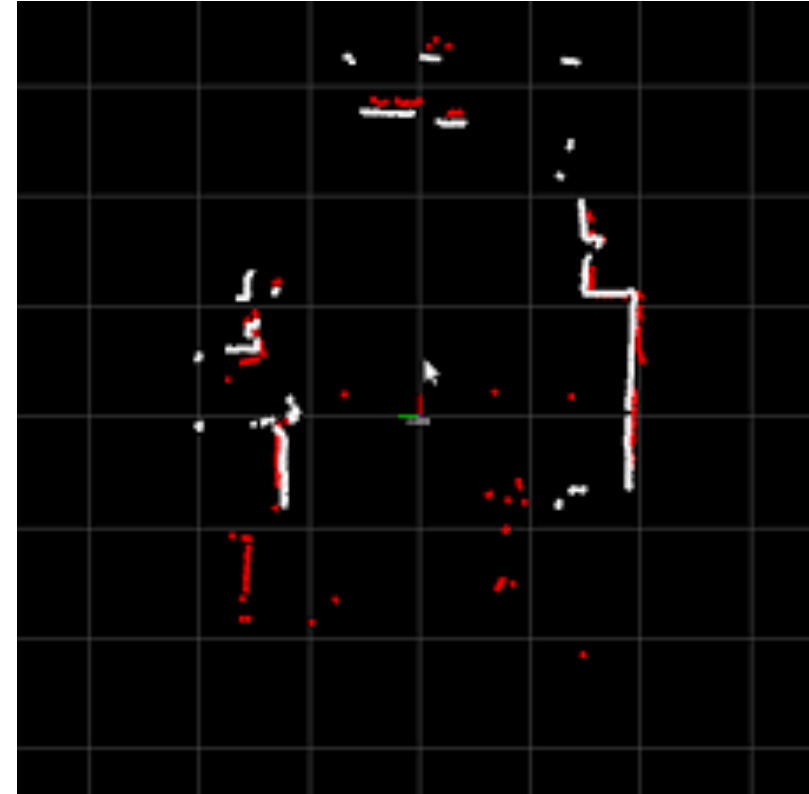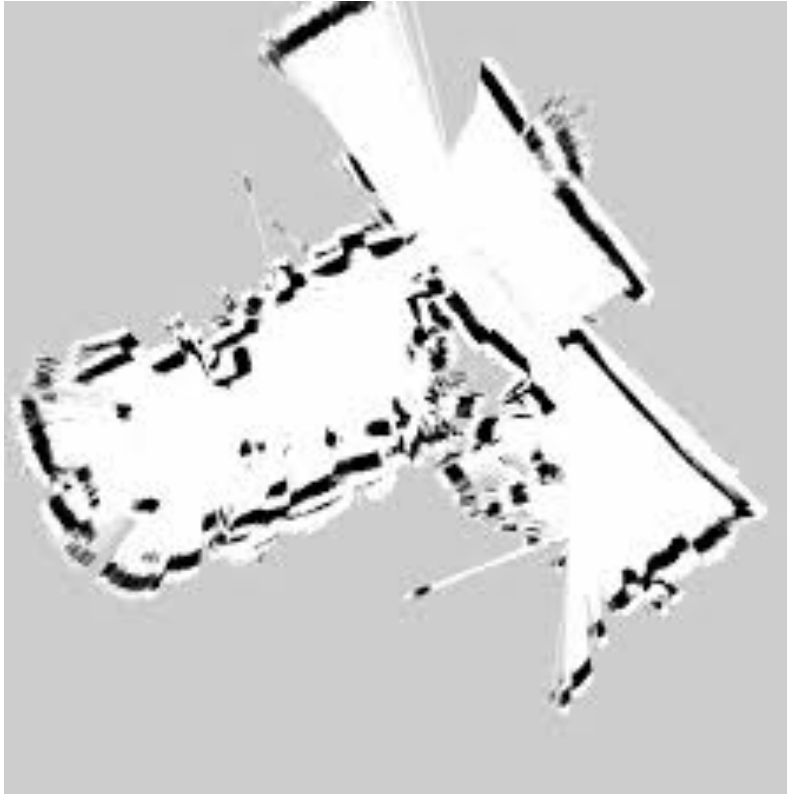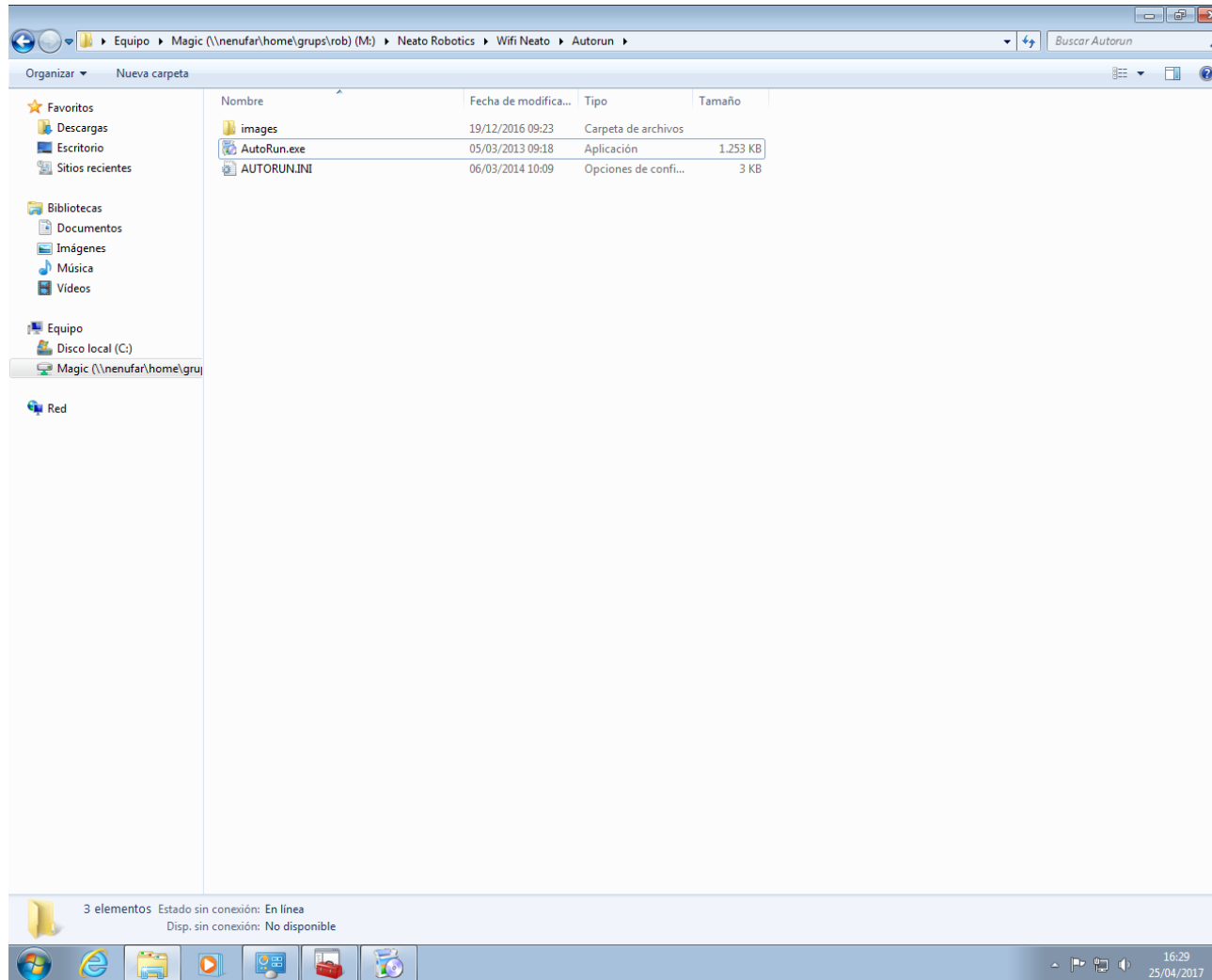