

# Raspberry Pi e Raspberry Pi OS

*Resumo:* A aula de hoje consiste em:

- a) Instalar os softwares e fazer funcionar Raspberry Pi.
- b) Fazer Raspberry acessar internet via wifi.
- c) Fazer Raspberry se comunicar com o computador de várias formas.

## 1 Apresentação

[Texto adaptado de: [https://pt.wikipedia.org/wiki/Raspberry\\_Pi](https://pt.wikipedia.org/wiki/Raspberry_Pi) ]

Raspberry Pi é um computador do tamanho de um cartão de crédito, que se conecta a um monitor e usa um teclado e um mouse padrão. Foi desenvolvido no Reino Unido pela Fundação Raspberry Pi. É capaz de fazer tudo o que se esperaria de um computador desktop como navegar na internet; reproduzir vídeo; manipular planilhas; editar textos; compilar e executar programas; e jogar jogos. Além disso, Raspberry pode interagir com o mundo exterior através da conexão GPIO (General Purpose Input/Output). Raspberry tem sido usado numa ampla gama de projetos digitais como sistema embarcado.

Raspberry Pi 3 Model B, lançado em fevereiro de 2016, vem com WiFi e Bluetooth na placa e utiliza um “system on a chip” da Broadcom, que inclui um processador central compatível com ARM e um processador gráfico (GPU, a VideoCore IV). CPU tem 4 núcleos, o seu clock é 1,2 GHz e possui memória RAM de 1 GBytes. Esta é a placa que vamos usar neste ano (2º semestre de 2023). O modelo mais atual desta placa é Raspberry Pi 4 Model B, lançado em junho de 2019. A fundação fornece Raspberry Pi OS (conhecida anteriormente como Raspbian), uma distribuição de Linux baseada em Debian. Este ano, vamos usar o sistema operacional Raspberry Pi OS 10 (Buster with desktop, 32 bits).



Figura 1: Raspberry Pi 3 Model B [do Wikipedia].

Arquitetura de processador ARM (Advanced RISC Machine) é usada principalmente em sistemas embarcados, smartphones e tablets. Quase certamente, o processador do seu celular é ARM. Os processadores ARM são conhecidos pela sua versatilidade, pois possuem poucas instruções. O padrão RISC (Reduced Instruction Set Computer) permite que estes processadores tenham menos transistores que processadores CISC (Complex Instruction Set Computer, como x86). Essa abordagem reduz custo, calor liberado e consumo de energia. Essas são características desejáveis

para dispositivos portáteis. A empresa britânica ARM Holdings desenvolve a arquitetura e a licença para outras empresas que projetam seus próprios produtos.

O site abaixo traz informações sobre o conjunto de instruções e a linguagem assembly de ARM:  
<https://azeria-labs.com/writing-arm-assembly-part-1/>

Nos meus testes “grosseiros”, Intel i7 era umas 6 vezes mais rápido que Raspberry Pi 3B em programas sequenciais (não-paralelas).

Nota 1: Intel i7 é aproximadamente 2 vezes mais rápido que Raspberry Pi 4B em programas sequenciais.

Nota 2: No mercado, há outras placas semelhantes à Raspberry Pi, entre elas: Banana Pi, Odroid e Orange Pi.

Nota 3: O programa *hardinfo* pode ser usado para calcular vários tipos de benchmarks. Eu testei usando programa *bench.cpp*.

## 2 Instalação do sistema Raspberry Pi OS

### 2.1 Introdução

[Olhar o site de Raspberry e descrever como instalar sistema sem monitor e teclado]

Raspberry Pi OS é um sistema operacional “open source” baseado em Debian e otimizado para Raspberry. Antes, chamava-se Raspbian. A forma mais simples de instalar Raspberry Pi OS 10 é usando Raspberry Pi Imager, que pode ser baixado do site acima.

<https://www.raspberrypi.org/software/>

Vamos usar a versão “Buster with desktop, 32 bits” de Raspberry Pi OS (versão ligeiramente antiga) para ficar compatível com o resto do sistema. Você pode usar uma versão recente do sistema, mas neste caso você mesmo deve resolver os problemas de incompatibilidade. Você pode ver a versão do sistema com o comando:

```
$ cat /etc/os-release
```

A documentação da Raspberry e do seu sistema operacional é bem didática e está em:

<https://www.raspberrypi.org/documentation/>

A página abaixo explica como começar a usar Raspberry:

<https://www.raspberrypi.com/documentation/computers/getting-started.html>

Depois que SD card estiver gravado com o sistema operacional, você deve inseri-lo no Raspberry, e ligar Raspberry a um monitor (ou TV, via HDMI ou DVI), um teclado e um mouse. Depois, deve ligar Raspberry à fonte de alimentação 5V, de pelo menos 2,5A. Ligando a força, deve aparecer na tela do monitor o sistema operacional do Raspberry. Depois, conecte Raspberry à internet configurando a conexão sem fio.

*Cuidado:* Sempre desligue a energia de Raspberry “de forma ordenada”, clicando em “shutdown” no ambiente gráfico ou “sudo poweroff” na linha de comando e esperando até que o monitor fique preto indicando que Raspberry se desligou completamente. O cartão SD pode ficar corrompido se desligar Raspberry durante uma operação de escrita. Ao ligar Raspberry, ligue de uma vez, sem provocar liga-desliga intermitente. Para ligar novamente Raspberry que foi desligado, o único jeito é desligar a alimentação, esperar um pouco e ligar novamente. Se você desconfiar que Raspberry está com o cartão de memória corrompido (por exemplo, desligou Raspberry de forma não ordenada), o seguinte comando checará o cartão na próxima vez que der boot:

```
$ sudo touch /forcefsck  
$ reboot
```

## 2.2 Instruções especiais para PSI3422, 2º semestre de 2023

Este ano (2º semestre de 2023), Maurício já instalou “Raspberry Pi OS 10” no SD card entregue no kit didático, juntamente com o Raspberry. O usuário e a senha do sistema estão em edisciplinas:

<https://edisciplinas.usp.br/course/view.php?id=111037&section=9#tabs-tree-start>

Além disso, Raspberry já está programado para se conectar à rede GDLAB da sala GD-06 e habilitar servidor de VNC (Virtual Network Computing) para trabalhar remotamente sem monitor, teclado ou mouse. Basta ligar Raspberry na sala GD-06 que ele vai se conectar automaticamente à rede GDLAB e ficar pronto para ser acessado remotamente através de VNC. Para acessá-los remotamente de um computador desktop ou do seu notebook, é necessário conhecer o endereço IP (Internet Protocol) do Raspberry. Os endereços IP os números MAC (Media Access Control) dos Raspberries entregues junto com os kits estão em:

<https://edisciplinas.usp.br/course/view.php?id=111037&section=9#tabs-tree-start>

Nesta apostila mais adiante, há mais informações sobre VNC, IP e MAC.

