|  |
| --- |
| 基本資料 |
| * 姓名：王豪逸 * 標題：B024020014\_Assignment 9(基本) |
| 作業目的與設計 |
| * 目標：語音辨識控制開關 * 電路設計說明(總分1分)：   LED燈GPIO: 紅16, 黃20, 綠21。共用電阻接地。     電路設計照片(總分1分)：     * 程式邏輯設計(總分1分)：   為了不讓說話的語句太死或是差一兩個字而無法執行，採用提取片段資訊來辨識要執行的動作。先檢測語句是否包含動作(打開、關閉)，之後在搜尋語句中是否有紅、黃、綠，包含則讓該顏色LED執行先前檢測的動作。   * 影片URL：https://youtu.be/6Ax9djdhMDI |
| 心得(總分1分) |
| 在家中安靜的環境下辨識得效果感覺比在教室好很多，上課時常常辨識不出結果或是被背景聲音干擾導致一直收音無法進行下個動作。  第一次接上音響語音輸出沒聲音一度以為他壞掉，最後轉到幾乎最大聲才聽得見。 |
| 程式碼 |
| 程式碼：  ====基本題===(總分1分)  # -\*- coding: utf-8 -\*-  import speech\_recognition as sr  import RPi.GPIO as GPIO  from gtts import gTTS  import os  filename = "tmp.mp3"  leds = {"紅": 16, "黃": 20, "綠": 21}  def main():  GPIO.setmode(GPIO.BCM)  for key, value in leds.items():  GPIO.setup(value, GPIO.OUT)  GPIO.output(value, False)  r = sr.Recognizer()  r.energy\_threshold = 4000  try:  while True:  with sr.Microphone() as source:  print("Start.....")  audio = r.listen(source)  try:  stt = r.recognize\_google(audio, language="zh-tw").encode('utf-8')  print("Input: " + stt)  action = None  toSay = ""  if stt.find("打開") >= 0:  action = True  toSay = "打開"  elif stt.find("關閉") >= 0:  action = False  toSay = "關閉"  if action is not None:  for k, pin in leds.items():  if stt.find(k) >= 0:  GPIO.output(pin, action)  Say("{}色的LED燈{}了".format(k, toSay))  except sr.UnknownValueError:  print("Google Speech Recognition could not understand audio")  except sr.RequestError as e:  print("Could not request results from Google Speech Recognition service; {0}".format(e))  except KeyboardInterrupt:  GPIO.cleanup()  def Say(str):  aud = gTTS(text=str, lang="zh-tw")  aud.save(filename)  os.system("mpg321 "+filename)  if \_\_name\_\_=="\_\_main\_\_":  main() |