# Mobile Robot Lab: NEATO
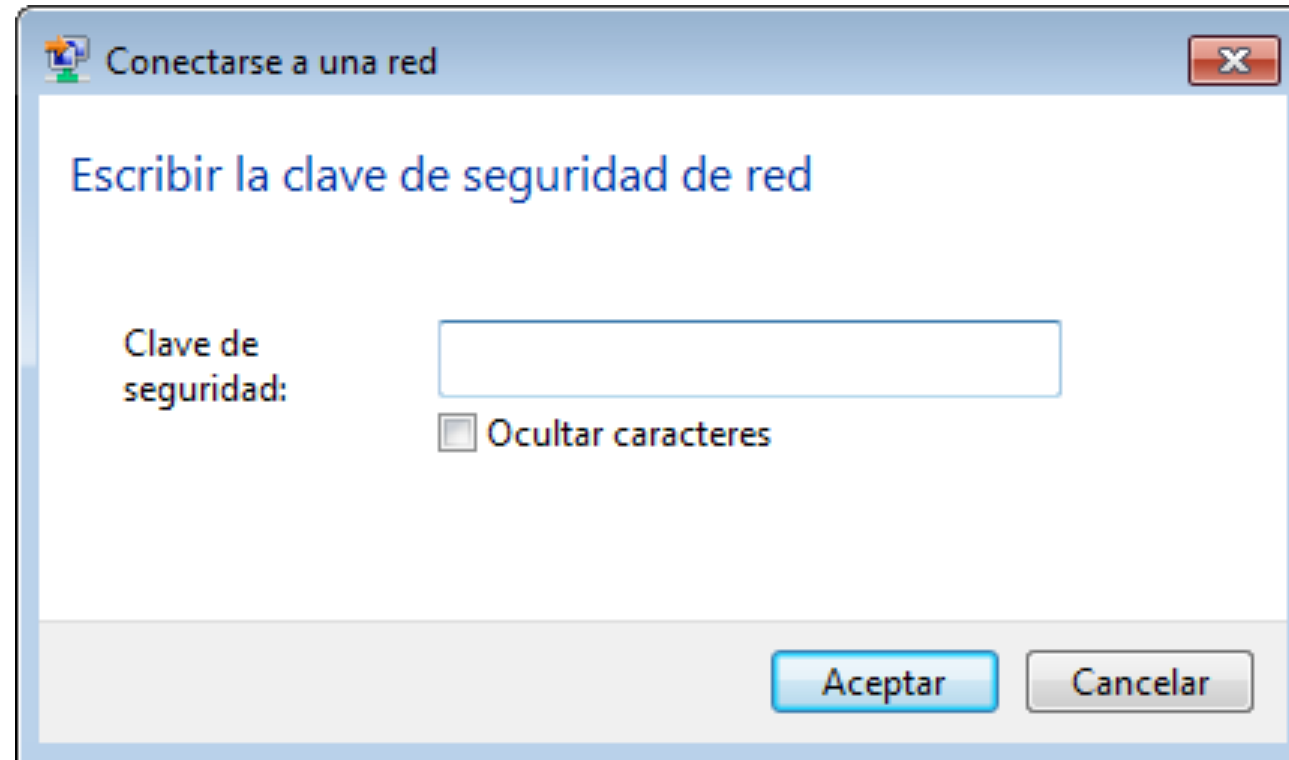
Installing Wifi Drivers