Se você quiser trabalhar fora da sala GD-06, você precisará configurar manualmente o Raspberry e o seu computador.

[Se você já conhece bem o sistema Linux, pode pular as seções 2.3-2.8 e ir direto para a seção 3.]

## 2.3 Os principais comandos de Linux

Pratique várias características de Linux para se familiarizar. Procure aprender e praticar pelo menos o que os seguintes comandos fazem:

ls, cd, pwd, mkdir, rmdir, rm, cp, mv, cat, chmod, ssh, scp, sudo, df, du, zip, unzip, pipe (<, >, >>, |), &, man, find, whereis, ps, kill.

*Nota para mim:* Escrever exemplos de uso do pipe.

## 2.4 Atualização

Para atualizar o sistema [<https://www.raspberrypi.com/documentation/computers/os.html> ]:

```
sudo apt update
sudo apt full-upgrade
```

Sugiro que atualize o sistema após a instalação. Site do Raspberry diz que kernel e firmware também serão atualizadas pelos dois comandos acima.

## 2.5 Instalar um programa

Um aplicativo pode ser instalado usando o programa “início → preferences → add/remove software” ou “início → preferences → recommended software”.

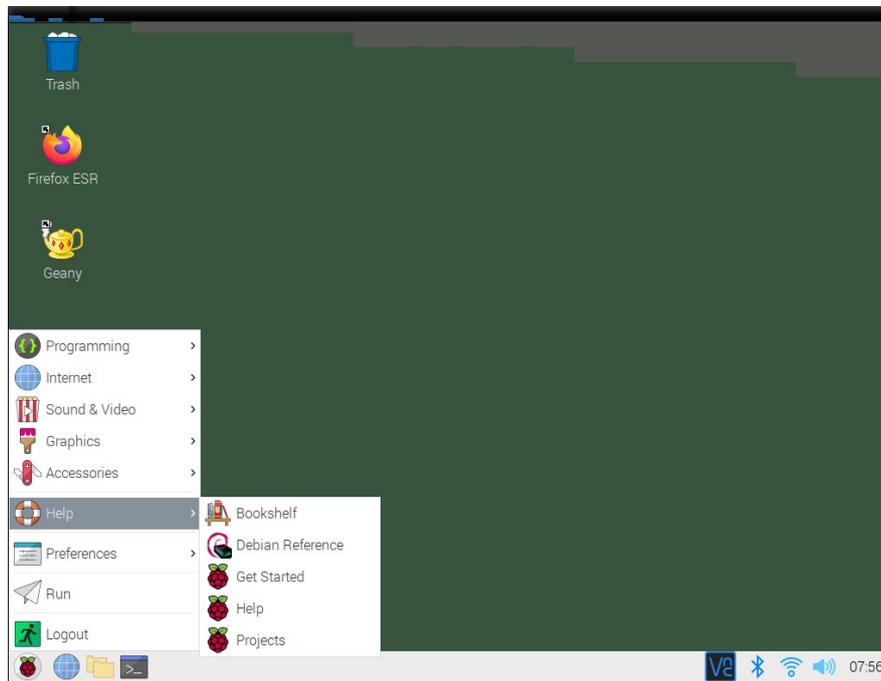


Figura 2: Menu para instalar um programa em Raspberry.

Também pode ser instalado pela linha de comando “apt” ou “apt-get” dentro do terminal. Instale um aplicativo usando “sudo apt install programa” (apt e apt-get são praticamente iguais), por exemplo:

```
$ sudo apt install zip (compactador de arquivos)
$ sudo apt install firefox-esr (Browser Firefox)
$ sudo apt install smplyer (Visualizador de vídeo)
```

Um aplicativo pode ser removido usando:

```
$ sudo apt remove programa
```

Se quiser apagar também os arquivos de configuração, use:

```
$ sudo apt purge programa
```

## 2.6 Editor de texto

Existem vários editores de texto “GUI”. Você pode usar o que gostar mais para editar o seu programa:

geany, leafpad, idle (do python), scite, gedit, xedit, etc.

Raspberry já vem com geany instalado. Também existem editores de linha de comando (úteis quando, por algum problema, não dispuser de um ambiente gráfico):

nano, vi, vim, emacs, etc.

Editor *nano* é bem fácil de usar.

## 2.7 Variáveis de ambiente

Uma variável de ambiente é uma variável do sistema operacional que contém informações sobre como um processo se comporta. Tanto Linux como Windows utilizam variáveis de ambiente.

No Linux, você pode listar todas as variáveis com:

```
$ printenv  
$ printenv | sort [para imprimir em ordem alfabética]
```

Você pode ver o conteúdo de uma variável de ambiente usando:

```
$ echo $NOMEVAR
```

Exemplos de variáveis de ambiente:

a) Variável HOME contém o diretório raiz do usuário.

*Exemplo:* `$ echo $HOME`

b) Variável PATH contém a lista de diretórios onde pode estar um arquivo executável. No Linux, os diretórios são separados por dois pontos ":".

*Exemplo:* `$ echo $PATH`

c) Em Linux, LD\_LIBRARY\_PATH indica a lista de diretórios onde pode estar bibliotecas compartilhadas (\*.so).

d) O compilador GCC/G++ usa as variáveis de ambiente (tanto no Linux como no Windows):

- LIBRARY\_PATH: Lista de diretórios onde procurar biblioteca durante a linkagem.
- C\_INCLUDE\_PATH: Lista de diretórios onde procurar os arquivos include (\*.h) da linguagem C (Ex: #include <stdio.h>).
- CPLUS\_INCLUDE\_PATH: Lista de diretórios onde procurar os arquivos include da linguagem C++ (Ex: #include <iostream>).

