**TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM  
KHOA CÔNG NGHỆ THÔNG TIN**

**-----\*\*\*-----**



**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN “XỬ LÝ ẢNH”**

***ĐỀ TÀI:***

***XÂY DỰNG CHƯƠNG TRÌNH NHẬN DẠNG QR CODE***

***Sinh viên thực hiện: Nguyễn Văn Bách - 88596***

***Hải Phòng, ngày 10 tháng 04 năm 202******3***

MỤC LỤC

[CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN 2](#_Toc132576893)

[1.1 Giới thiệu bài toán 2](#_Toc132576894)

[1.2 Mô tả bài toán 3](#_Toc132576895)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 4](#_Toc132576896)

[2.1. Giới thiệu về QR Code 4](#_Toc132576897)

[2.1.1 Giới thiệu chung 4](#_Toc132576898)

[2.1.2 Đặc điểm 5](#_Toc132576899)

[2.1.3 Cấu trúc mã QR 7](#_Toc132576900)

[2.1.4 Ứng dụng 10](#_Toc132576901)

[2.2 Chương trình nhận dạng QR code 11](#_Toc132576902)

[2.2.1 Ngôn ngữ lập trình 11](#_Toc132576903)

[2.2.2 Thư viện 11](#_Toc132576904)

[2.2.3 Source Code 12](#_Toc132576905)

[2.2.4 Chi tiết chương trình chính 14](#_Toc132576906)

[CHƯƠNG 3: CÀI ĐẶT THỬ NGHIỆM 16](#_Toc132576907)

[3.1 Công cụ cài đặt 16](#_Toc132576908)

[3.2 Kết quả chương trình 17](#_Toc132576909)

[3.3 Kết luận 19](#_Toc132576910)

# CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN

## 1.1 Giới thiệu bài toán

Trong những năm gần đây công nghệ thông tin đang phát triển một cách mạnh mẽ và càng ngày càng xuất hiện nhiều hơn trong cuộc sống hàng ngày của chúng ta, và nhờ có những công nghệ mới mà đã giúp cuộc sống chúng ta thuận tiện hơn rất nhiều. Trong nhiều năm gần đây, có lẽ chúng ta đã không còn xa lạ với hình ảnh những ô vuông nhỏ xen kẽ các dòng đen trắng bên trong được xuất hiện ở rất nhiền nơi từ internet cũng như trước các tấm áp phích quảng cáo hay tại nhiều cơ quan nhà nước. Đặc biệt trong đợt dịch Covid-19 nhà nước đã sử dụng mã QR để người dân có thể khai báo y tế khi tới những nơi công cộng hay cơ quan. Vâng đó chính là mã QR, thứ mà chỉ cần có điện thoại khi dùng ứng dụng chuyên dụng quét mã ta sẽ tới được thông tin mà ta cần.

## 1.2 Mô tả bài toán

Sự phổ biến của QR code với cuộc sống chúng ta là điều không phải bàn cãi. Bài toán hôm nay chúng ta sẽ làm một chương trình có thể nhận diện và chỉ định được mã QR và hiển thị đường dẫn của nó thông qua webcam cũng như camera laptop với đầu vào là những hình ảnh mã QR được hiện trên camera

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## QR CODE LÀ GÌ? - Bảo Hộ Thương Hiệu2.1. Giới thiệu về QR Code

### 2.1.1 Giới thiệu chung

Mã QR (mã phản hồi nhanh) là một mã vạch ma trận (hay mã vạch hai chiều) được phát triển bởi công ty Denso Wave (Nhật Bản) vào năm 1994. So với mã vạch chỉ có thể lưu được tối đa 20 ký tự, QR code ra đời đã đáp ứng được nhu cầu lưu trữ nhiều thông tin hơn nữa.

Chữ "QR" xuất phát từ "Quick Response", trong tiếng Anh có nghĩa là *đáp ứng nhanh hay xử lí nhanh*, vì người tạo ra nó có ý định cho phép mã được giải mã ở tốc độ cao.

