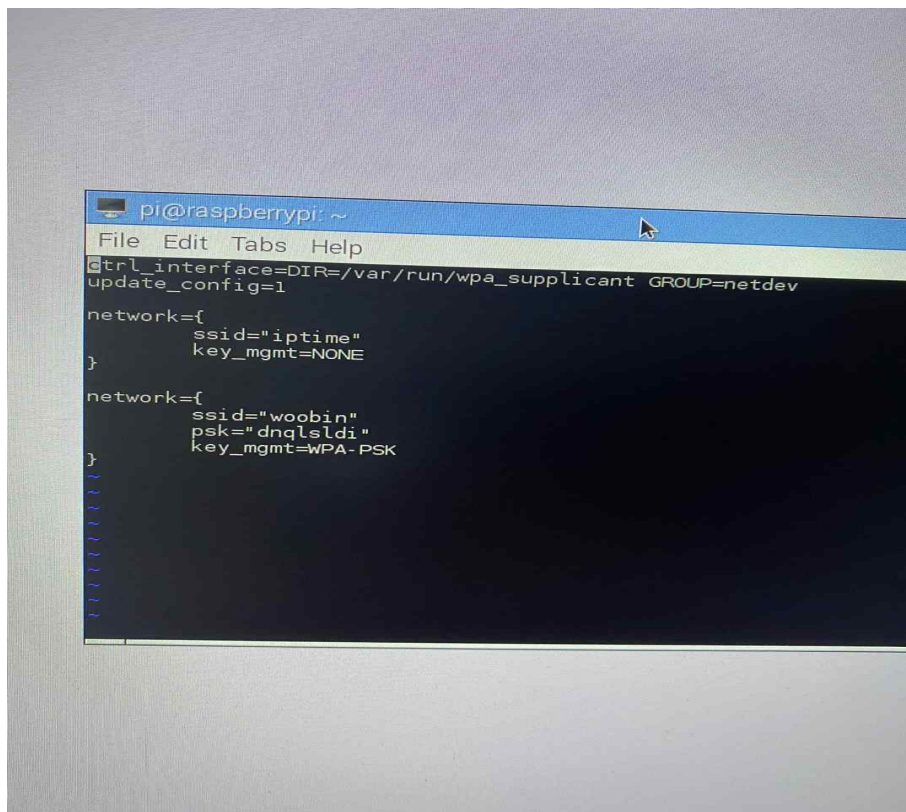


먼저 pdf에 나와있는 것처럼 sd카드에 usb에 꽂고 부팅시킨 뒤 홈페이지에 링크의 라즈비안 Os를 올렸습니다. 그리고나서 키보드, hdmi케이블, 마우스등을 차례로 연결시킨뒤 라즈베리파이를 구동시켰습니다.

먼저 구동 후에 터미널 창을 열어서 키보드 레이아웃을 변경하고 GPIO핀을 사용하기 위해 SPI를 활성화 시켰습니다.

