trong QuickTime, Windows Media Player và một số phần mềm hỗ trợ khác.

WAV có kích thước lớn hơn MP3, bạn sẽ rất tốn nhiều thời gian cho việc chia sẻ file qua Internet.

Một trong những ưu điểm chính của WAV được liên kết với việc sử dụng Điều chế mã xung tuyến tính (LPCM) để lưu trữ luồng âm thanh.



File WAV thường dùng trong phần mềm chỉnh sửa âm thanh

#### ****Chi tiết kỹ thuật****

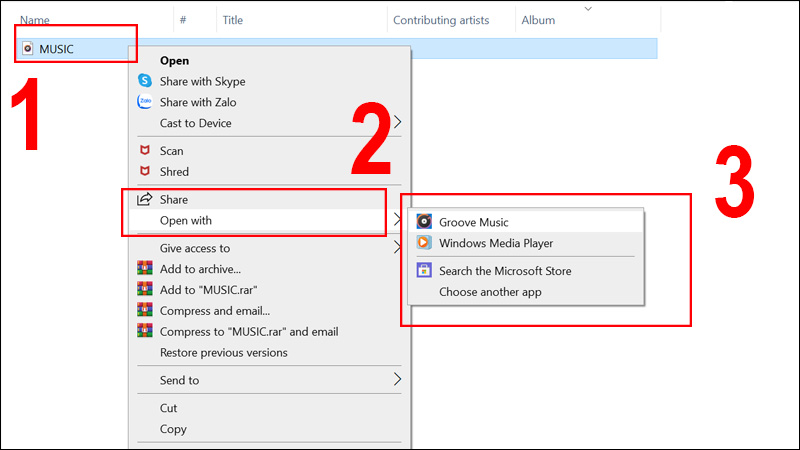
* WAV được ghi thành 2 loại 8 bit và 16 bit cho mỗi mẫu.
* Một tùy chọn tiêu chuẩn cho CD Audio là một dòng âm thanh 16 bit cho mỗi mẫu và tần số lấy mẫu là 44,1 KHz.
* Một giây của âm thanh tương ứng với 88 Kb bộ nhớ trong.
* Các tệp WAV có thể lưu trữ siêu dữ liệu trong đoạn INFO và chúng cũng bao gồm các danh sách IFF được tích hợp.
* Trong một số trường hợp, định dạng chuẩn có thể được sử dụng để phát sóng.



WAV được phát triển bởi Microsoft và IBM

#### ****Cách mở file WAV, WAVE****

Nhấn chuột phải vào file cần mở > **Open with**>Chọn phần mềm bạn muốn dùng là hoàn tất.



Chọn file muốn mở với Open with

### Biến đổi Fourier

**Biến đổi Fourier** hay **chuyển hóa Fourier**, được đặt tên theo nhà toán học người Pháp [Joseph Fourier](https://vi.wikipedia.org/wiki/Joseph_Fourier), là phép [biến đổi](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_t%C3%ADch_ph%C3%A2n" \o "Biến đổi tích phân) một [hàm số](https://vi.wikipedia.org/wiki/H%C3%A0m_s%E1%BB%91" \o "Hàm số) hoặc một tín hiệu theo miền thời gian sang miền tần số. Chẳng hạn như một bản nhạc có thể được phân tích dựa trên tần số của nó.