QR Code bao gồm những chấm đen và ô vuông mẫu trên nền trắng, có thể chứa những thông tin như URL, thời gian, địa điểm của sự kiện, mô tả, giới thiệu một sản phẩm nào đó,...

QR Code cho phép quét và đọc mã nhanh hơn bằng các thiết bị như máy đọc mã vạch hoặc điện thoại có camera với ứng dụng cho phép quét mã, vô cùng tiện lợi cho người dùng.

### 2.1.2 Đặc điểm

* Khả năng lưu trữ lớn:

Có thể nhập tối đa 7089 chữ số hoặc 4296 ký tự, bao gồm dấu câu và ký tự đặc biệt trong một Mã. Ngoài các số và ký tự, từ và cụm từ (ví dụ: địa chỉ Internet) cũng có thể được mã hóa. Khi có thêm dữ liệu được thêm vào Mã QR, kích thước Mã tăng lên và cấu trúc Mã trở nên phức tạp hơn.

Đối với từng loại dữ liệu thì được mã hóa cụ thể số lượng các kí tự như sau:

* Số đơn thuần Tối đa 7.089 ký tự
* Số và chữ cái Tối đa 4.296 ký tự
* Số nhị phân (8 bit) Tối đa 2.953 byte
* Kanji/Kana Tối đa 1.817 ký tự
* Đọc tất cả các hướng (360 °) tốc độ cao

Mã QR có các mẫu tìm kiếm để thông báo vị trí của biểu tượng được sắp xếp theo ba góc của nó để cho phép đọc tốc độ cao theo mọi hướng (360 °). Tỷ lệ giữa màu đen và trắng giữa các dòng quét chạy qua các mẫu tìm luôn luôn là 1: 1: 3: 1: 1 khi nhìn từ bất kỳ hướng nào trong số 360 ° xung quanh nó.

* Chịu được các ký hiệu bị méo

Các ký hiệu thường bị méo khi gắn vào bề mặt cong hoặc bởi đầu đọc bị nghiêng. Để sửa méo méo này, mã QR có các mẫu căn chỉnh được sắp xếp với khoảng thời gian thông thường trong phạm vi biểu tượng.

* Chức năng khôi phục dữ liệu (Chống lại các biểu tượng bị mờ hoặc bị hư hại)

QR code có khả năng ứng phó với trường hợp bị bẩn hay rách. Ngay cả khi một bộ phận code bị mất đi nữa, bản thân QR code có khả năng sửa chữa lỗi, khôi phục data mà nó chứa.

Khả năng sửa chữa lỗi của QR code được chia làm 4 mức độ: L, M, Q, H. Mức độ sửa chữa lỗi càng cao thì khả năng kháng lại lỗi rách, hỏng của QR code càng cao.

|  |  |  |
| --- | --- | --- |
| **Mức độ sửa chữa lỗi của QR code** | **Độ khôi phục** | **Ứng dụng** |
| Level L | 7% | Sử dụng trong môi trường ít bị bẩn |
| Level M | 15% | Sử dụng trong môi trường thông thường |
| Level Q | 25% | Sử dụng trong môi trường dễ bị bẩn như xưởng sản xuất |
| Level H | 30% | Sử dụng trong môi trường dễ bị bẩn như xưởng sản xuất |

### 2.1.3 Cấu trúc mã QR

Nhìn mắt thường chúng ta sẽ thấy các mẫu pixel đen và trắng xuất hiện ngay từ cái nhìn đầu tiên là một trò chơi ô chữ nhỏ và dường như được sáng tác một cách ngẫu nhiên

Nhưng nếu bạn nhìn kỹ, các cấu trúc nhất định có thể được xác định. Để máy quét nhận dạng mã QR như vậy, mã phải luôn là hình vuông. Một số yếu tố bổ sung đảm bảo rằng thông tin được đọc chính xác.