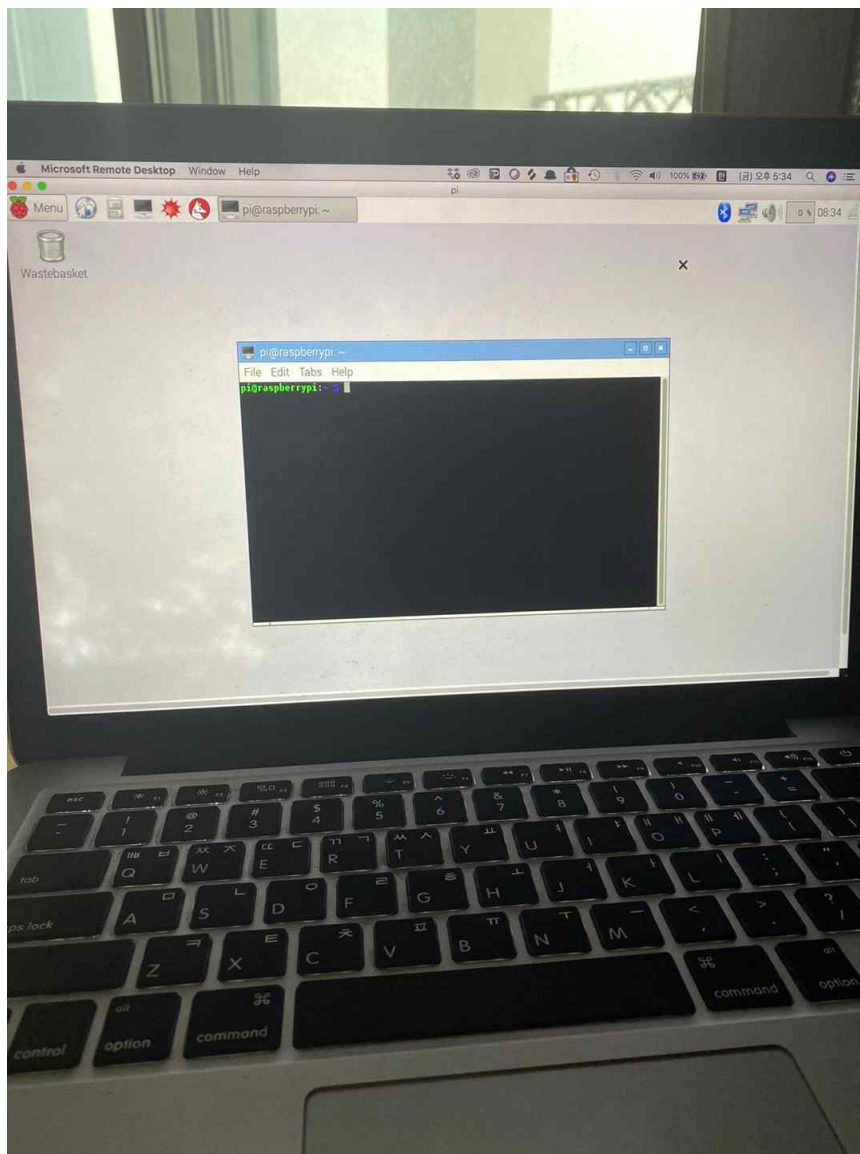
그리고 나서 라즈베리파이를 재부팅 시킨 후 다음과 같이 와이파이를 사용할 수 있게끔 vi 에디터를 통해 입력하였습니다. (이 때 sudo로 명령어를 실행해야 파일을 저장하고 쓸 수 있었습니다.) Wifi 설정을 한 후 라즈베리파이를 재부팅 시키고 나서 ifconfig 명령어를 통해 와이파이가 잘 연결되었는 것을 확인하였습니다.



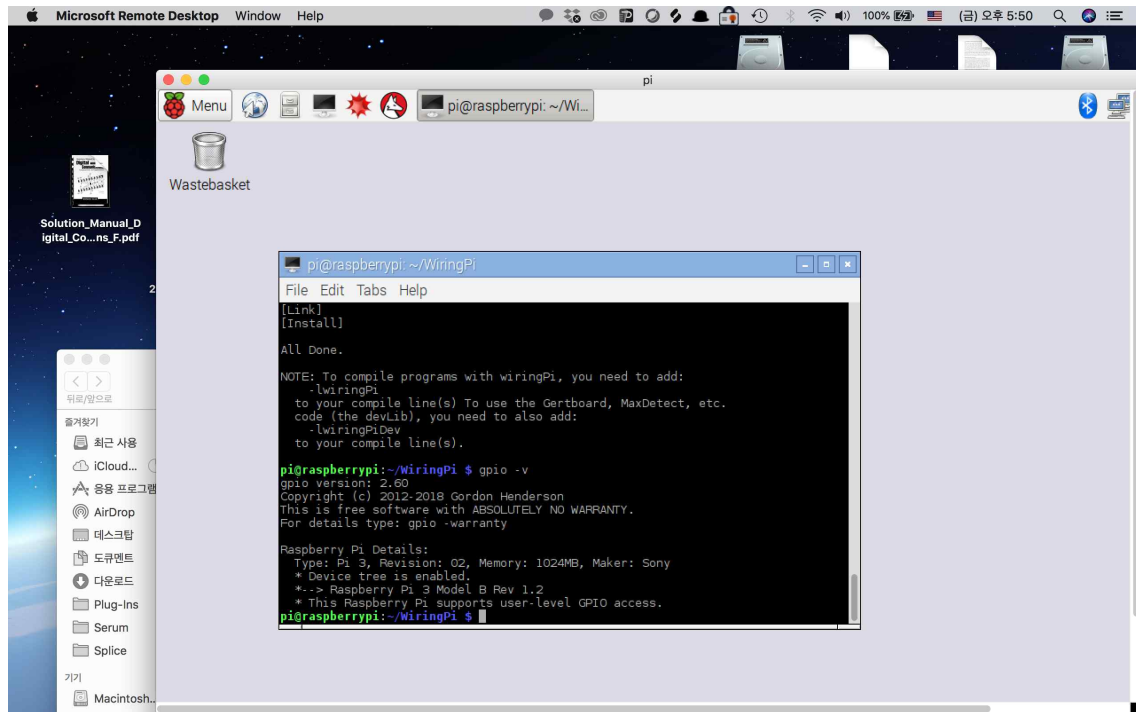
<집에서 사용중인 iptime 공유기는 비밀번호가 없습니다. 이때의 key_mgmt= None>