#### Ứng dụng

Biến đổi Fourier có rất nhiều ứng dụng [khoa học](https://vi.wikipedia.org/wiki/Khoa_h%E1%BB%8Dc), ví dụ như trong [vật lý](https://vi.wikipedia.org/wiki/V%E1%BA%ADt_l%C3%BD_h%E1%BB%8Dc" \o "Vật lý học), [số học](https://vi.wikipedia.org/wiki/S%E1%BB%91_h%E1%BB%8Dc" \o "Số học), [xử lý tín hiệu](https://vi.wikipedia.org/wiki/X%E1%BB%AD_l%C3%BD_t%C3%ADn_hi%E1%BB%87u" \o "Xử lý tín hiệu), [xác suất](https://vi.wikipedia.org/wiki/X%C3%A1c_su%E1%BA%A5t" \o "Xác suất), [thống kê](https://vi.wikipedia.org/wiki/Khoa_h%E1%BB%8Dc_Th%E1%BB%91ng_k%C3%AA" \o "Khoa học Thống kê), [mật mã](https://vi.wikipedia.org/wiki/M%E1%BA%ADt_m%C3%A3_h%E1%BB%8Dc" \o "Mật mã học), [âm học](https://vi.wikipedia.org/wiki/%C3%82m_h%E1%BB%8Dc" \o "Âm học), [hải dương học](https://vi.wikipedia.org/wiki/H%E1%BA%A3i_d%C6%B0%C6%A1ng_h%E1%BB%8Dc" \o "Hải dương học), [quang học](https://vi.wikipedia.org/wiki/Quang_h%E1%BB%8Dc" \o "Quang học), [hình học](https://vi.wikipedia.org/wiki/H%C3%ACnh_h%E1%BB%8Dc" \o "Hình học) và rất nhiều lĩnh vực khác. Trong xử lý tín hiệu và các ngành liên quan, biến đổi Fourier thường được nghĩ đến như sự chuyển đổi [tín hiệu](https://vi.wikipedia.org/wiki/T%C3%ADn_hi%E1%BB%87u" \o "Tín hiệu) thành các thành phần [biên độ](https://vi.wikipedia.org/wiki/Bi%C3%AAn_%C4%91%E1%BB%99" \o "Biên độ) và [tần số](https://vi.wikipedia.org/wiki/T%E1%BA%A7n_s%E1%BB%91" \o "Tần số). Sự ứng dụng rộng rãi của biến đổi Fourier bắt nguồn từ những [tính chất](https://vi.wikipedia.org/w/index.php?title=T%C3%ADnh_ch%E1%BA%A5t&action=edit&redlink=1" \o "Tính chất (trang không tồn tại)) hữu dụng của biến đổi này:

* Tính [tuyến tính](https://vi.wikipedia.org/wiki/Tuy%E1%BA%BFn_t%C3%ADnh" \o "Tuyến tính): {\displaystyle {\mathfrak {F}}[a.f+b.g]=a.{\mathfrak {F}}[f]+b.{\mathfrak {F}}[g]}
* Tồn tại biến đổi nghịch đảo, và thực tế là biến đổi Fourier nghịch đảo gần như có cùng dạng với biến đổi thuận.
* Những hàm số sin cơ sở là các [hàm riêng](https://vi.wikipedia.org/w/index.php?title=H%C3%A0m_ri%C3%AAng&action=edit&redlink=1" \o "Hàm riêng (trang không tồn tại)) của phép [vi phân](https://vi.wikipedia.org/wiki/%C4%90%E1%BA%A1o_h%C3%A0m_v%C3%A0_vi_ph%C3%A2n_c%E1%BB%A7a_h%C3%A0m_s%E1%BB%91), có nghĩa là khai triển này biến những [phương trình vi phân](https://vi.wikipedia.org/wiki/Ph%C6%B0%C6%A1ng_tr%C3%ACnh_vi_ph%C3%A2n" \o "Phương trình vi phân) tuyến tính với các hệ số không đổi thành các [phương trình đại số](https://vi.wikipedia.org/wiki/Ph%C6%B0%C6%A1ng_tr%C3%ACnh_%C4%91%E1%BA%A1i_s%E1%BB%91" \o "Phương trình đại số) cơ bản. Ví dụ, trong một hệ vật lý tuyến tính không phụ thuộc thời gian, [tần số](https://vi.wikipedia.org/wiki/T%E1%BA%A7n_s%E1%BB%91" \o "Tần số) là một đại lượng không đổi, do đó những thành phần tần số khác nhau có thể được tính toán một cách độc lập.
* Theo [định lý tích tổng chập](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BB%8Bnh_l%C3%BD_t%C3%ADch_t%E1%BB%95ng_ch%E1%BA%ADp&action=edit&redlink=1" \o "Định lý tích tổng chập (trang không tồn tại)), biến đổi Fourier chuyển một [tích tổng chập](https://vi.wikipedia.org/w/index.php?title=T%C3%ADch_t%E1%BB%95ng_ch%E1%BA%ADp&action=edit&redlink=1" \o "Tích tổng chập (trang không tồn tại)) phức tạp thành một tích đại số đơn giản.
* [Biến đổi Fourier rời rạc](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_r%E1%BB%9Di_r%E1%BA%A1c) có thể được tính toán một cách nhanh chóng bằng máy tính nhờ thuật toán [FFT](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_nhanh) (*fast Fourier transform*).
* Theo [định lý Parseval-Plancherel](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BB%8Bnh_l%C3%BD_Parseval-Plancherel&action=edit&redlink=1" \o "Định lý Parseval-Plancherel (trang không tồn tại)), [năng lượng](https://vi.wikipedia.org/wiki/N%C4%83ng_l%C6%B0%E1%BB%A3ng" \o "Năng lượng) của [tín hiệu](https://vi.wikipedia.org/wiki/T%C3%ADn_hi%E1%BB%87u" \o "Tín hiệu) (tích phân của bình phương giá trị tuyệt đối của hàm) không đổi sau biến đổi Fourier.

