Here is a listing of the complete program for reading a list of IP addresses representing devices, connecting to each via Telnet, extracting the version string, and printing it out. Do not worry about understanding all these details now, you will become more familiar with them as you gain a greater knowledge of Python.

```
#!/usr/bin/python

import re
import pexpect

#-----------------------------------------------------------------------
def get_devices_list():

# - Devices is a file in the same folder as the python file that has the list of IP’s for the devices in question

    devices_list = []
    file = open('devices', 'r')

    for line in file:
        devices_list.append( line.rstrip() )

    file.close()

    print 'devices list:', devices_list
    return devices_list

#-----------------------------------------------------------------------
def connect(ip_address, username, password):

    print 'establishing telnet session:', ip_address, username, password
    telnet_command = 'telnet ' + ip_address

    # Connect via telnet to device
    session = pexpect.spawn('telnet ' + ip_address, timeout=20)
    result = session.expect(['Username:', pexpect.TIMEOUT])

    # Check for error, if so then print error and exit
    if result != 0:
        print '!!! TELNET failed creating session for: ', ip_address
        exit()

    # Enter the username, expect password prompt afterwards
    session.sendline(username)
    result = session.expect(['Password:', pexpect.TIMEOUT])

    # Check for error, if so then print error and exit
    if result != 0:
        print '!!! Username failed: ', username
        exit()

    session.sendline(password)
    result = session.expect(['>', pexpect.TIMEOUT])
    # Check for error, if so then print error and exit
    if result != 0:
        print '!!! Password failed: ', password
        exit()

    print '--- connected to: ', ip_address
    return session

#-----------------------------------------------------------------------
def get_version_info(session):

    print '--- getting version information'

    session.sendline('show version | include Version')
    result = session.expect(['>', pexpect.TIMEOUT])

    # Extract the 'version' part of the output
    version_output_lines = session.before.splitlines()
    version_output_parts = version_output_lines[1].split(',')
    version = version_output_parts[2].strip()

    print '--- got version: ', version
    return version

#-----------------------------------------------------------------------

devices_list = get_devices_list()    # Get list of devices

version_file_out = open('version-info-out', 'w')

# Loop through all the devices in the devices list
for ip_address in devices_list:

    # Connect to the device via CLI and get version information
    session = connect(ip_address, 'cisco', 'cisco')
    device_version = get_version_info(session)

    session.close()  # Close the session

    version_file_out.write('IP: '+ip_address+'  Version: '+device_version+'\n')

# Done with all devices and writing the file, so close
version_file_out.close()
```