## 2.8 Arquivo .bashrc

No diretório HOME dos sistemas derivados de Debian (Mint, Ubuntu e Raspberry Pi OS) fica o arquivo .bashrc que é executado toda vez que terminal de comando é inicializado. Esse arquivo não aparece na lista de arquivos, pois arquivos que começam com ponto são considerados "arquivos escondidos". Você deve escrever "ls -a" para visualizar arquivos escondidos.

Você pode mudar algumas configurações em .bashrc para deixar o sistema de acordo com as suas preferências. Por exemplo, colocando a linha

```
alias del='rm'
```

nesse arquivo, você poderá escrever "del" no lugar de "rm". Se você acrescentar o comando:

```
export PATH=".:$PATH"
```

podrá escrever "programa" em vez de "./programa" para executar o programa que se encontra no diretório default ".".

[Se você já conhece bem o sistema Linux, pode pular até aqui.]

### 3 Instalar Linux no computador

Para fazer este projeto, você deve ter um computador com sistema Linux (Mint ou Ubuntu). Seria possível fazer o projeto em Windows, mas vai dar bastante trabalho pois os comandos para acessar a Internet serão diferentes em Linux (Raspberry) e Windows (computador). Assim, teria que desenvolver dois programas completamente diferentes.

Se o seu computador só tem o sistema Windows, o site abaixo explica detalhadamente como instalar Linux Mint dentro do VirtualBox em Windows.

[http://www.lme.usp.br/~mauricio/linux\\_psi/](http://www.lme.usp.br/~mauricio/linux_psi/)

Entre no site abaixo usando usuário/senha disponíveis em edisciplinas.

<https://edisciplinas.usp.br/course/view.php?id=111037&section=9#tabs-tree-start>

Configurando assim, Linux Mint vai rodar sobre o sistema Windows, dentro de uma janela (figura 3) ou ocupando a tela toda.

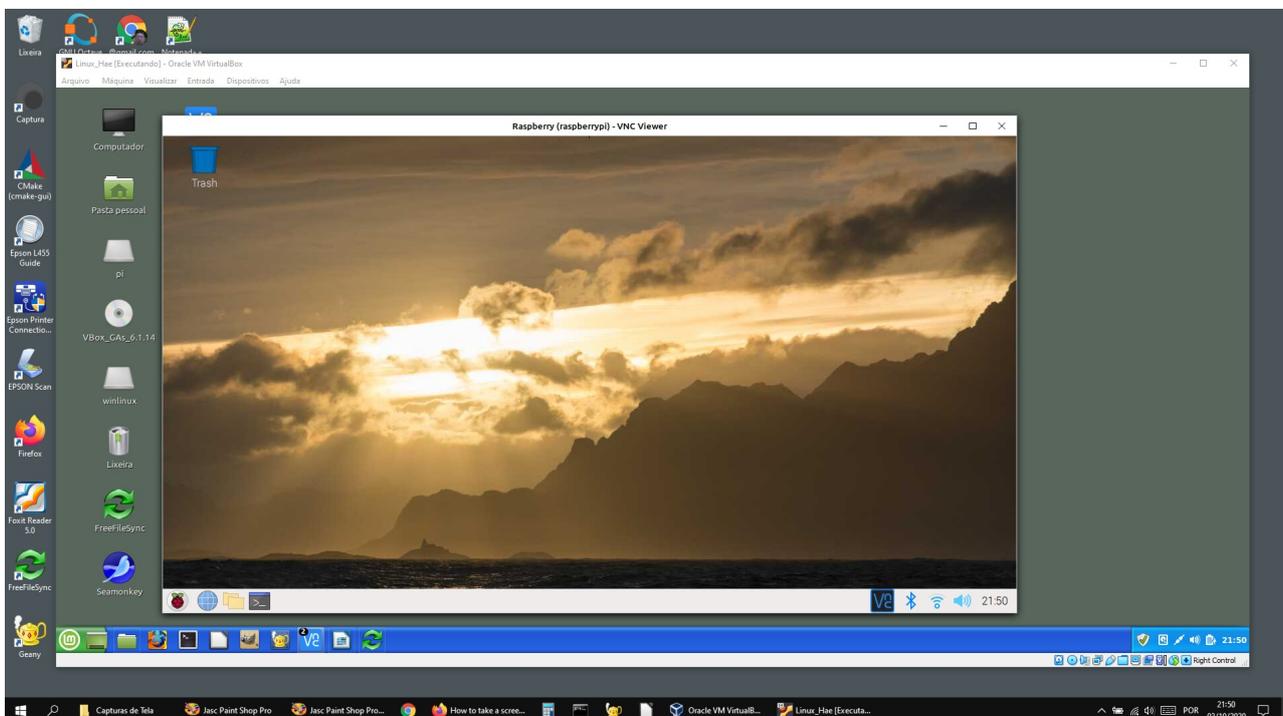


Figura 3: Linux Mint está rodando dentro de uma janela do sistema operacional Windows através do VirtualBox. A tela do Raspberry aparece como uma janela do Linux Mint através de VNC.

## 4 Controlar Raspberry remotamente

[Olhar o site de Raspberry e descrever como instalar sistema sem monitor e teclado. Descrever como detecta IP de Raspberry remotamente. nmap.]

*PSI3422, 2º semestre de 2023: O Raspberry do kit que você recebeu para já vem configurado para ser acessado remotamente na sala GD-06. Conectando Raspberry na rede GDLAB da sala GD-06, os números IP dos Raspberries serão fixos, conforme:*

<https://edisciplinas.usp.br/course/view.php?id=111037&section=9#tabs-tree-start>.

### 4.1 Números IP e MAC

Depois que Raspberry estiver completamente configurado (incluindo acesso à Internet via wifi), será possível controlá-lo remotamente a partir de um computador, sem que seja necessário ligá-lo a monitor/teclado/mouse (modo “headless”). Porém, para configurá-lo, será necessário primeiro ligar Raspberry num monitor/teclado/mouse.

A documentação de Raspberry Pi explica bem como controlá-lo remotamente.

<https://www.raspberrypi.com/documentation/computers/remote-access.html>

Descrevo abaixo os passos essenciais.