**Dấu vị trí (Hoa văn định vị)**

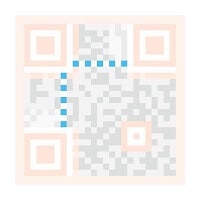
Chúng biểu thị hướng mà mã được in.

Nhờ vào hoa văn định vị này, camerra có thể xác định được phạm vi QR code cũng như đọc được thông tin ngay cả trong trường hợp QR code bị biến dạng, nhờ đó ta có thể quét được QR code một cách nhanh chóng ở bất kỳ góc độ nào.

  
**Ký hiệu căn chỉnh**

Nếu Mã QR lớn, phần tử bổ sung này sẽ giúp định hướng.

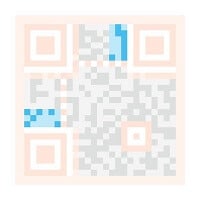
Giúp cho việc điều chỉnh lại những chênh lệch phát sinh do camera bị lệch trong quá trình quét.



**Mẫu thời gian**

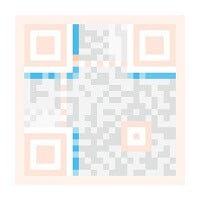
Sử dụng những dòng này, máy quét xác định độ lớn của ma trận dữ liệu.

Các ô vuông đen trắng được đặt xen kẽ nhau nhằm giúp cho việc xác định toạ độ của QR code.



**Thông tin phiên bản**

Chúng chỉ định phiên bản Mã QR đang được sử dụng. Hiện tại có 40 phiên bản Mã QR khác nhau.



**Thông tin định dạng**

Các mẫu định dạng chứa thông tin về khả năng chịu lỗi và mẫu mặt nạ dữ liệu và giúp bạn dễ dàng quét Mã.



**Dữ liệu và các phím sửa lỗi**

Những mẫu này chứa dữ liệu thực tế.



**Khu vực yên tĩnh**

Khoảng cách này rất quan trọng đối với chương trình quét để phân biệt Mã QR với môi trường xung quanh.

### 2.1.4 Ứng dụng

* Kiểm kê hàng hóa, thông tin sản phẩm.
* Lưu trữ URL: Sử dụng điện thoại để có thể đọc QR Code để lấy URL và tự mở web trên trình duyệt.
* Ứng dụng tại các bến xe bus, xe lửa, tàu điện ngầm: Người sử dụng khi quét mã QR của bến xe sẽ biết được thông tin về các chuyến xe.
* Sử dụng tại các viện bảo tàng, tức người tham quan sẽ quét mã QR đặt cạnh vật trưng bày sẽ biết được thông tin về đồ vật đó.
* Ứng dụng tại các siêu thị để biết thông tin về sản phẩm xuất xứ, hạn sử dụng…
* Sử dụng trong các buổi hội thảo, thuyết trình, tổ chức sự kiện, chương trình nhạc hội, bar, club, liveshow người tham gia có thể sử dụng QR Code thay cho Business Card của mình.
* Sử dụng tại khách sạn, nhà hàng, coffee shop…. Để biết được các công thức, cách chế biến món ăn, đồ uống, thông tin khách sạn…
* Sử dụng trong truyền thông quảng cáo, marketing: QR đang dần thay thế các quảng cáo dưới hình thức in ấn và phát tờ rơi.

## 2.2 Chương trình nhận dạng QR code

### Ngôn ngữ lập trình

Chương trình của chúng ta sẽ sử dụng Python để lập trình

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML). Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.

### 2.2.2 Thư viện

* Thư viện OpenCV

OpenCV là một thư viện mã nguồn mở hàng đầu cho thị giác máy tính (computer vision), xử lý ảnh và máy học, và các tính năng tăng tốc GPU trong hoạt động thời gian thực.

* Sử dụng thư viện Numpy

Numpy (Numeric Python): là một thư viện toán học phổ biến và mạnh mẽ của Python. Cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng “core Python” đơn thuần.