이후 `sudo password root` 명령어를 통해 root권한으로 관리자 명령어를 실행할 수 있게끔 비밀번호를 설정해 주었습니다.

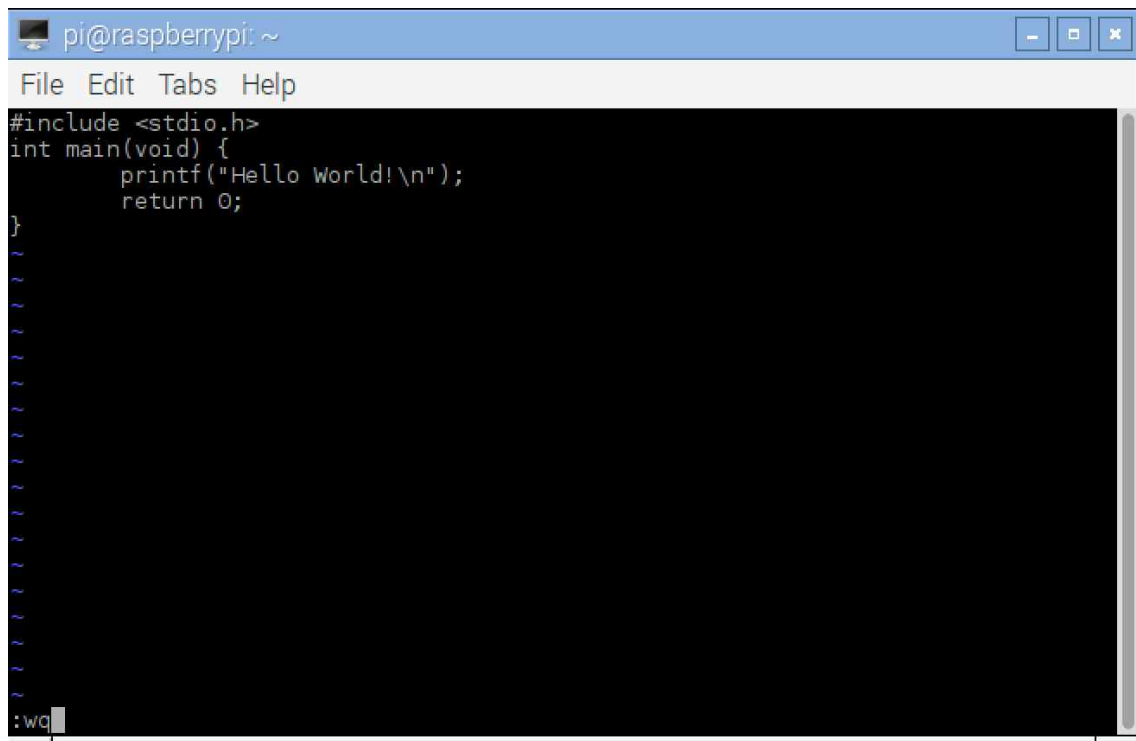
문득 pdf를 따라하며 설정을 하다가 라즈베리파이에 이것저것 연결하지 말고 노트북을 통해 사용해보고 싶다는 생각이 들었습니다. 유튜브를 통해 라즈베리파이가 연결되어 있는 ip주소만 알면 맥북의 원격 모니터링 앱을 통해서 맥북에서도 라즈베리파이를 제어할 수 있다는 것을 알았습니다.