Figure 1 Go to Magic; M:\Neato Robotics\Wifi Neato\Autorun

Conecting to Neato=RaspberryPi
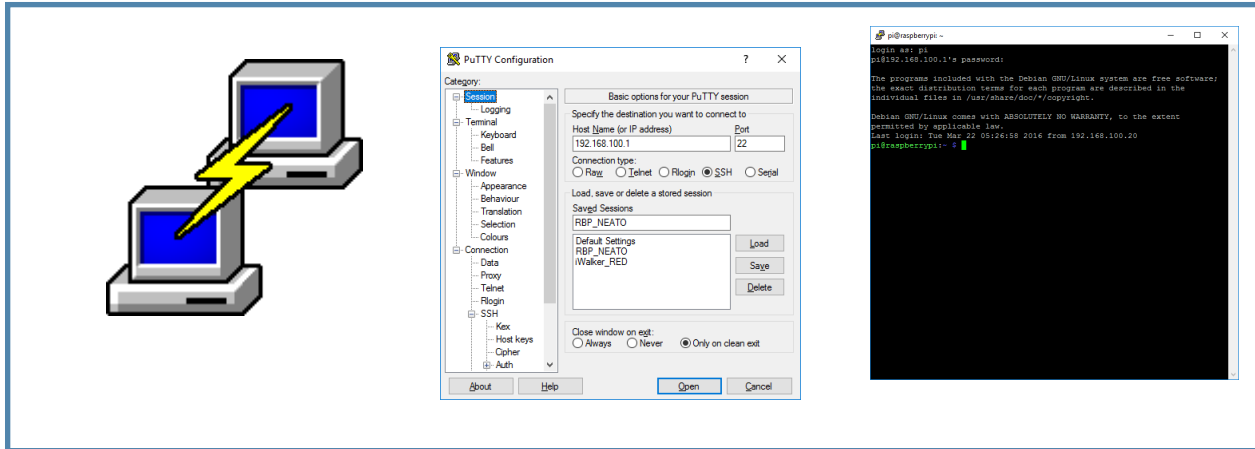

Select the SSID Wifi:
NEATO_{ A B C D E F K }

Password: neato2016
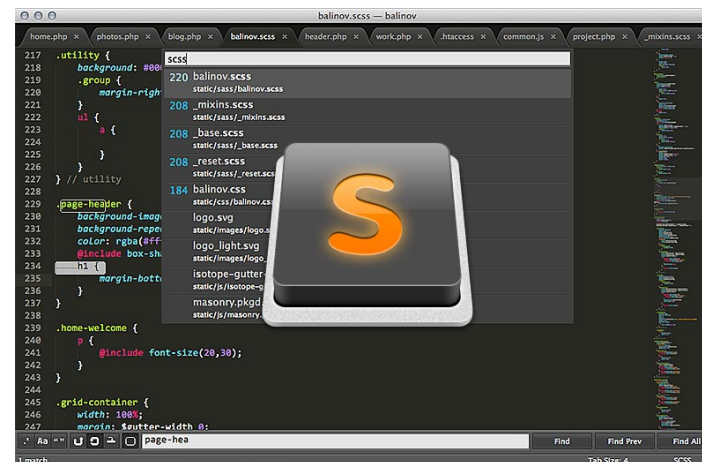
Enviroment

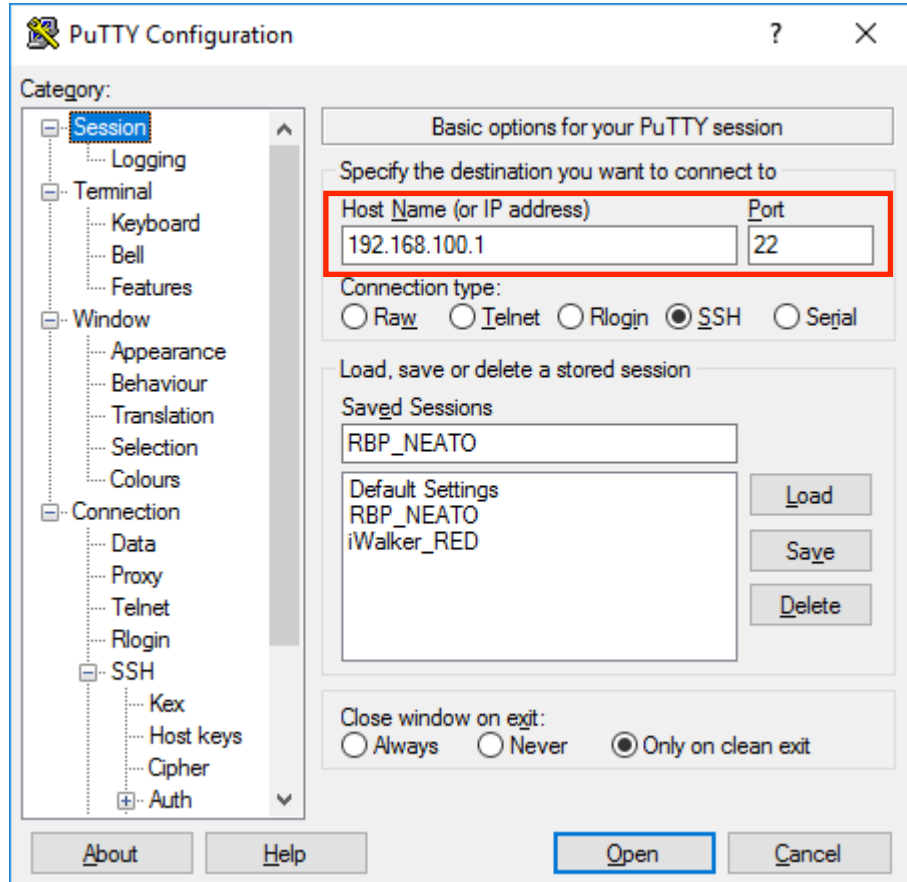Putty → connect remotely to RPI and execute