* Sử dụng thư viện Zbar

ZBar là bộ phần mềm nguồn mở để đọc mã vạch từ nhiều nguồn khác nhau, chẳng hạn như luồng video, tệp hình ảnh và cảm biến cường độ thô. Nó hỗ trợ nhiều *ký hiệu* phổ biến (các loại mã vạch) bao gồm EAN-13/UPC-A, UPC-E, EAN-8, Code 128, Code 39, Interleaved 2 of 5 và QR Code.

### 2.2.3 Source Code

#khai báo thư viện

import cv2

import numpy as np

from pyzbar.pyzbar import decode

#thiết lập webcam, camera để nhận các hình ảnh QR code

cap = cv2.VideoCapture(0)

cap.set(3, 640)

cap.set(4, 480)

#chương trình chính

while True:

success, img = cap.read()

for barcode in decode(img):

myData = barcode.data.decode('utf-8')

print(myData)

pts = np.array([barcode.polygon], np.int32)

pts = pts.reshape((-1, 1, 2))

cv2.polylines(img, [pts], True, (255, 0, 255), 5)

pts2 = barcode.rect

cv2.putText(img, myData, (pts2[0], pts2[1]), cv2.FONT\_HERSHEY\_SIMPLEX, 0.9, (255, 0, 255), 2)

cv2.imshow('Result', img)

cv2.waitKey(1)

### 2.2.4 Chi tiết chương trình chính

* Biến các dữ liệu của mã QR từ kiểu byte về kiểu chuỗi

myData = barcode.data.decode('utf-8')

* Vẽ hộp giới hạn là khung đa giác bao quanh QR code

cv2.polylines(img, [pts], True, (255, 0, 255), 5)

cv2.polylines(image, [pts], isClosed, color, thickness)

+ image: khung hình

+ pts: mảng các đường cong đa giác

+ isClosed: có đóng các đa giác không

+ color: màu sắc

+ thickness: độ dày

* Hàm hiện chữ ở trên hộp giới hạn QR code

cv2.putText(img, myData, (pts2[0], pts2[1]), cv2.FONT\_HERSHEY\_SIMPLEX, 0.9, (255, 0, 255), 2)

cv2.putText(image, text, org, font, fontScale, color[, thickness[, lineType[, bottomLeftOrigin]]])

+ img: khung hình

+ text: dữ liệu hiện lên

+ org: vị trí của dữ liệu (text) hiện trên màn hình

+ fontFace: font chữ

+ fontScale: cỡ chữ

+ color: màu

+ thickness: độ dày (không bắt buộc)

+ lineType: loại dòng (không bắt buộc)

+ bottomLeftOrigin: nằm góc dưới cùng hay trên cùng bên trái (không bắt buộc)

# CHƯƠNG 3: CÀI ĐẶT THỬ NGHIỆM

## 3.1 Công cụ cài đặt

* Tải Pycharm từ đường dẫn dưới tùy theo hệ điều hành:

<https://www.jetbrains.com/pycharm/download/>



* Tạo project mới
* Chọn File => Settings => Project: Tutorial => Project Interpreter
* Cài đặt 3 thư viện OpenCV, Numpy, Zbar theo các cú pháp