Primeiro, habilite a câmera e as comunicações SSH e VNC. Sem isso, será impossível controlá-lo remotamente: a comunicação e a câmera não vão funcionar “misteriosamente” e Raspberry não vai explicar por que não está funcionando. Vá para:

Início → preferences → Raspberry Pi Configuration

E habilite câmera, SSH e VNC (os 3 primeiros itens da figura 4).

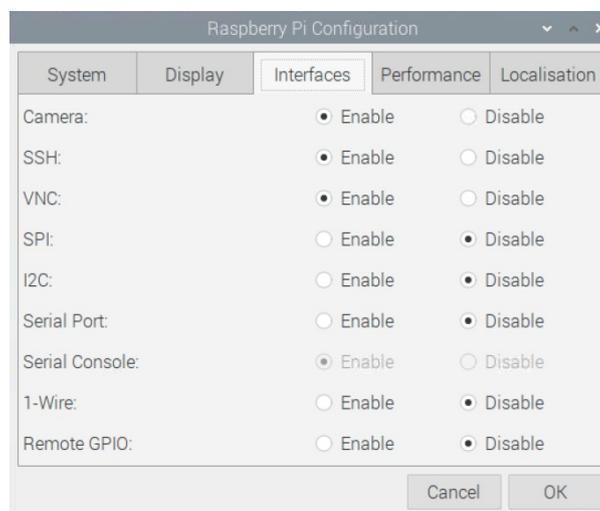


Figura 4: Configurações do Raspberry para acessar câmera, SSH e VNC.

Você vai precisar do número IP (Internet Protocol) do Raspberry para acessá-lo remotamente. O problema é que o número IP do Raspberry pode mudar toda vez que desligá-lo e ligá-lo novamente à internet. Assim, precisa achar uma forma de descobrir o número IP do seu Raspberry sem ter o trabalho de ligá-lo a um monitor/teclado.

## 1) Descobrir remotamente o endereço IP do seu Raspberry

Há programas que conseguem achar o número IP do Raspberry a partir de um computador ou celular ligado no mesmo roteador de Raspberry. Porém, como no laboratório vai ter dezenas de Raspberrys ligados ao mesmo roteador, você precisa conseguir identificar qual deles é o seu Raspberry. Para isso, vamos usar o endereço MAC (Media Access Control).

“O endereço MAC (Media Access Control ou Controle de Acesso de Mídia) é um endereço físico e único, que é associado à interfaces de comunicação utilizadas em dispositivos de rede. A identificação é gravada em hardware por fabricantes de placa de rede, tornando-se posteriormente, parte de equipamentos como computadores, roteadores, smartphones, tablets, impressoras de rede e diversos outros equipamento que usam comunicação em rede. Como a identificação é única, ela é usada para fazer o “controle de acesso” em diversos tipos de redes de computadores.”

<https://www.techtudo.com.br/dicas-e-tutoriais/noticia/2014/11/o-que-e-endereco-mac-saiba-como-encontrar.html>

Para descobrir o número IP e MAC do Raspberry, com Raspberry conectado a monitor / teclado / mouse, digite no terminal do Raspberry:

```
$ ifconfig
```

*ifconfig* (abreviação de interface config) é um utilitário de administração de sistemas Linux para configuração de interface de rede. O Raspberry deve responder algo como:

```
eth0: flags=(...)  
      ether b9:24:ea:cb:df:35 (...)  
      (...)  
  
lo: flags=(...)  
     inet 127.0.0.1 netmask 255.0.0.0  
     inet6 ::1 prefixlen 128 scopeid 0x10<host>  
     (...)  
  
wlan0: flags=(...)  
       inet 192.168.0.110 netmask 255.255.255.0 broadcast 192.168.0.255  
       inet6 fe81::882e:d6c2:9a12:9166 (...)  
       ether b9:28:ec:9b:80:65 (...)  
       (...)
```

Figura 5: Exemplo de saída do ifconfig.

Raspberry tem 3 dispositivos de comunicação: a placa de rede cabeada (eth0), o dispositivo de loopback local (lo) e a placa de rede wifi sem fio (wlan0). Raspberry está conectado à Internet via wifi, assim o endereço IP está atribuído à placa sem fio (wlan0). No exemplo da figura 5, o endereço IP do Raspberry é 192.168.0.110 e o endereço MAC da placa wifi é b9:28:ec:9b:80:65. Anote esses dois números. Conhecendo endereço MAC (que é fixo), é possível descobrir o número IP do seu Raspberry (necessário para acessá-lo remotamente mas que pode ficar mudando).

### a) Fing do celular

A partir do celular (Android ou iOS) conectado no mesmo roteador do Raspberry, use o aplicativo Fing para descobrir o endereço IP do Raspberry.

### b) Nmap do computador (network scanning)

<https://www.tecmint.com/nmap-command-examples/>

A partir do computador conectado no mesmo roteador do Raspberry, execute o programa *nmap*. Se o programa *nmap* não estiver instalado no seu computador Linux, instale-o:

```
$ sudo apt-get install nmap
```

Vamos supor que os endereços IP do Raspberry esteja no intervalo de 192.168.0.0 a 255. Para escanear esses endereços para verificar se é endereço de algum Raspberry:

```
$ sudo nmap -sn 192.168.0.*          ou
$ sudo nmap -sn 192.168.0.0-255
```

## 2) Fixar o número IP do seu Raspberry

O número IP do seu computador e do Raspberry podem mudar toda vez que se conecta ao roteador. Para facilitar o acesso remoto, é conveniente que os números IP permaneçam constantes. Para isso, deve programar o roteador, acessando <http://192.168.0.1/> com um navegador de internet ou através do aplicativo próprio do roteador. A forma de configurar depende do modelo do roteador. Estou usando 192.168.0.110 (ou 192.168.68.110) para meu Raspberry e 192.168.0.109 (ou 192.168.68.109) para meu notebook. É possível configurar o roteador para manter o número IP do seu Raspberry fixo. Tente aprender o que são IP interno (local ou privado) e IP externo (público). A figura 6 mostra as telas para fixar os números IP de roteador D-link (esquerda) e TP-link Deco (direita).