#### Các dạng của biến đổi Fourier

##### Biến đổi Fourier liên tục

Thông thường, tên gọi biến đổi Fourier[[1]](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier#cite_note-1) được gắn cho [biến đổi Fourier liên tục](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_li%C3%AAn_t%E1%BB%A5c" \o "Biến đổi Fourier liên tục), biến đổi này biểu diễn một hàm bình phương khả tích *f(t)* bất kì theo tổng của các hàm e lũy thừa phức với tần số góc ω và biên độ phức *F*(ω):

{\displaystyle f(t)={\mathcal {F}}^{-1}(F)(t)={\frac {1}{2\pi }}\int \limits \_{-\infty }^{\infty }F(\omega )e^{i\omega t}\,d\omega .} Diagram, schematic

Description automatically generated

Đây là biến đổi nghịch đảo của biến đổi Fourier liên tục, trong khi biến đổi Fourier biểu diễn hàm *F*(ω) theo *f(t)*.

##### Chuỗi Fourier

Biến đổi Fourier liên tục là dạng tổng quát của một khái niệm có từ trước, đó là [chuỗi Fourier](https://vi.wikipedia.org/wiki/Chu%E1%BB%97i_Fourier" \o "Chuỗi Fourier). Chuỗi Fourier khai triển các [hàm tuần hoàn](https://vi.wikipedia.org/wiki/H%C3%A0m_tu%E1%BA%A7n_ho%C3%A0n" \o "Hàm tuần hoàn) *f(x)* với chu kì 2π (hoặc các hàm có tập xác định bị chặn) theo chuỗi của các hàm sin:

A picture containing schematic

Description automatically generated

trong đó {\displaystyle F\_{n}}*F*n là biên độ phức. Cho các hàm thực, chuỗi Fourier có thể được viết dưới dạng:

Text, letter

Description automatically generated

trong đó *an* và *bn* là các hằng số Fourier (giá trị thực).

##### Biến đổi Fourier rời rạc

Trong [toán học](https://vi.wikipedia.org/wiki/To%C3%A1n_h%E1%BB%8Dc" \o "Toán học), phép **biến đổi Fourier rời rạc (DFT)**, đôi khi còn được gọi là [biến đổi Fourier hữu hạn](https://vi.wikipedia.org/w/index.php?title=Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_h%E1%BB%AFu_h%E1%BA%A1n&action=edit&redlink=1" \o "Biến đổi Fourier hữu hạn (trang không tồn tại)), là một biến đổi trong [giải tích Fourier](https://vi.wikipedia.org/wiki/Gi%E1%BA%A3i_t%C3%ADch_Fourier" \o "Giải tích Fourier) cho các tín hiệu thời gian rời rạc. Đầu vào của biến đổi này là một chuỗi hữu hạn các [số thực](https://vi.wikipedia.org/wiki/S%E1%BB%91_th%E1%BB%B1c" \o "Số thực) hoặc [số phức](https://vi.wikipedia.org/wiki/S%E1%BB%91_ph%E1%BB%A9c" \o "Số phức), làm biến đổi này là một công cụ lý tưởng để xử lý thông tin trên các [máy tính](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh" \o "Máy tính). Đặc biệt, biến đổi này được sử dụng rộng rãi trong [xử lý tín hiệu](https://vi.wikipedia.org/wiki/X%E1%BB%AD_l%C3%BD_t%C3%ADn_hi%E1%BB%87u" \o "Xử lý tín hiệu) và các ngành liên quan đến phân tích tần số chứa trong một tín hiệu, để giải [phương trình đạo hàm riêng](https://vi.wikipedia.org/wiki/Ph%C6%B0%C6%A1ng_tr%C3%ACnh_vi_ph%C3%A2n_ri%C3%AAng_ph%E1%BA%A7n" \o "Phương trình vi phân riêng phần), và để làm các phép như [tích chập](https://vi.wikipedia.org/wiki/T%C3%ADch_ch%E1%BA%ADp" \o "Tích chập). Biến đổi này có thể được tính nhanh bởi thuật toán [biến đổi Fourier nhanh](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_nhanh" \o "Biến đổi Fourier nhanh) (FFT).