+ pip install opencv-python

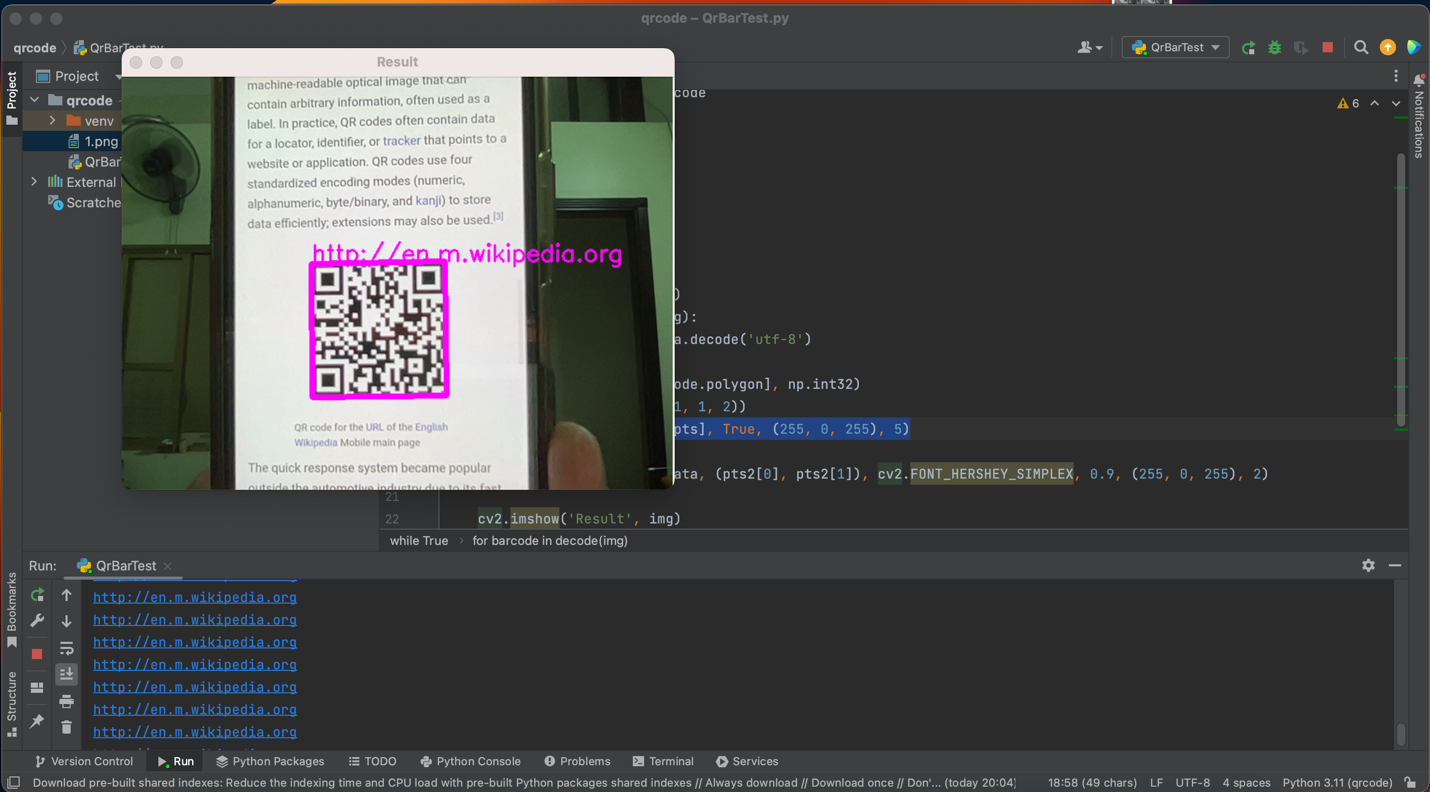
+ pip install numpy