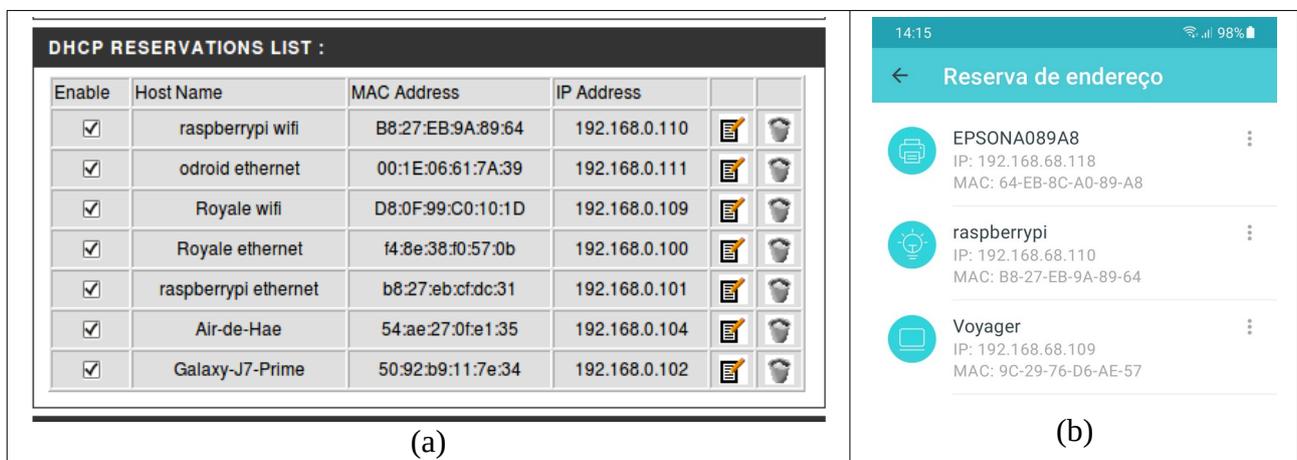


Figura 6: (a) Reserva de número IP do roteador D-link. (b) Reserva de número IP no TP-link Deco.

*PSI3422, 2º semestre de 2023: Se o número IP do seu Raspberry estará fixo se Raspberry se conectar à rede GDLAB da sala GD-06.*

## 4.2 Acesso remoto a Raspberry

Podemos classificar o acesso remoto a Raspberry em três tipos principais:

**Tipo A)** Acessar o sistema gráfico do Raspberry no computador. Ex: VNC.

**Tipo B)** Acessar o terminal do Raspberry. Neste acesso, um terminal Linux (janela de comando ou bash) do Raspberry é aberto no computador remoto e é possível executar os programas do Raspberry remotamente no computador. Ex: ssh.

**Tipo C)** Acessar e copiar os arquivos de/para Raspberry. Neste tipo de acesso, não é possível executar remotamente os programas no Raspberry. Ex: sshfs, scp, rsync, FreeFileSync, Filezilla, etc.

**Tipo A)** VNC (Virtual Network Computing). É possível acessar remotamente o desktop do raspberry a partir do computador. Leia a documentação da Raspberry a respeito:

<https://www.raspberrypi.com/documentation/computers/remote-access.html>

A figura 7 mostra VNC funcionando. A tela do Raspberry aparece como uma janela na tela do computador.

*Nota:* Use “VNC Viewer” de RealVNC no computador para acessar Raspberry (link abaixo). Parece que TightVNC e outros programas VNC não funcionam com Raspberry.

<https://www.realvnc.com/en/connect/download/viewer/>

*PSI3422, 2º semestre de 2023: Cliente VNC já está instalado nos computadores da sala GD-06 e servidor VNC já está instalado e habilitado nos Raspberries.*

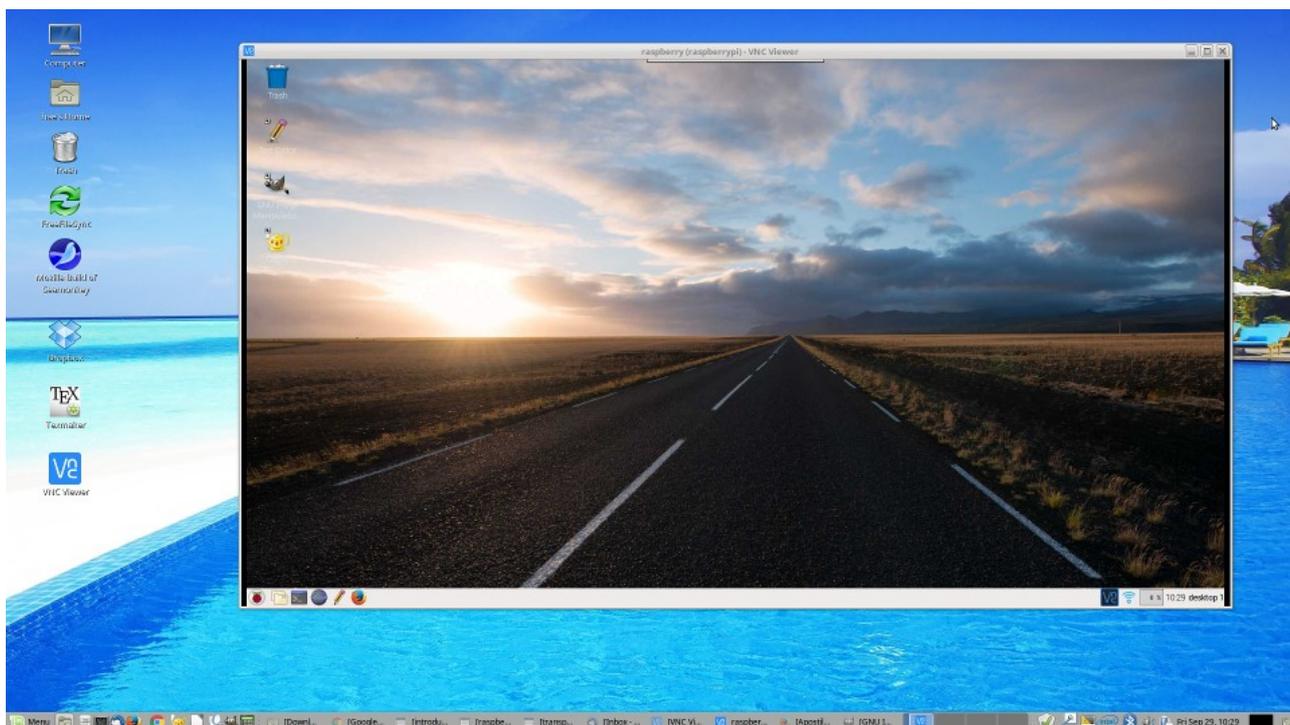


Figura 7: Controlar Raspberry remotamente através de VNC. A tela do Raspberry aparece como uma janela no computador.

