

# Raspberry e Raspbian

## 1 Apresentação

Trechos extraídos de wikipedia.

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 que se esperaria de um computador desktop, como navegar na internet, reproduzir vídeo, fazer planilhas, processamento de texto, e jogar jogos. Além do mais, o Raspberry Pi tem a capacidade de interagir com o mundo exterior através da conexão GPIO (General Purpose Input/Output), e tem sido usado numa ampla gama de projetos digitais como um 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, a sua velocidade é 1,2 GHz e possui memória RAM de 1 GBytes. A fundação fornece Raspbian, uma distribuição de Linux baseado em Debian para download e vários outros sistemas operacionais de terceiros.



Figura 1: Raspberry Pi 3 Model B (do Wikipedia).

Arquitetura ARM (primeiramente Acorn RISC Machine, posteriormente Advanced RISC Machine) é uma arquitetura de processador usada principalmente em sistemas embarcados, smartphones, e tablets. Os processadores ARM são conhecidos pela sua versatilidade, pois possuem poucas instruções para programação. O padrão RISC do processador permite que estes processadores tenham menos transistores que processadores CISC (como x86). Essa abordagem reduz custos, liberação de calor e consumo de energia. Essas são características desejáveis para dispositivos portáteis. A empresa britânica ARM Holdings desenvolve a arquitetura e a licencia para outras empresas, que projetam seus próprios produtos que implementam uma dessas arquiteturas - incluindo sistemas em chips (SoC) que incorporam memória, interfaces, etc.

Nos meus testes "grosseiros", Intel i7 foi umas 6 vezes mais rápido Raspberry Pi 3 em programas sequenciais (sem usar paralelismo e sem usar instruções vetoriais).

No mercado, há outras placas semelhantes à Raspberry Pi, entre elas: Banana Pi, Odroid e Orange Pi, e alguns destes modelos possuem desempenho consideravelmente superior a Raspberry Pi.

## 2 Instalação do sistema Raspbian

Raspbian é um sistema operacional livre baseado em Debian e otimizado para Raspberry Pi. A documentação da Raspberry/Raspbian é bem didática e está em:

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

Não irei repetir aqui o que está escrito no site oficial. Leia-o.

### 2.1 Instalação

O pesquisador Maurício Pérez deixou uma boa lista de instruções para instalar Raspbian:

[http://www.lps.usp.br/hae/apostilaraspi/Instalacao\\_raspbian.pdf](http://www.lps.usp.br/hae/apostilaraspi/Instalacao_raspbian.pdf)

No site do Raspberry também tem boas informações:

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

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

Se você já conhece bem o Linux, pode pular o resto desta apostila.

### 2.2 Principais comandos

Pratique várias características do Raspian para se familiarizar. Para aprender os principais comandos de Linux, leia:

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

Procure aprender e praticar pelo menos:

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

### 2.3 Atualização

Para atualizar o sistema Raspbian:

```
sudo apt-get update
```

```
sudo apt-get dist-upgrade
```

Nota: O comando "sudo apt-get upgrade" pode não fazer a atualização completa.

Para atualizar firmware:

```
sudo rpi-update
```

### 2.4 Instalação de um programa

Um aplicativo pode ser instalado usando o programa "início → preferences → add/remove software" ou pela linha de comando.

Instale um aplicativo usando "sudo apt install programa" (apt e apt-get são praticamente iguais), por exemplo:

```
sudo apt install firefox-esr (Browser de internet.)
```

```
sudo apt install gimp (Manipulador de imagens)
```

```
sudo apt install geany (Editor de textos e programas GUI)
```

```
sudo apt install smplayer (Visualizador de vídeo)
```

Remova um aplicativo usando "sudo apt remove programa". Se quiser apagar também os arquivos de configuração, use "sudo apt purge programa".

## 2.5 Editor de texto

Existem vários editores de texto "GUI" no Raspbian. Você pode usar o que gostar mais para editar o seu programa:

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

No Raspberry, já vem leafpad e scite. Eu gostei do geany. 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.

## 2.6 Variáveis de ambiente

Uma variável de ambiente é uma variável de um sistema operacional que contém informações sobre a maneira como um processo se comporta. Tanto Linux como Windows utilizam-nas.

Por exemplo, HOME (Linux) ou HOMEPATH (Windows) contém o diretório raiz do usuário corrente.

PATH contém a lista de diretórios onde pode estar um arquivo executável (tanto Linux como Windows). No Linux, os diretórios são separados por dois pontos ":". No Windows, os diretórios são separados por ponto-e-vírgula ";".

