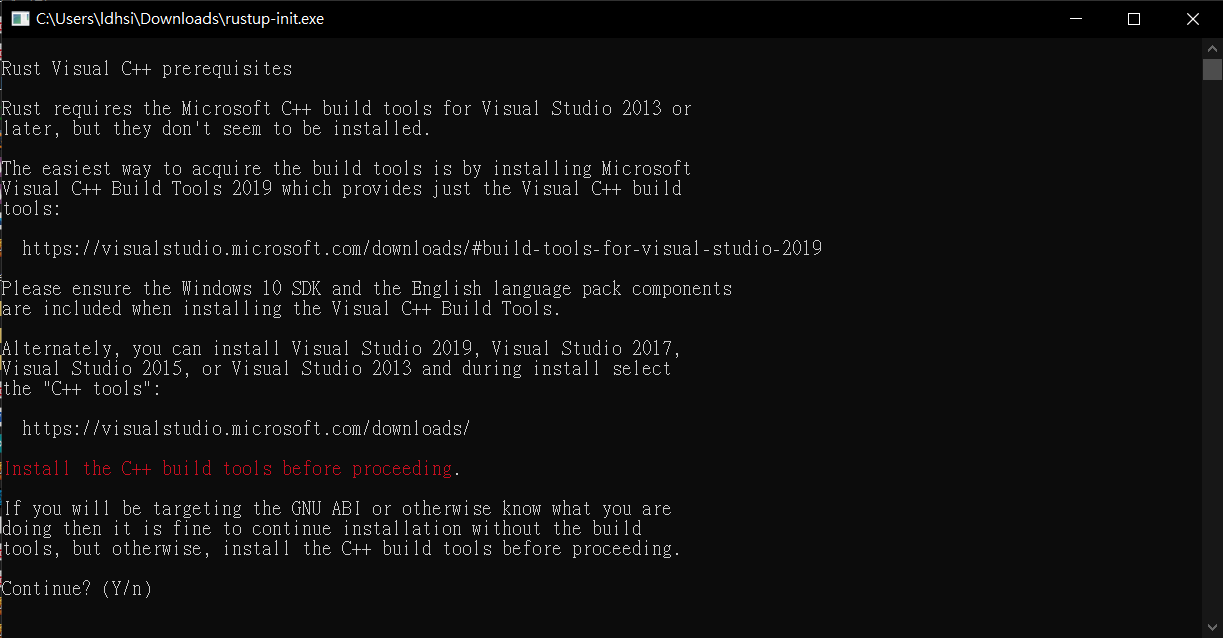
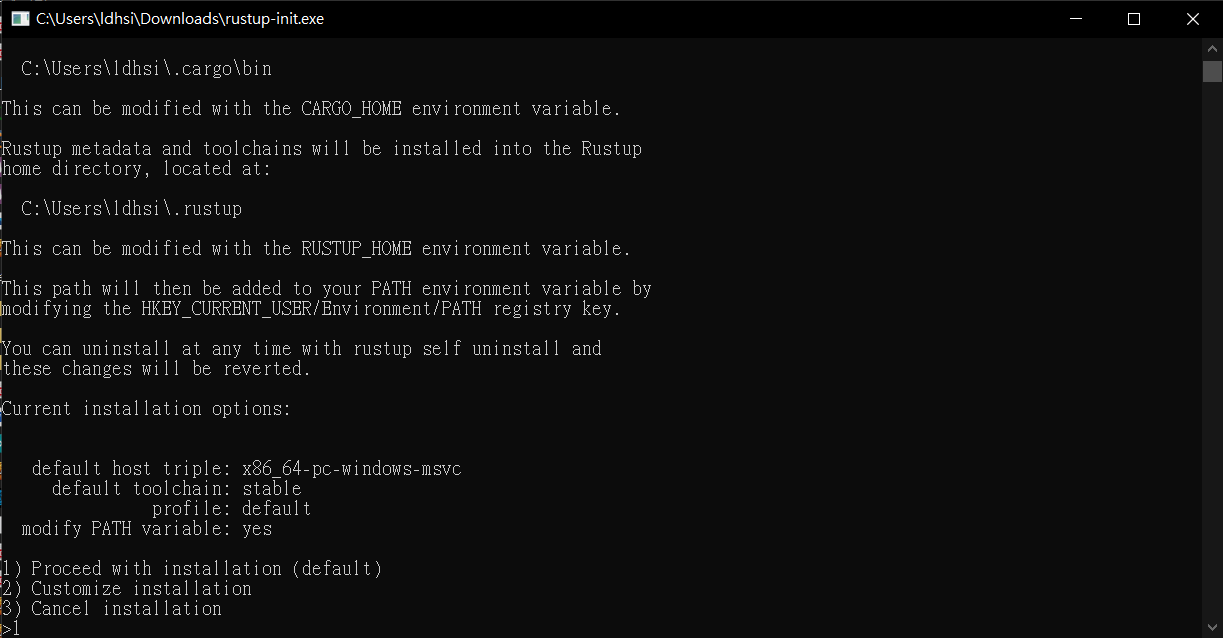
資工二 蔡宏庠 110710505