*Exercício:* Mostre VNC funcionando. Isto é, mostre uma tela como a da figura 7.

*Exercício opcional:* Como é possível fazer funcionar VNC quando o computador e o Raspberry não estão conectados ao mesmo roteador? Dica: “NAT” – Network Address Translation. Porém, usar NAT pode deixar Raspberry mais vulnerável a ataques de hackers.

**Tipo B)** Também é possível acessar o terminal de comandos do Raspberry através de SSH (secure shell):

<https://www.raspberrypi.com/documentation/computers/remote-access.html>

É possível acessar remotamente o terminal do Raspberry tanto de Linux como de Windows. O comando seguinte permite o acesso (troque os números pelo número IP do seu Raspberry):

```
$ ssh pi@192.168.0.110
```

É possível rodar remotamente os programas GUI (isto é, programas com janela ou Graphical User Interface) do Raspberry e transmitir a janela do aplicativo para o computador através das opções `-X` ou `-X -C`:

```
$ ssh -X pi@192.168.0.110
```

```
$ ssh -X -C pi@192.168.0.110 (faz compactação antes de transmitir)
```

Depois de estabelecida a conexão “ssh -X”, é possível enxergar no computador as janelas dos programas gráficos do Raspberry, chamando-os da linha de comando do Raspberry. Por exemplo:

```
pi@raspberrypi$ geany & => Vai rodar geany no Raspberry e transmitir a janela para o Computador
```

```
pi@raspberrypi$ xeyes & => Vai rodar xeyes no Raspberry e transmitir a janela para o Computador
```

*Nota:* "xeyes" pode ser instalada com "sudo apt install x11-apps"

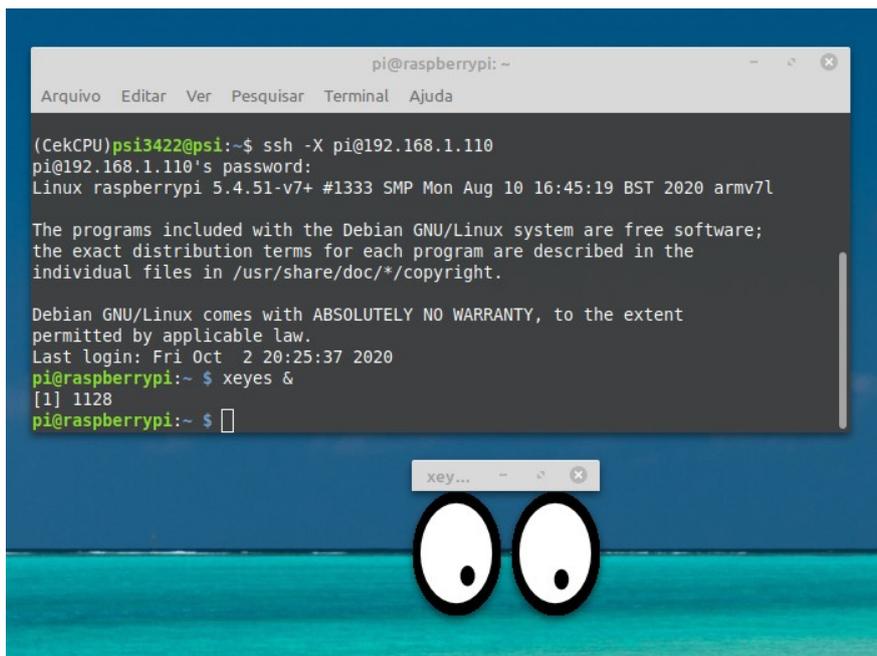


Figura 8: Janela de execução do programa em Raspberry pode ser mostrada no computador através de "ssh -X".

[Lição de casa #1 da aula 1] Mostre "ssh -X" funcionando. Pode ser uma tela como a figura 8.

*Exercício opcional:* Como é possível fazer funcionar ssh quando o computador e o Raspberry não estiverem conectados ao mesmo roteador? Dica: “NAT”.

**Tipo C)** Use sshfs (SSH filesystem) para montar a estrutura de diretórios do Raspberry como um subdiretório virtual do computador:

<https://www.raspberrypi.com/documentation/computers/remote-access.html>

Exemplo: Crie um diretório (por exemplo) /home/usuario/pi do computador (troque “usuario” pelo seu username).

```
$ cd ~
$ mkdir pi
```

Depois, dê o seguinte comando no computador:

```
$ sshfs pi@192.168.0.110: /home/usuario/pi
[troque 192.168.0.110 pelo número IP do seu Raspberry]
```

*Nota:* Se sshfs não estiver instalado no seu computador, instale-o

```
$ sudo apt install sshfs
```

Depois disso, se você acessar o diretório /home/usuario/pi do computador, estará acessando na verdade a estrutura de diretório do Raspberry.

Para desmontar o volume:

```
$ fusermount -u /home/usuario/pi
```

*Opcional:* Se você não quiser decorar e ficar digitando todos os comandos acima, pode criar “script bash”. Por exemplo, criando o arquivo “montar-pi” com o conteúdo:

```
sshfs pi@192.168.0.110: /home/usuario/pi
```

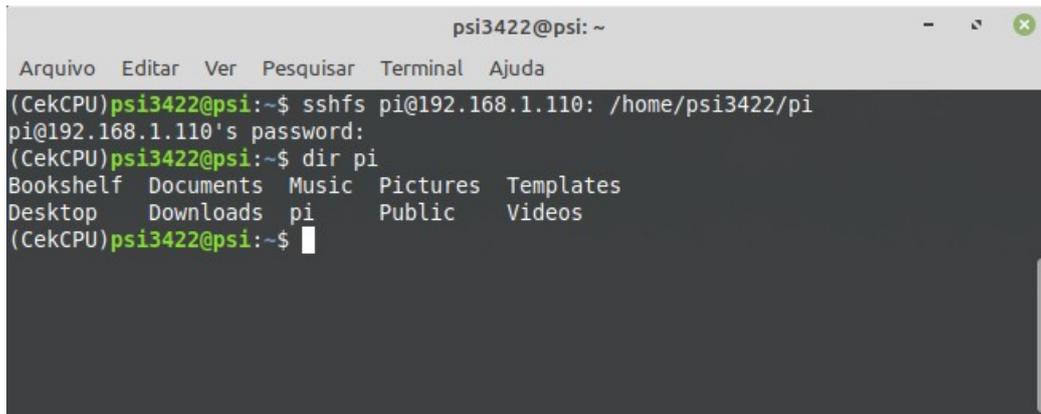
e tornando-o executável e deixando-o em algum diretório listado no PATH:

```
$ chmod +x montar-pi
```

you can type “montar-pi” instead of typing the whole command. You can do the same for “desmontar-pi”.