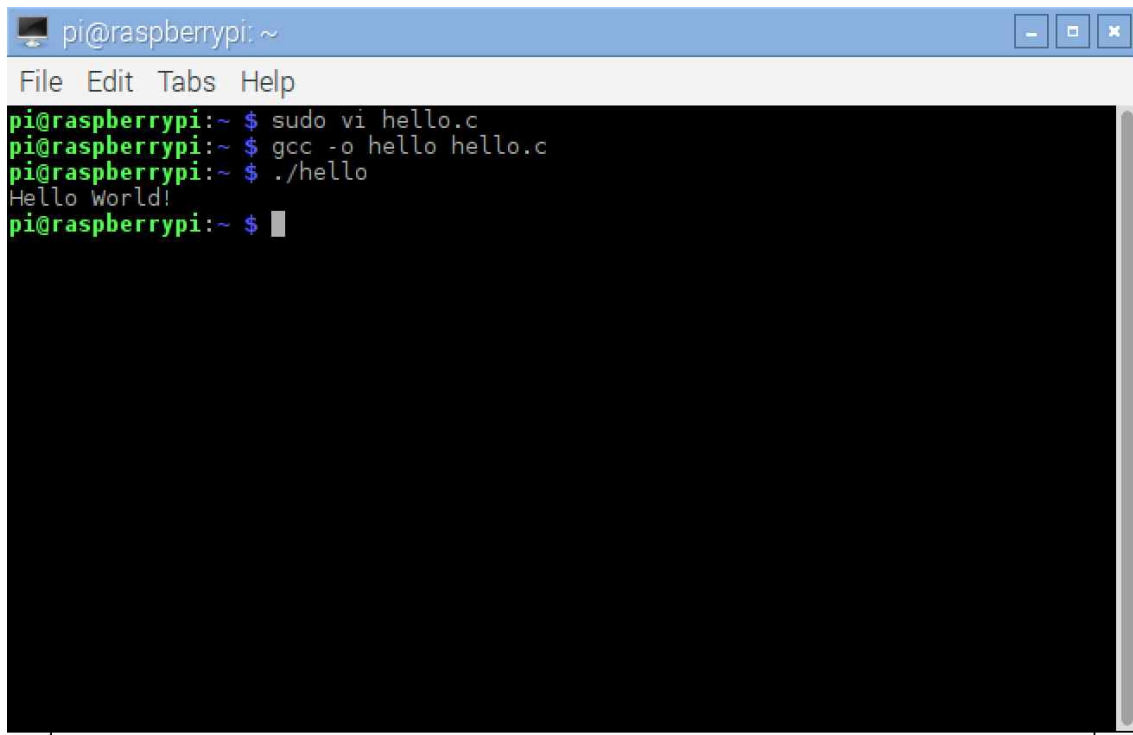
라즈베리파이에 `sudo apt-get install xrdp` 입력 후 맥에서의 원격 모니터링 앱을 통해 라즈베리파이의 주소를 입력하니 위 사진처럼 화면공유가 되었습니다.