###### Định nghĩa

Dãy của *N* [số phức](https://vi.wikipedia.org/wiki/S%E1%BB%91_ph%E1%BB%A9c" \o "Số phức):{\displaystyle x\_{0},...,x\_{N-1}} x0, …, xN-1 được biến đổi thành chuỗi của *N* số phức *X*0,..., *XN*−1 bởi công thức sau đây:

Text, letter

Description automatically generated

với *e* là [cơ số của lôgarit tự nhiên](https://vi.wikipedia.org/wiki/S%E1%BB%91_e" \o "Số e), {\displaystyle i\,}i là [đơn vị ảo](https://vi.wikipedia.org/wiki/%C4%90%C6%A1n_v%E1%BB%8B_%E1%BA%A3o" \o "Đơn vị ảo) ({\displaystyle i^{2}=-1}i2 = -1), và π là [pi](https://vi.wikipedia.org/wiki/Pi). Phép biến đổi đôi khi được ký hiệu bởi {\displaystyle {\mathcal {F}}}

Phép **biến đổi Fourier rời rạc ngược (IDFT)** được cho bởi công thức sau

Text, letter

Description automatically generated

Những phương trình này có thể được mô tả đơn giản như sau: các số phức *Xk* đại diện cho biên độ và pha ở các bước sóng khác nhau của "tín hiệu vào" *xn*. Phép biến đổi DFT tính các giá trị *Xk* từ các giá trị *xn*, trong khi IDFT tính *xn* bằng tổng của các sóng thành phần {\displaystyle {\frac {1}{N}}X\_{k}e^{{\frac {2\pi i}{N}}kn}} Text

Description automatically generated với [tần số](https://vi.wikipedia.org/wiki/T%E1%BA%A7n_s%E1%BB%91" \o "Tần số) *k / N*. Khi viết các phương trình dưới dạng như trên, ta đã sử dụng [công thức Euler](https://vi.wikipedia.org/wiki/C%C3%B4ng_th%E1%BB%A9c_Euler" \o "Công thức Euler) để biểu diễn các hàm lượng giác dưới dạng lũy thừa số phức để biến đổi được dễ dàng. Khi viết *Xk* dưới dạng [tọa độ cực](https://vi.wikipedia.org/wiki/H%E1%BB%87_t%E1%BB%8Da_%C4%91%E1%BB%99_c%E1%BB%B1c" \o "Hệ tọa độ cực), ta thu được biên độ *Ak / N* và pha *φk* từ modulus và argument của *Xk*:

Text, letter

Description automatically generated

trong đó [atan2](https://vi.wikipedia.org/w/index.php?title=Atan2&action=edit&redlink=1) là dạng hai đối số của hàm [arctan](https://vi.wikipedia.org/w/index.php?title=Arctan&action=edit&redlink=1). Cần ghi chú rằng các thừa số chuẩn hóa của DFT và IDFT (ở đây là 1 và 1/*N*) và dấu của các số mũ chỉ là quy ước, và có thể khác nhau trong các tài liệu khác nhau. Điều kiện duy nhất cho các quy ước này là DFT và IDFT có dấu ngược nhau ở các số mũ và tích của hai thừa số chuẩn hóa phải là 1/*N*.

###### Ứng dụng

DFT có nhiều ứng dụng rộng rãi trong nhiều ngành khác nhau. Tất cả các ứng dụng của DFT đều dựa trên một tính chất quan trọng là DFT và IDFT đều có thể được tính nhanh chóng bằng thuật toán [biến đổi Fourier nhanh](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_nhanh" \o "Biến đổi Fourier nhanh).