[Lição de casa #2 da aula 1] Mostre sshfs funcionando. Pode mostrar o acesso ao diretório do Raspberry via terminal e/ou via gerenciador de arquivos (como na figura 9).

*Exercício opcional:* Como é possível fazer funcionar sshfs quando o computador e o Raspberry não estiverem conectados ao mesmo roteador? Dica: “NAT”.



```
psi3422@psi: ~
Arquivo Editar Ver Pesquisar Terminal Ajuda
(CekCPU)psi3422@psi:~$ sshfs pi@192.168.1.110: /home/psi3422/pi
pi@192.168.1.110's password:
(CekCPU)psi3422@psi:~$ dir pi
Bookshelf Documents Music Pictures Templates
Desktop Downloads pi Public Videos
(CekCPU)psi3422@psi:~$
```

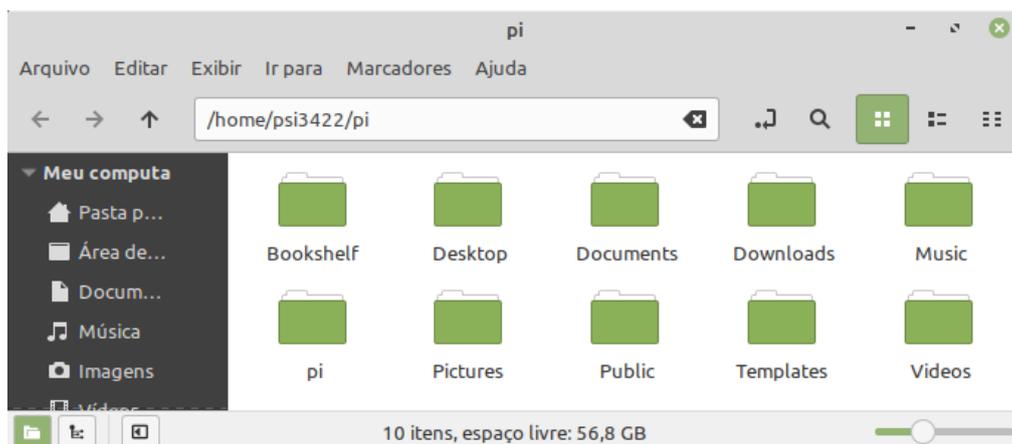


Figura 9: Sshfs permite acessar o sistema de arquivos do Raspberry de como um subdiretório do computador.

6) Para copiar arquivos entre computador e Raspberry, pode usar scp (secure copy protocol), rsync (remote sync), FreeFileSync, FileZilla ou simplesmente sshfs junto com cp.

<https://www.raspberrypi.org/documentation/remote-access/ssh/scp.md>

<https://www.raspberrypi.org/documentation/remote-access/ssh/rsync.md>

<http://www.freefilesync.org/>

7) Para sincronizar as pastas do projeto do computador e do Raspberry:

Primeiro monte a estrutura de diretórios de Raspberry como ~/pi:

```
>sshfs pi@192.168.0.110: /home/usuario/pi
```

Gosto de usar FreeFileSync para manter os dois diretórios sincronizados, pois é bem amigável. No *Linux Mint*, esse aplicativo pode ser instalado através do “gerenciador de aplicativos”. Infelizmente, não há uma versão pré-compilada deste programa para Raspberry. Porém, sempre é possível fazer sincronização a partir do computador.

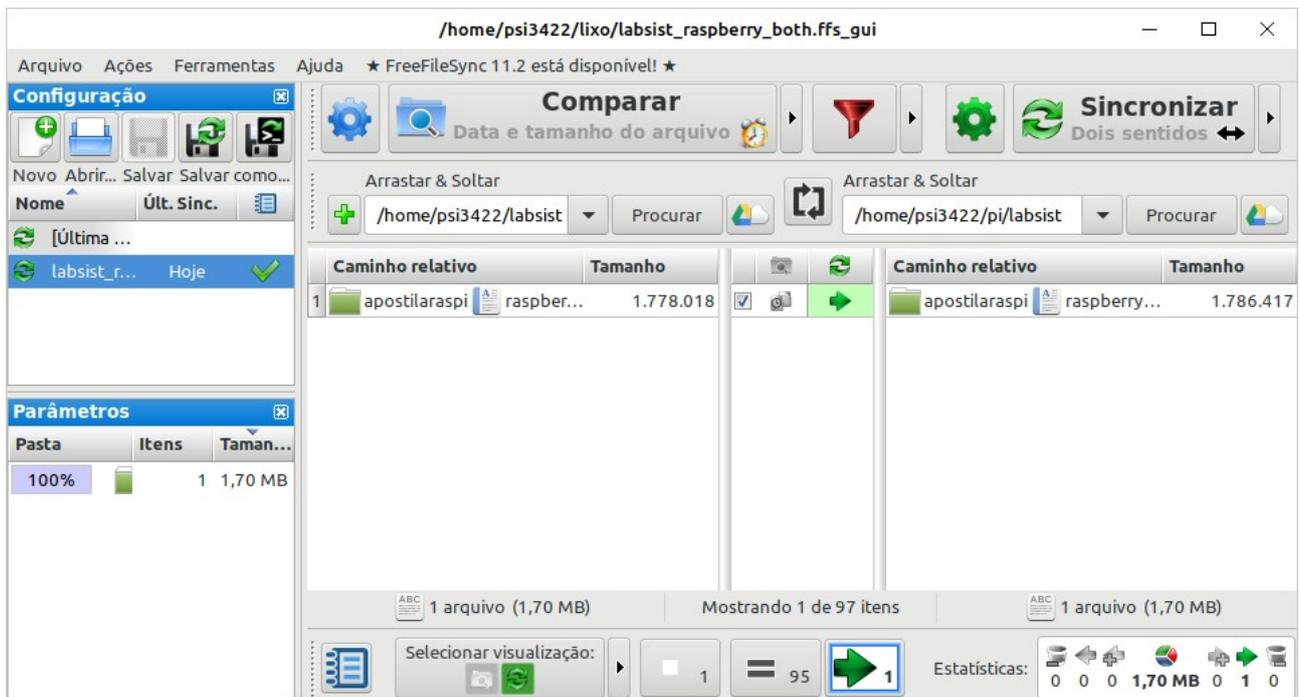


Figura 10: Janela do FreeFileSync.