<페러럴즈처럼 두가지 os를 다뤄보았습니다.>



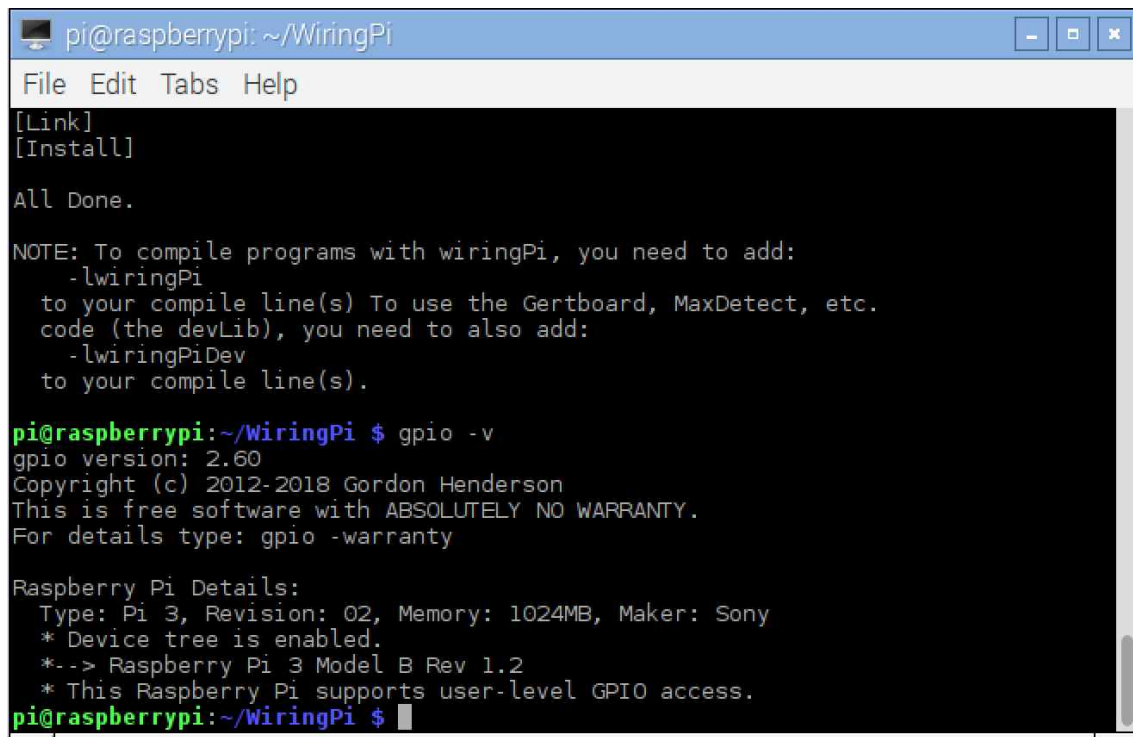
이후 gcc 컴파일러를 통해 라즈비안 os에서의 코딩이 잘 이루어지는 지 테스트를 해보았습니다. vi 에디터를 통해 hello.c 파일을 위 사진처럼 편집하고 저장종료하였습니다.



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~ $ sudo vi hello.c  
pi@raspberrypi:~ $ gcc -o hello hello.c  
pi@raspberrypi:~ $ ./hello  
Hello World!  
pi@raspberrypi:~ $
```

성공적으로 object 파일이 만들어지고 디버깅이 잘 되는 것을 확인할 수 있었습니다.

이후의 실습을 위해 WiringPi를 설치하려 하였으나 pdf 링크의 github의 주소로는 해당 라이브러리를 다운받을 수가 없었습니다. 어떤 오류인지 몰라서 헤메였으나 질문게시판의 김성수 학우가 올려준 다른 방법을 통해 원활하게 설치를 하였습니다.



```
pi@raspberrypi: ~/WiringPi  
File Edit Tabs Help  
[Link]  
[Install]  
  
All Done.  
  
NOTE: To compile programs with wiringPi, you need to add:  
-lwiringPi  
to your compile line(s) To use the Gertboard, MaxDetect, etc.  
code (the devLib), you need to also add:  
-lwiringPiDev  
to your compile line(s).  
  
pi@raspberrypi:~/WiringPi $ gpio -v  
gpio version: 2.60  
Copyright (c) 2012-2018 Gordon Henderson  
This is free software with ABSOLUTELY NO WARRANTY.  
For details type: gpio -warranty  
  
Raspberry Pi Details:  
Type: Pi 3, Revision: 02, Memory: 1024MB, Maker: Sony  
* Device tree is enabled.  
*-> Raspberry Pi 3 Model B Rev 1.2  
* This Raspberry Pi supports user-level GPIO access.  
pi@raspberrypi:~/WiringPi $
```

gpio -v 명령어로 설치가 잘 되어있나 확인하는 사진입니다.

pdf에 나와있는 대로 개발환경을 잘 구축하였고 이 뒤의 실습에서 필요한 사항이 있으면
개선해 나가겠습니다.