## 在Windows上

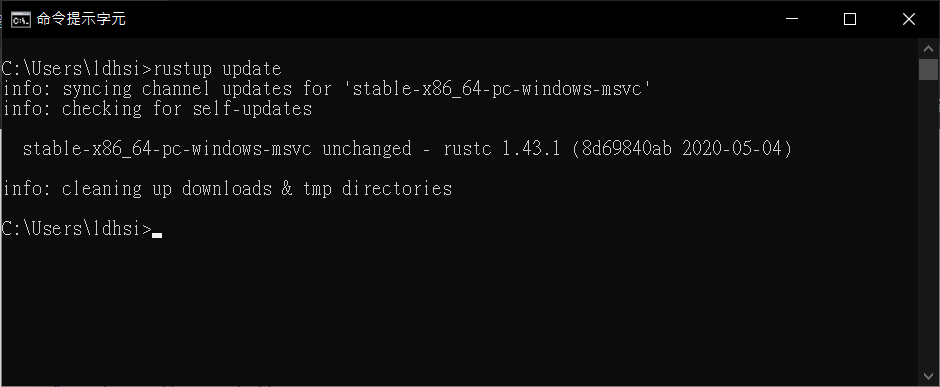
### 1. rustup從<https://www.rust-lang.org/learn/get-started>安裝

* 

### 2.運行可執行文件（輸入“ y”）

* [](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/install_screenshot_executable.png)  
  [](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/install_screenshot_executable_installation.png)

### 3.更新並檢查

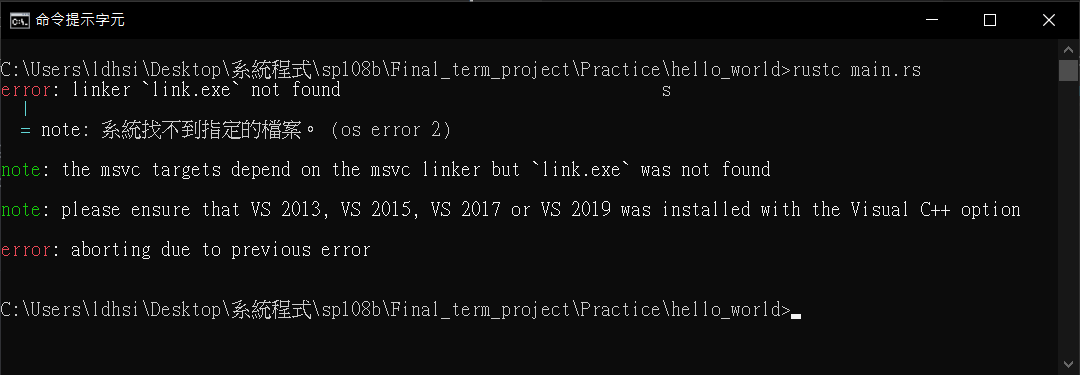
* 打開命令提示符（cmd）
* 用於rustup update更新  
  [](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/install_screenshot_rustUpdate.png)
* 用於rustc --version檢查您是否rustup成功安裝
* 成功！  
  [](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/install_screenshot_rustSuccess.png)

HELLO WORLD

## 處理

### 創建一個以“ .rs”作為文件擴展名的文件>輸入以下代碼：

* fn main(){
* println!("Hello, world!");
* }

然後出現一個錯誤  
  
[](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/compile_screenshot_HelloWorld_failure.png)

### 這是因為我沒有裝C++ build tools =>安裝他!

### 一切安裝完畢後重啟cmd

### 輸入 rustc main.rs

### 輸入 .\main.exe

### [compile_screenshot_HelloWorld_success](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/compile_screenshot_HelloWorld_success.png)