FileZilla → To transfers files to the RPI

Sublime Text → For editing
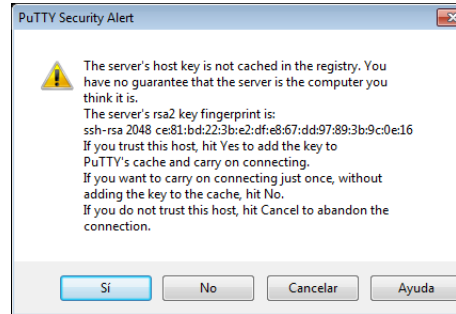
Config Putty (user: pi / pass: raspberry)
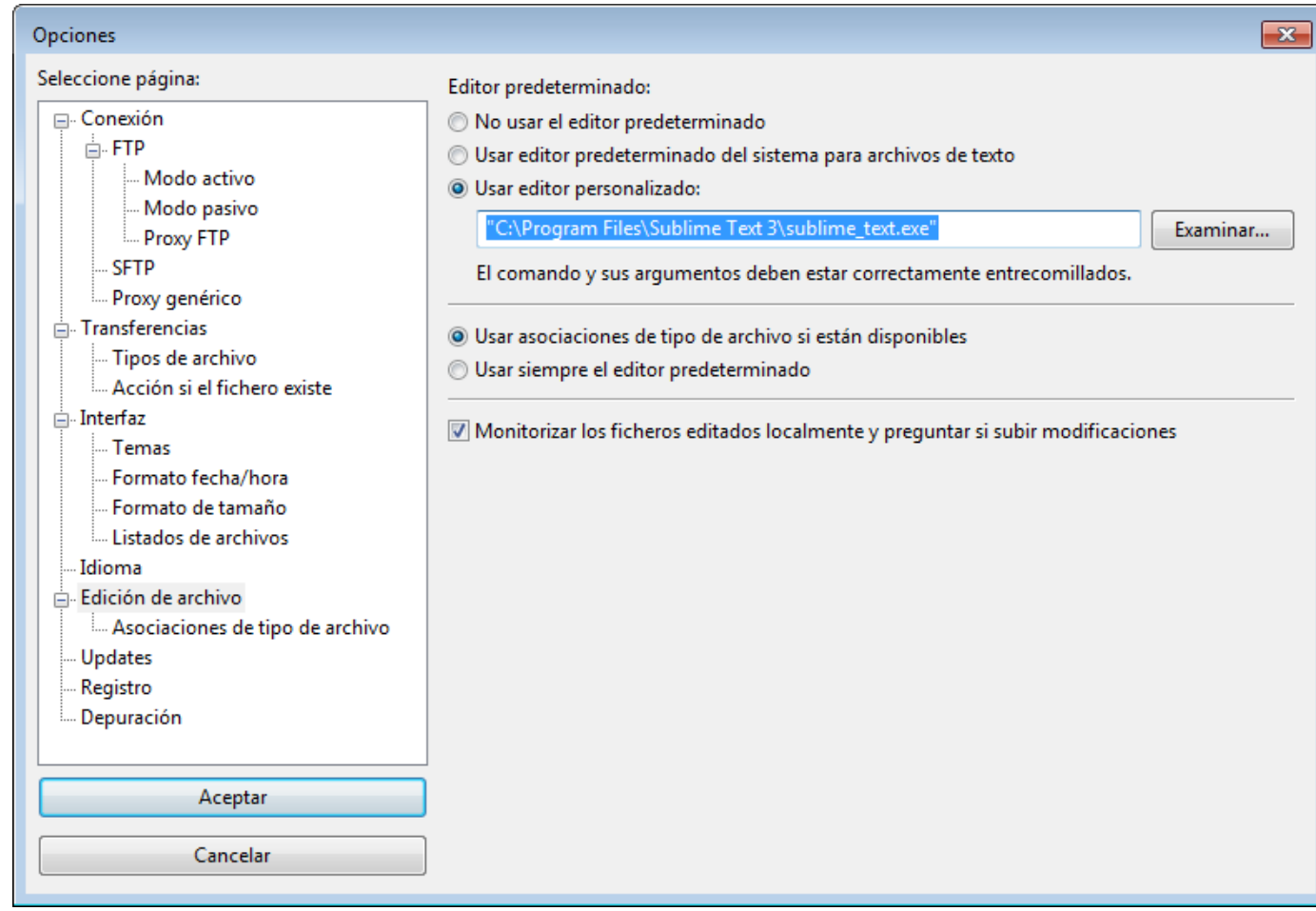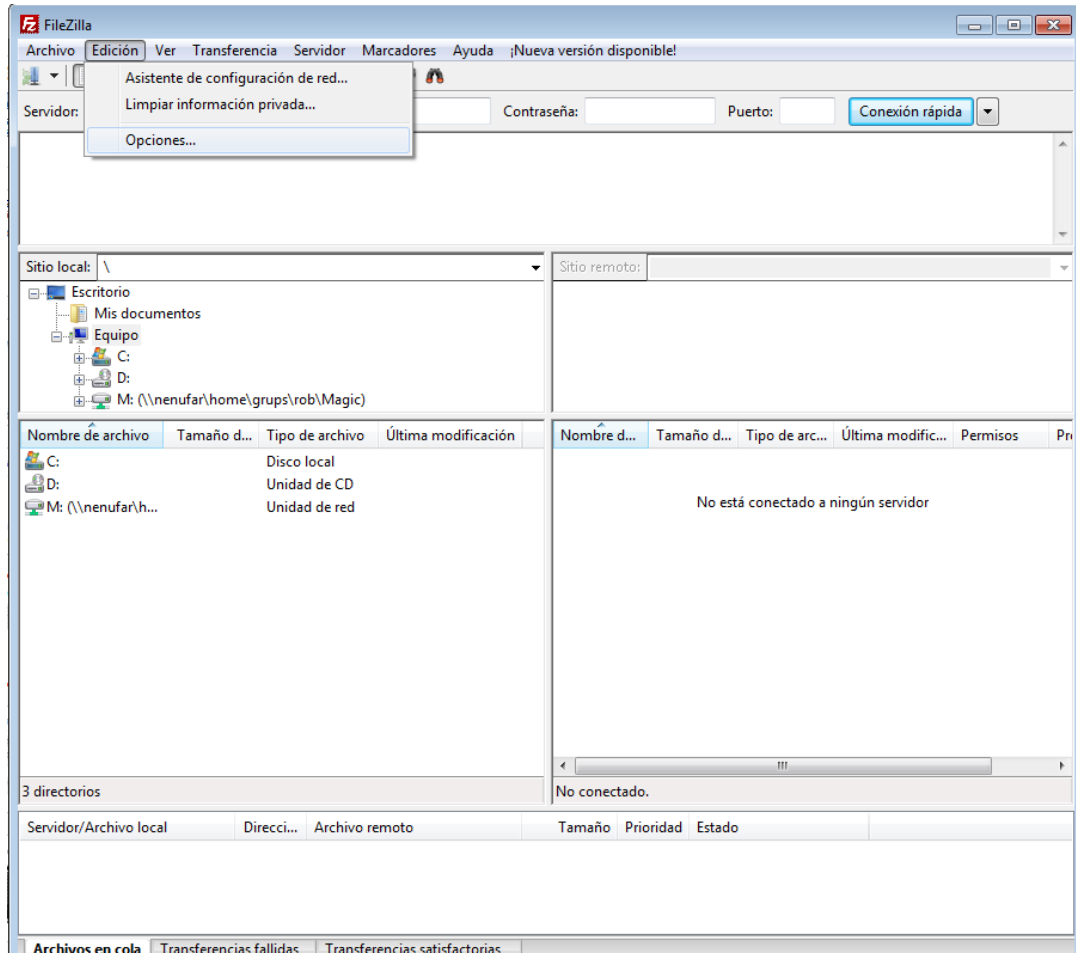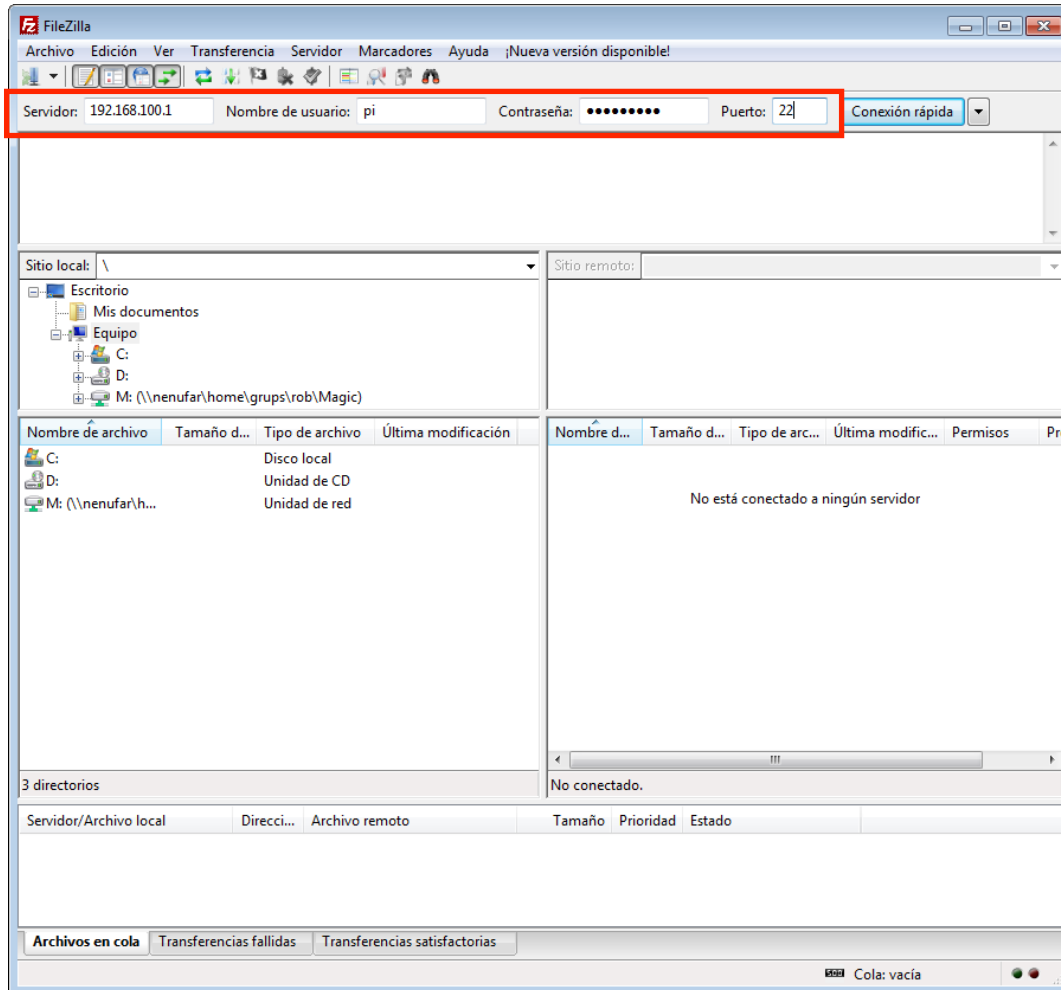