+ pip install pyzbar

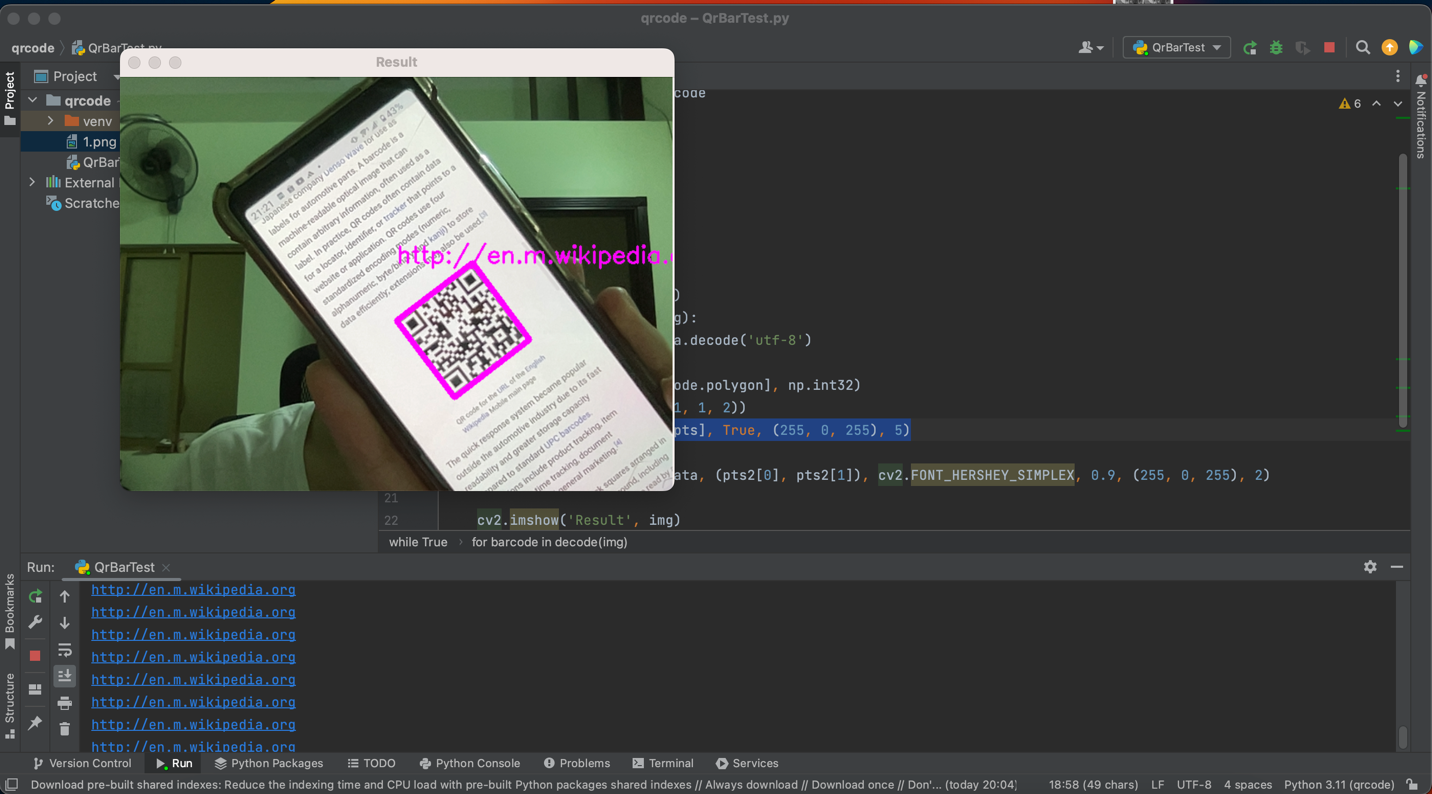
* Run (chạy chương trình)