Também é possível sincronizar os diretórios /home/psi3422/labsist (do Linux) e /home/psi3422/pi/labsist (do Raspberry) executando rsync nas duas direções:

```
rsync -rtuv /home/psi3422/labsist/ /home/psi3422/pi/labsist
rsync -rtuv /home/psi3422/pi/labsist/ /home/psi3422/labsist
```

A barra / após o diretório de origem é realmente necessária. Para mais detalhes, veja:

- <https://www.digitalocean.com/community/tutorials/how-to-use-rsync-to-sync-local-and-remote-directories-on-a-vps>
- <https://stackoverflow.com/questions/1602324/rsync-how-do-i-synchronize-in-both-directions>
- <https://linux.die.net/man/1/rsync>

8) Iremos desenvolver um sistema servidor-cliente (para ser executado no Raspberry e no computador). É recomendável que você crie um ambiente onde se pode trabalhar com os dois dispositivos ao mesmo tempo. Caso contrário, provavelmente irá perder muito tempo chaveando entre um dispositivo e outro. Figura 11 mostra um ambiente de desenvolvimento, onde na coluna esquerda há o editor de texto e terminal do computador e na coluna direita há o editor de texto e terminal do Raspberry.

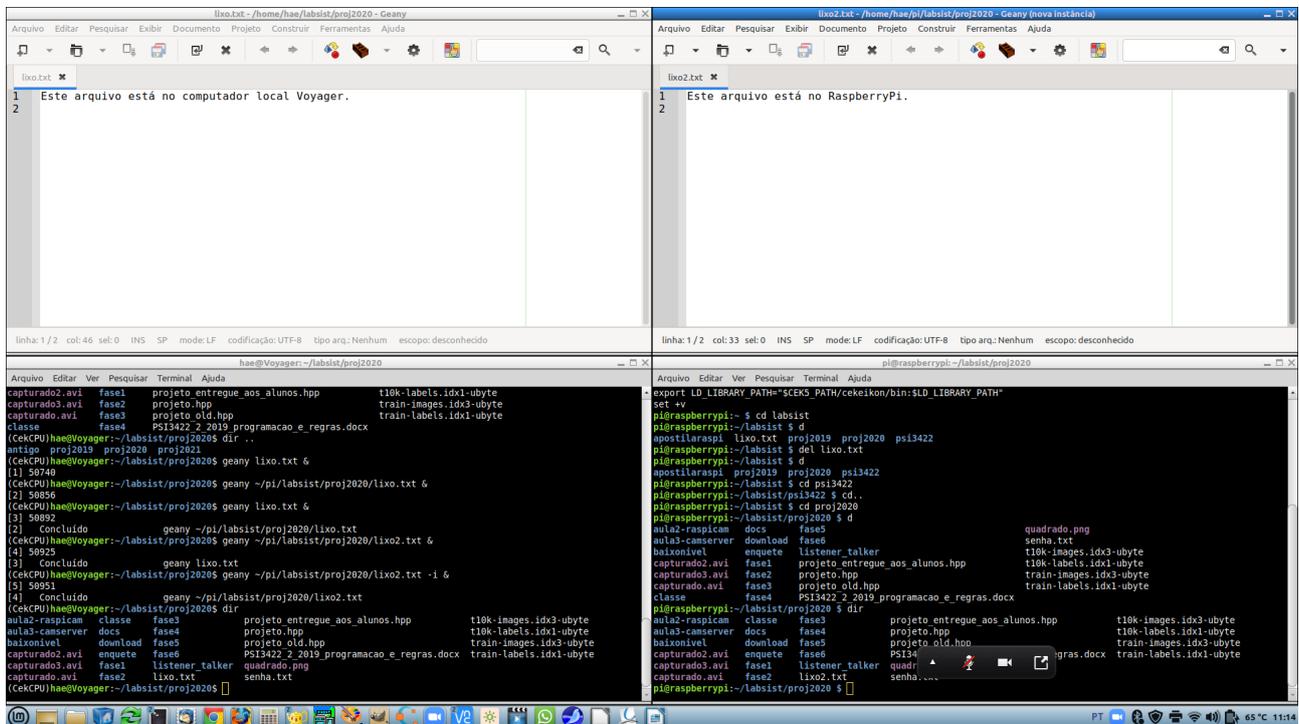


Figura 11: Um ambiente para trabalhar ao mesmo tempo em computador (coluna da esquerda) e Raspberry (coluna da direita).

Existem duas formas de criar ambiente como o da figura 11:

1) Executando remotamente o editor (no Raspberry) e visualizando a janela do editor no computador:

```
ssh -X pi@192.168.0.110
raspberrypi$ geany programa.cpp &
```

2) Executando o editor localmente no computador e acessando o arquivo de Raspberry remotamente via sshfs:

```
cd ~ && mkdir pi [Se diretório ~/pi ainda não existe]
sshfs pi@192.168.0.110: /home/usuario/pi
computador$ geany ~/pi/diretorio/programa.cpp &
```

As duas opções funcionam. Porém, o editor se torna mais responsivo na opção (2), pois o editor é executado localmente no computador. Nesta opção, o editor somente vai acessar o disco de Raspberry para carregar ou salvar arquivo remotamente no Raspberry. Usando opção (1), o editor fica lento se a conexão raspberry-computador for ruim, pois toda atualização da janela deve ser transmitida do Raspberry para computador.

*Exercício:* Mostre um ambiente como o da figura 11 funcionando.

[Aula 1 parte 2. Fim.]