Answer: YES

IP: 192.168.100.1
Port: 22
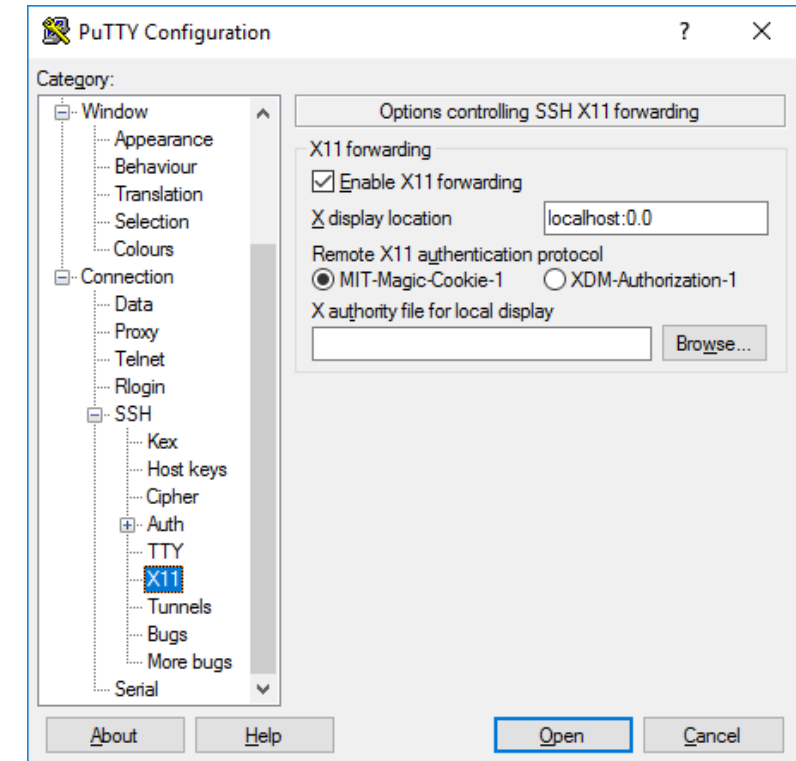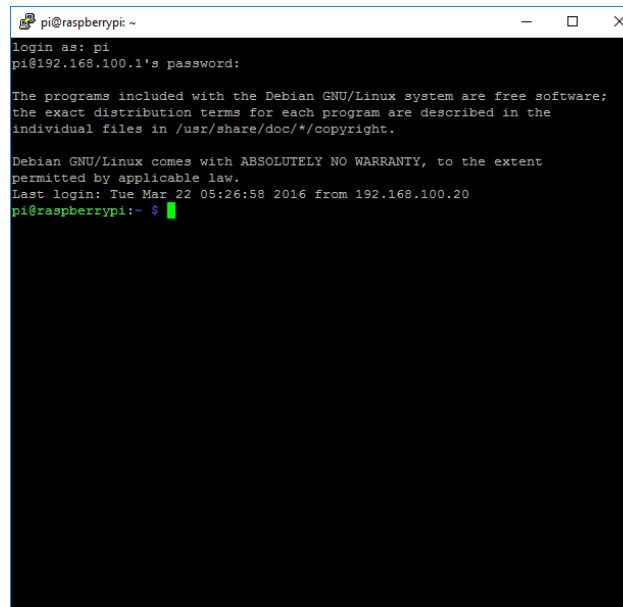
Login as: pi
Psw: raspberry

Configuring Filezilla

Connecting Filezilla to Raspberry

IP: 192.168.100.1 / usr: pi / psw: raspberry / Port: 22

Executing: test_DrivingNR.py and test_NeatoCommands.py

Python test_DrivingNR.py
Python test_NeatoCommands.py

How Neato 'firmware' interpret: set_motor command

Command: SetMotor

Description: Sets the specified motor to run in a direction at a requested speed. (TestMode Only)

Usage: SetMotor [LWheelDist <LWheelDist_value>] [RWheelDist <RWheelDist_value>] [Speed <Speed_value>] [Accel <Accel_value>] [RPM <RPM_value>] [Brush] [VacuumOn] [VacuumOff] [VacuumSpeed <VacuumSpeed_value>] [RWheelDisable] [LWheelDisable] [BrushDisable] [RWheelEnable] [LWheelEnable] [BrushEnable]

Options:

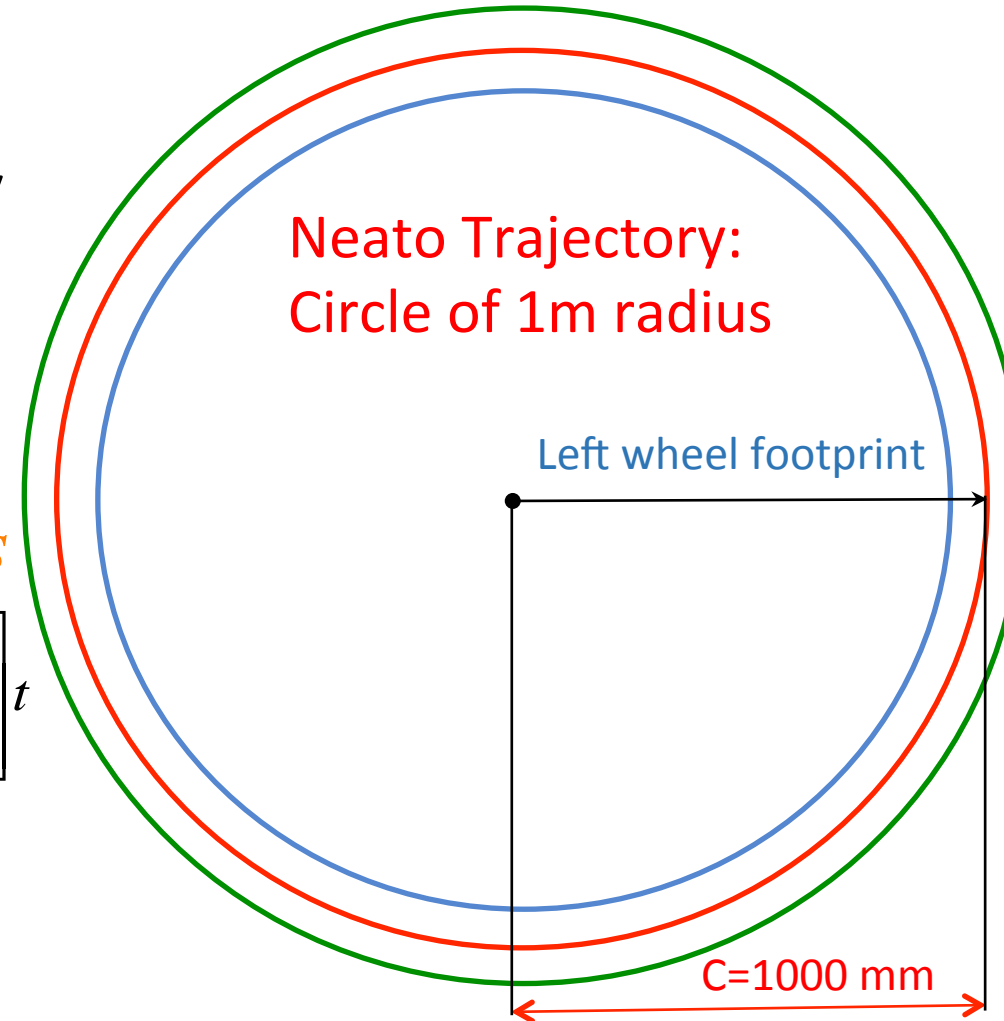| Flag	Description
LWheelDist	Distance in millimeters to drive Left wheel. (Pos = forward, neg = backward)
RWheelDist	Distance in millimeters to drive Right wheel. (Pos = forward, neg = backward)
Speed	Speed in millimeters/second. (Required only for wheel movements)
Accel	Acceleration in millimeters/second. (Used only for wheel movements. Defaults to 'Speed'.)
RPM	Next argument is the RPM of the motor. Not used for wheels, but applied to all other motors specified in the command line.
Brush	Brush motor forward (Mutually exclusive with wheels and vacuum.)

How Neato 'firmware' interpret: set_motor command

Understanding test_DrivingNR.py

$$R = 2\pi\left(1000 + \frac{243}{2}\right) = 7047mm$$

$$L = 2\pi\left(1000 - \frac{243}{2}\right) = 5520mm$$

Neato Trajectory:
Circle of 1m radius