## 3.2 Kết quả chương trình

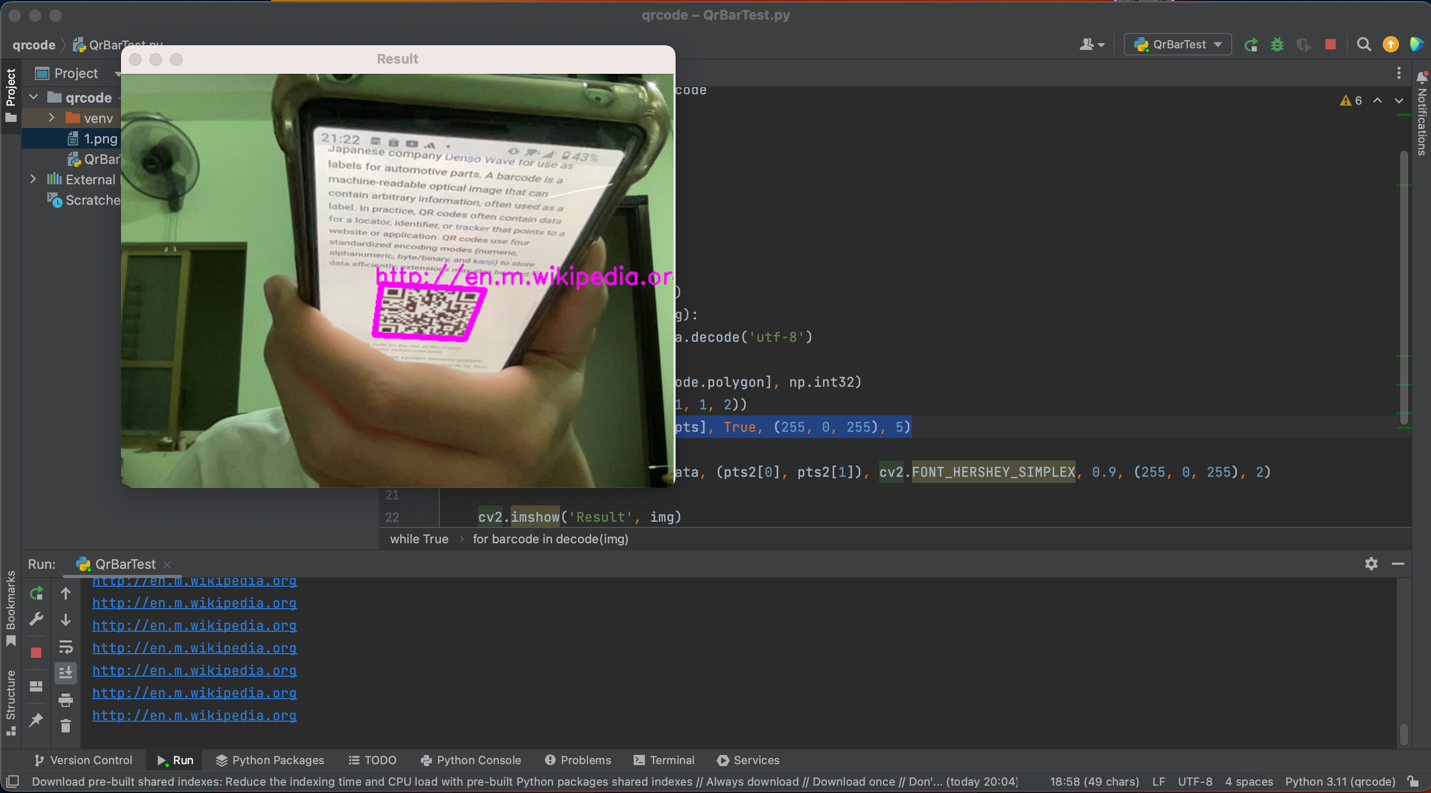
* Chương trình chạy thông thường khung hình vuông, chữ nhật