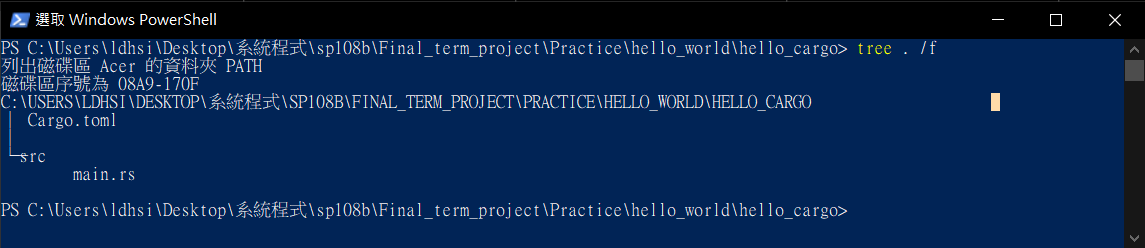
### Cargo

### Cargo是Rust的構建系統和包管理器。大多數Rustacean們使用Cargo來管理他們的Rust項目，因為它可以為你處理很多任務，比如構建代碼、下載依賴庫並編譯這些庫。（我們把代碼所需要的庫叫做**依賴**（dependencies））。

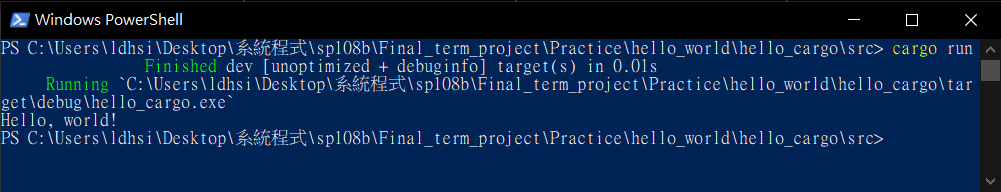
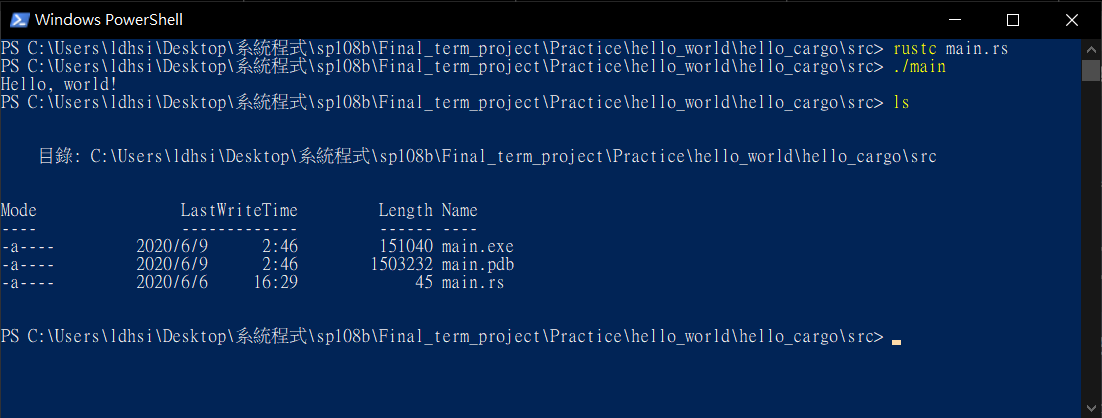
**處理**

**輸入cargo --version以確保cargo可用**

**輸入cargo new hello\_cargo創建貨物目錄**

**cd到hello\_cargo和tree. /f它**  
  
[](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/cargo_screenshot_creat_file.png)**Cargo.toml**

* [package]
* name = "hello\_cargo"
* version = "0.1.0"
* authors = ["ArthurLiao0816 <ArthurLiao0816@yahoo.com>"]
* edition = "2018"
* # See more keys and their definitions at https://doc.rust-lang.org/cargo/reference/manifest.html
* [dependencies]
* [package] 定義一個包
* name 是這個包裹的名字
* version 是此軟件包的版本  
  + "x.y.z" ：
  + x ：進行不兼容的API更改時的主要版本。
  + y ：以向後兼容的方式添加功能時為MINOR版本。
  + z ：進行向後兼容的bug修復時的PATCH版本。
* authors 是這個軟件包的作者
* edition 銹版
* [dependencies] 包庫依賴

**6. cd到src目錄並運行裡面的文件**  
  
[](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/cargo_screenshot_cargo-run_src_main.png)[](https://github.com/ArthurLiao0816/sp108b/blob/master/Final_term_project/Picture/cargo_screenshot_compile_src_main.png)