**Phân tích phổ**

Khi sử dụng DFT để [phân tích phổ](https://vi.wikipedia.org/w/index.php?title=Ph%E1%BB%95_t%E1%BA%A7n_s%E1%BB%91&action=edit&redlink=1" \o "Phổ tần số (trang không tồn tại)), dãy {*x\_n*} thường đại diện cho một dãy hữu hạn các mẫu tại các thời điểm cách đều nhau của một tín hiệu *x(t)*, trong đó *t* để chỉ thời gian. Việc chuyển từ thời gian liên tục sang mẫu (thời gian rời rạc) chuyển [biến đổi Fourier liên tục](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_li%C3%AAn_t%E1%BB%A5c" \o "Biến đổi Fourier liên tục) của *x(t)* thành [biến đổi Fourier thời gian rời rạc](https://vi.wikipedia.org/w/index.php?title=Bi%E1%BA%BFn_%C4%91%E1%BB%95i_Fourier_th%E1%BB%9Di_gian_r%E1%BB%9Di_r%E1%BA%A1c&action=edit&redlink=1" \o "Biến đổi Fourier thời gian rời rạc (trang không tồn tại)) (DTFT), và thường gây ra [hiệu ứng răng cưa](https://vi.wikipedia.org/w/index.php?title=Hi%E1%BB%87u_%E1%BB%A9ng_r%C4%83ng_c%C6%B0a&action=edit&redlink=1" \o "Hiệu ứng răng cưa (trang không tồn tại)). Việc chọn lựa tần số lấy mẫu thích hợp (xem [tần số Nyquist](https://vi.wikipedia.org/wiki/T%E1%BA%A7n_s%E1%BB%91_Nyquist" \o "Tần số Nyquist)) là vô cùng quan trọng cho việc giảm thiểu hiệu ứng này.

## CÔNG CỤ VÀ NGÔN NGỮ LẬP TRÌNH ĐƯỢC SỬ DỤNG

### Tổng quan về VISUAL STUDIO 2019

* **Microsoft Visual Studio** là một [môi trường phát triển tích hợp](https://vi.wikipedia.org/wiki/M%C3%B4i_tr%C6%B0%E1%BB%9Dng_ph%C3%A1t_tri%E1%BB%83n_t%C3%ADch_h%E1%BB%A3p" \o "Môi trường phát triển tích hợp) (IDE) từ [Microsoft](https://vi.wikipedia.org/wiki/Microsoft). Microsoft Visual Studio còn được gọi là "Trình soạn thảo mã nhiều người sử dụng nhất thế giới ", được dùng để lập trình [C++](https://vi.wikipedia.org/wiki/C%2B%2B) và [C#](https://vi.wikipedia.org/wiki/C) là chính. Nó được sử dụng để phát triển [chương trình máy tính](https://vi.wikipedia.org/wiki/Ch%C6%B0%C6%A1ng_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh" \o "Chương trình máy tính) cho [Microsoft Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), cũng như các trang web, các ứng dụng web và các [dịch vụ web](https://vi.wikipedia.org/wiki/D%E1%BB%8Bch_v%E1%BB%A5_web" \o "Dịch vụ web). Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như [Windows API](https://vi.wikipedia.org/wiki/Windows_API), [Windows Forms](https://vi.wikipedia.org/wiki/Windows_Forms), [Windows Presentation Foundation](https://vi.wikipedia.org/wiki/Windows_Presentation_Foundation), [Windows Store](https://vi.wikipedia.org/wiki/Windows_Store) và [Microsoft Silverlight](https://vi.wikipedia.org/wiki/Microsoft_Silverlight). Nó có thể sản xuất cả hai [ngôn ngữ máy](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_m%C3%A1y" \o "Ngôn ngữ máy) và [mã số quản lý](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_s%E1%BB%91_qu%E1%BA%A3n_l%C3%BD&action=edit&redlink=1" \o "Mã số quản lý (trang không tồn tại)).
* **Ưu điểm:**
* Đa nền tảng.
* Đa ngôn ngữ lập trình
* Hỗ trợ website.
* Kho tiện ích mở rộng phong phú.
* Lưu trữ phân cấp.
* Kho lưu trữ an toàn.
* …
* **Hạn chế:**
* Tuy có giao diện đẹp, nhưng khá khó sử dụng cho người mới học lập trình. Vì khá nhiều bước thiết lập và cài đặt.
* Vì Visual Studio là một IDE nặng nên cần sử dụng nhiều tài nguyên để khởi động và vận hành.
* …

### Ngôn ngữ C++

C++ là một loại ngôn ngữ lập trình bậc trung (middle-level). Đây là ngôn ngữ lập trình đa năng được tạo ra bởi Bjarne Stroustrup như một phần mở rộng của ngôn ngữ lập trình C, hoặc "C với các lớp Class", Ngôn ngữ đã được mở rộng đáng kể theo thời gian và C ++ hiện đại có các tính năng: lập trình tổng quát, lập trình hướng đối tượng, lập trình thủ tục, ngôn ngữ đa mẫu hình tự do có kiểu tĩnh, dữ liệu trừu tượng, và lập trình đa hình, ngoài ra còn có thêm các tính năng, công cụ để thao tác với bộ nhớ cấp thấp. Từ thập niên 1990, C++ đã trở thành một trong những ngôn ngữ thương mại ưa thích và phổ biến của lập trình viên.