Left wheel footprint

Right wheel footprint

C=1000 mm

2S= 24,3cm

$r_{wheel} = 38.5mm$

S

Inverse Kinematics

$$\begin{bmatrix} \dot{R} \\ \dot{L} \end{bmatrix} t = \underbrace{\begin{bmatrix} 1 & S \\ 1 & -S \end{bmatrix}}_{Jacobian} \begin{bmatrix} v_C \\ \dot{\theta} \end{bmatrix} t$$
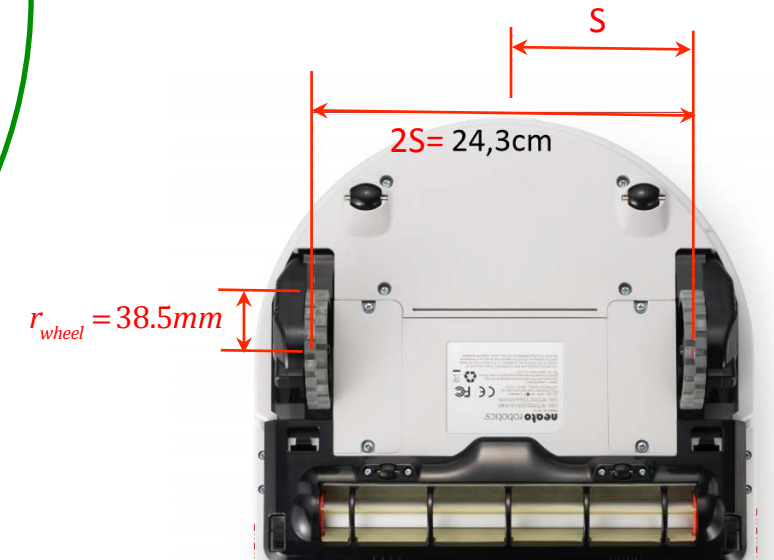
$$\begin{bmatrix} R \\ L \end{bmatrix} = \begin{bmatrix} 1 & S \\ 1 & -S \end{bmatrix}\begin{bmatrix} C \\ \theta \end{bmatrix}$$

```
distancia_R = (((speed * pow(-1, direccion) ) + (S * tita_dot)) * tiempo) * pow(-1, direccion)
distancia_L = (((speed * pow(-1, direccion) ) + (-S * tita_dot)) * tiempo) * pow(-1, direccion)

comando = 'SetMotor LWheelDist ' + str(distancia_L) + ' RWheelDist ' + str(distancia_R) + ' Speed ' + str(speed * pow(-1, direccion))
envia(ser,comando, 0.2)
```