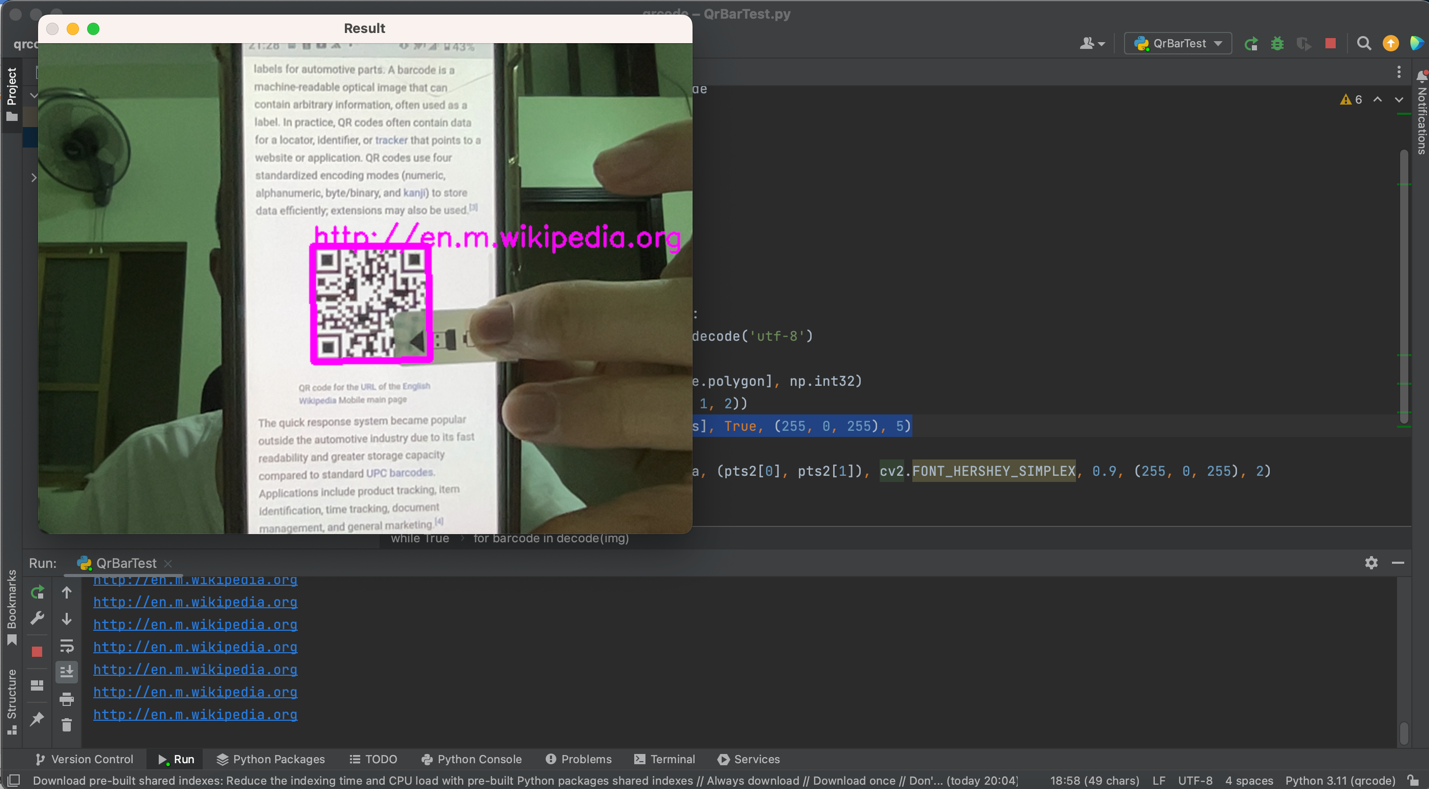
* Chương trình khi xoay QR code để test khung hình đa giác



* Chương trình khi nghiêng QR để test khả năng quét nhiều hướng



* Chương trình test khả năng sửa lỗi của QR khi bị che khuất



## 3.3 Kết luận

* Qua bài tập sinh viên đã hiểu thêm và biết thêm nhiều điều hay về QR code và có thể tạo được một chương trình nhận diện đơn giản
* Ưu điểm: chương trình khá nhạy bén, có thể nhận diện được nhiều QR code cùng một lúc, đáp ứng được các đặc điểm của QR code như khả năng đọc từ nhiều hướng, khôi phục dữ liệu, …
* Nhược điểm: khả năng nhận diện của chương trình chưa được tối ưu khi nhiều khung hình thì hộp giới hạn vẫn còn nhấp nháy