* **Ưu điểm:**
* Là ngôn ngữ lập trình hướng đối tượng.
* Cho phép thoải mái quản lý vùng nhớ
* Là một ngôn ngữ low-level dễ dàng giao tiếp với phần cứng.
* …
* **Hạn chế:**
* Tràn vùng nhớ.
* OOP trong C++ khá phức tạp và khó hiểu.
* Con trỏ là một cái gì đó rất … khó.
* …

### Ngôn ngữ C#

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

**C#** (**C Sharp**, đọc là *"xi-sáp"*) là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh" \o "Ngôn ngữ lập trình) [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng" \o "Lập trình hướng đối tượng) đa năng, mạnh mẽ được phát triển bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft), C# là phần khởi đầu cho kế hoạch [.NET](https://vi.wikipedia.org/wiki/.NET_Framework) của họ. Tên của ngôn ngữ bao gồm ký tự thăng theo Microsoft nhưng theo [ECMA](https://vi.wikipedia.org/w/index.php?title=ECMA&action=edit&redlink=1) là **C#**, chỉ bao gồm dấu số thường. Microsoft phát triển C# dựa trên [C++](https://vi.wikipedia.org/wiki/C%2B%2B) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)). C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, [Visual Basic](https://vi.wikipedia.org/wiki/Visual_Basic), [Delphi](https://vi.wikipedia.org/wiki/Delphi_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)).

* **Ưu điểm:**
* C Sharp là ngôn ngữ đơn giản.
* Ngôn ngữ đa nền tảng, an toàn và hiệu quả.
* Là ngôn ngữ hiện đại thông dụng, ít từ khóa dễ hiểu.
* Là ngôn ngữ lập trình thuần hướng đối tượng
* …
* **Hạn chế:**
* Nhược điểm lớn nhất của C# là chỉ chạy trên nền Windows và có cài .NET Framework. Thao tác đối với phần cứng yếu hơn so với ngôn ngữ khác, hầu hết phải dựa vào windows.
* …

# Đặc tả bài toán

## BÀI TOÁN

1. Vận dụng kiến thức lập trình C, C#. Biết làm việc với file nhị phân. Tìm hiểu biến đổi Fourier để phân tích phổ
   1. **Mức 1:** Đọc và thể hiện các thông tin header của file wav, hiển thị dạng sóng
   2. **Mức 2:** Phân tích phổ và hiển thị

# CÀI ĐẶT CHƯƠNG TRÌNH

## Code chương trình

### Mức 1

#### Đọc thông tin header file wav

##### **Code**

|  |
| --- |
| *#define \_CRT\_SECURE\_NO\_DEPRECATE*  *#include <iostream>*  *#include <string>*  *#include <fstream>*  *#include <cstdint>*  *using std::cin;*  *using std::cout;*  *using std::endl;*  *using std::fstream;*  *using std::string;*  *typedef struct WAV\_HEADER*  *{*  */\* RIFF Chunk Descriptor \*/*  *uint8\_t RIFF[4]; // RIFF Header Magic header*  *uint32\_t ChunkSize; // RIFF Chunk Size*  *uint8\_t WAVE[4]; // WAVE Header*  */\* "fmt" sub-chunk \*/*  *uint8\_t fmt[4]; // FMT header*  *uint32\_t Subchunk1Size; // Size of the fmt chunk*  *uint16\_t AudioFormat; // Audio format 1=PCM,6=mulaw,7=alaw, 257=IBM Mu-Law, 258=IBM A-Law, 259=ADPCM*  *uint16\_t NumOfChan; // Number of channels 1=Mono 2=Sterio*  *uint32\_t SamplesPerSec; // Sampling Frequency in Hz*  *uint32\_t bytesPerSec; // bytes per second*  *uint16\_t blockAlign; // 2=16-bit mono, 4=16-bit stereo*  *uint16\_t bitsPerSample; // Number of bits per sample*  */\* "data" sub-chunk \*/*  *uint8\_t Subchunk2ID[4]; // "data" string*  *uint32\_t Subchunk2Size; // Sampled data length*  *} wav\_hdr;*  *// Function prototypes*  *int getFileSize(FILE\* inFile);*  *int main(int argc, char\* argv[])*  *{*  *wav\_hdr wavHeader;*  *int headerSize = sizeof(wav\_hdr), filelength = 0;*  *const char\* filePath;*  *string input;*  *if (argc <= 1)*  *{*  *cout << "Input wave file name: ";*  *cin >> input;*  *cin.get();*  *filePath = input.c\_str();*  *}*  *else*  *{*  *filePath = argv[1];*  *cout << "Input wave file name: " << filePath << endl;*  *}*  *FILE\* wavFile = fopen(filePath, "r");*  *if (wavFile == nullptr)*  *{*  *fprintf(stderr, "Unable to open wave file: %s\n", filePath);*  *return 1;*  *}*  *//Read the header*  *size\_t bytesRead = fread(&wavHeader, 1, headerSize, wavFile);*  *cout << "Header Read " << bytesRead << " bytes." << endl;*  *if (bytesRead > 0)*  *{*  *//Read the data*  *uint16\_t bytesPerSample = wavHeader.bitsPerSample / 8; //Number of bytes per sample*  *//uint64\_t numSamples = wavHeader.ChunkSize / bytesPerSample; //How many samples are in the wav file?*  *static const uint16\_t BUFFER\_SIZE = 4096;*  *int8\_t\* buffer = new int8\_t[BUFFER\_SIZE];*  *while ((bytesRead = fread(buffer, sizeof buffer[0], BUFFER\_SIZE / (sizeof buffer[0]), wavFile)) > 0)*  *{*  */\*\* DO SOMETHING WITH THE WAVE DATA HERE \*\*/*  *cout << "Read " << bytesRead << " bytes." << endl;*  *}*  *delete[] buffer;*  *buffer = nullptr;*  *filelength = getFileSize(wavFile);*  *cout << "File is :" << filelength << " bytes." << endl;*  *cout << "RIFF header :" << wavHeader.RIFF[0] << wavHeader.RIFF[1] << wavHeader.RIFF[2] << wavHeader.RIFF[3] << endl;*  *cout << "WAVE header :" << wavHeader.WAVE[0] << wavHeader.WAVE[1] << wavHeader.WAVE[2] << wavHeader.WAVE[3] << endl;*  *cout << "FMT :" << wavHeader.fmt[0] << wavHeader.fmt[1] << wavHeader.fmt[2] << wavHeader.fmt[3] << endl;*  *cout << "Data size :" << wavHeader.ChunkSize << endl;*  *// Display the sampling Rate from the header*  *cout << "Sampling Rate :" << wavHeader.SamplesPerSec << endl;*  *cout << "Number of bits used :" << wavHeader.bitsPerSample << endl;*  *cout << "Number of channels :" << wavHeader.NumOfChan << endl;*  *cout << "Number of bytes per second :" << wavHeader.bytesPerSec << endl;*  *cout << "Data length :" << wavHeader.Subchunk2Size << endl;*  *cout << "Audio Format :" << wavHeader.AudioFormat << endl;*  *// Audio format 1=PCM,6=mulaw,7=alaw, 257=IBM Mu-Law, 258=IBM A-Law, 259=ADPCM*  *cout << "Block align :" << wavHeader.blockAlign << endl;*  *cout << "Data string :" << wavHeader.Subchunk2ID[0] << wavHeader.Subchunk2ID[1] << wavHeader.Subchunk2ID[2] << wavHeader.Subchunk2ID[3] << endl;*  *}*  *fclose(wavFile);*  *return 0;*  *}*  *// find the file size*  *int getFileSize(FILE\* inFile)*  *{*  *int fileSize = 0;*  *fseek(inFile, 0, SEEK\_END);*  *fileSize = ftell(inFile);*  *fseek(inFile, 0, SEEK\_SET);*  *return fileSize;*  *}* |

##### **Kết quả chạy thử chương trình**

**Text

Description automatically generated**

#### Hiển thị dạng sóng

##### **Code**

|  |
| --- |
| *using System;*  *using System.Collections.Generic;*  *using System.ComponentModel;*  *using System.Data;*  *using System.Drawing;*  *using System.Linq;*  *using System.Text;*  *using System.Threading.Tasks;*  *using System.Windows.Forms;*  *namespace Plotting\_Audio\_Waveforms*  *{*  *public partial class Form1 : Form*  *{*  *public Form1()*  *{*  *InitializeComponent();*  *}*  *private void openWaveToolStripMenuItem\_Click(object sender, EventArgs e)*  *{*  *OpenFileDialog open = new OpenFileDialog();*  *open.Filter = "Wave File (\*.wav)|\*.wav;";*  *if (open.ShowDialog() != DialogResult.OK) return;*  *chart1.Series.Add("wave");*  *chart1.Series["wave"].ChartType = System.Windows.Forms.DataVisualization.Charting.SeriesChartType.FastLine;*  *chart1.Series["wave"].ChartArea = "ChartArea1";*  *NAudio.Wave.WaveChannel32 wave = new NAudio.Wave.WaveChannel32(new NAudio.Wave.WaveFileReader(open.FileName));*  *byte[] buffer = new byte[16384];*  *int read = 0;*  *while (wave.Position < wave.Length)*  *{*  *read = wave.Read(buffer, 0, 16384);*  *for (int i = 0; i < read / 4; i++)*  *{*  *chart1.Series["wave"].Points.Add(BitConverter.ToSingle(buffer, i \* 4));*  *}*  *}*  *}*  *}*  *}* |

##### **Kết quả chạy thử chương trình**

**Chart, histogram

Description automatically generated**

##### **Kết quả chạy file wav trên phần mềm Audacity**

**Chart

Description automatically generated with medium confidence**

### Mức 2

Graphical user interface, application

Description automatically generated

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## KẾT QUẢ ĐẠT ĐƯỢC

* + Đọc được thông tin header của file wav, hiển thị dạng sóng.
  + Hiển thị phổ âm thanh.

## ƯU ĐIỂM

* + Chương trình dễ sử dụng, giao diện thân thiện.
  + Bài toán hiển thị phổ âm thanh có thể làm tiền đề để giải quyết các bài toán khó hơn, phức tạp hơn

## HẠN CHẾ

* Giao diện hệ thống chưa được tối ưu cho trải nghiệm người dùng.
* Code chương trình còn dài, còn nhiều chỗ chưa được tối ưu.

## HƯỚNG PHÁT TRIỂN

* Hoàn thiện thêm những chức năng hệ thống hiện có, cải thiện về giao diện, trải nghiệm người.

## KẾT LUẬN

Ngày nay, cuộc cách mạng công nghiệp 4.0 và vai trò của Internet ngày càng được khẳng định và không thể thiếu trong quản lý và chia sẻ thông tin của một quốc gia hay địa phương. Cùng với sự phát triển mạnh mẽ của hệ thống mạng toàn cầu – Internet và nhu cầu chia sẻ, tra cứu thông tin trên Internet.

Định dạng tệp WAV không nén, nên nó hoạt động tốt hơn về chất lượng âm thanh và tốc độ bit, không giới hạn về độ sâu bit.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Wikipedia tiếng việt, "C Sharp (ngôn ngữ lập trình)," 07 01 2021. [Online]. Available: https://vi.wikipedia.org/wiki/C\_Sharp\_(ng%C3%B4n\_ng%E1%BB%AF\_l%E1%BA%ADp\_  tr%C3%ACnh). [Accessed 07 01 2021]. |
| [2] | Wikipedia tiếng việt, “Biến đổi Fourier,” 7 1 2022. [Trực tuyến]. Available: https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn\_%C4%91%E1%BB%95i\_Fourier. [Đã truy  cập 7 1 2022]. |
| [3] | Wikipedia tiếng việt, "Biến đổi Fourier rời rạc," 7 1 2022. [Online]. Available: https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn\_%C4%91%E1%BB%95i\_Fourier\_r%E1%  BB%9Di\_r%E1%BA%A1c. [Accessed 7 1 2022]. |
| [4] | Wikipedia, “Microsoft Visual Studio,” 7 1 2022. [Trực tuyến]. Available: https://vi.wikipedia.org/wiki/Microsoft\_Visual\_Studio. [Đã truy cập 7 1 2022]. |
| [5] | Wikipedia tiếng việt, "C++," 7 1 2022. [Online]. Available: https://vi.wikipedia.org/wiki/C%2B%2B. [Accessed 7 1 2022]. |
| [6] | GiawaVideos, "C# Audio Tutorial 10 - Plotting Audio Waveforms," [Online]. Available: https://www.youtube.com/watch?v=ZnFoVuOVrUQ. |
| [7] | stackoverflow, "C++ Reading the Data part of a WAV file," [Online]. Available: https://stackoverflow.com/questions/13660777/c-reading-the-data-part-of-a-wav-file. |
| [8] | S. Harden, “Realtime Microphone Audio FFT Graph with C#,” [Trực tuyến]. Available: https://www.youtube.com/watch?v=q9cRZuosrOs. |