Em Linux, LD\_LIBRARY\_PATH indica a lista de diretórios onde pode estar biblioteca compartilhada (\*.so). Em Windows, os diretórios das bibliotecas compartilhadas (\*.dll) são indicadas pela variável PATH.

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>).

No Linux, você pode listar todas as variáveis com:

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

No Windows, use:

```
>set
```

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

```
echo $NOMEVAR (linux)  
echo %NOMEVAR% (windows)
```

No diretório HOME, fica o arquivo `.bashrc` que é executado toda vez que `bash` (terminal de comando) é inicializado. Você pode mudar algumas configurações nele para deixar Raspbian 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"
```

não haverá a necessidade de escrever `./programa` para executar programa que se encontra no diretório default. Bastará escrever `programa`.

## 2.7 Execução automática

1) Se você quiser rodar um programa não gráfico ao inicializar Raspberry, edite `/etc/rc.local`, conforme:

```
https://www.raspberrypi.org/documentation/linux/usage/rc-local.md
```

2) Se você quiser rodar um programa GUI no início da sessão LXDE (Lightweight X11 Desktop Environment) edite o arquivo `/home/pi/.config/lxsession/LXDE-pi/autostart` Conforme:

```
www.raspberrypi.com/pi/pi-operating-systems/raspbian/auto-running-programs-gui
```

3) Se você quiser rodar um programa quando inicia o terminal de comando, edite `.bashrc` que fica no diretório `/home/pi`. Esse arquivo pode não aparecer na lista de arquivos, pois arquivos que começam com ponto são considerados arquivos escondidos. Você deve escrever `ls -l` ou `ls -a` para visualizá-los.

## 2.8 Checar disco

Se você desconfiar que Raspberry está com cartão 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
```

# 3 Controlar Raspberry remotamente

Raspberry pode ser controlado remotamente por um computador com Linux, Windows ou em "Bash on Ubuntu on Windows 10" (abreviaremos aqui como BashWin). Ele pode funcionar sem monitor, teclado e mouse próprios (headless). Siga os passos de:

```
https://www.raspberrypi.org/documentation/remote-access/
```

Para ver o número IP do seu computador ou Raspberry escreva:

```
>hostname -I
```

1) [Pode pular este passo se constatar que o seu roteador não muda número IP toda vez que desliga/liga.] 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. A forma de configurar depende do modelo do roteador. Estou usando 192.168.0.110 para Raspberry e 192.168.0.109 para computador. Procure aprender o que são número MAC e endereço IP. No roteador D-link (figura 2), pode-se fixar os números IP na opção "network settings" → "DHCP reservations list".

Enable	Host Name	MAC Address	IP Address		
<input checked="" type="checkbox"/>	raspberrypi wifi	B8:27:EB:9A:89:64	192.168.0.110		
<input checked="" type="checkbox"/>	odroid ethernet	[REDACTED]	192.168.0.111		
<input checked="" type="checkbox"/>	Royale wifi	D8:0F:99:C0:10:1D	192.168.0.109		
<input checked="" type="checkbox"/>	Royale ethernet	f4:8e:38:f0:57:0b	192.168.0.100		
<input checked="" type="checkbox"/>	raspberrypi ethernet	b8:27:eb:cf:dc:31	192.168.0.101		
<input checked="" type="checkbox"/>	Air-de-Hae	[REDACTED]	192.168.0.104		
<input checked="" type="checkbox"/>	Galaxy-J7-Prime	[REDACTED]	192.168.0.102		

Figura 2: Menu de reserva de número IP do D-link (network setting → DHCP reservation list).

- 2) Podemos classificar o acesso remoto a Raspberry em três tipos principais:
- Acessar somente os arquivos de Raspberry. Neste tipo de acesso, não é possível executar remotamente os programas no Raspberry. Ex: sshfs, scp, rsync, FreeFileSync, etc.
  - Acessar o terminal do Raspberry. Neste acesso, há um terminal (janela de comando ou bash) do Raspberry no computador e é possível executar os programas do Raspberry remotamente. Ex: ssh.
  - Acessar o sistema gráfico do Raspberry no computador. Ex: VNC.

3) VNC (Virtual Network Computing). É possível acessar remotamente o desktop do raspberry de computador. Leia a documentação para saber como fazer isto.

<https://www.raspberrypi.org/documentation/remote-access/vnc/README.md>

A figura 3 mostra VNC funcionando. A tela do Raspberry aparece como uma janela do computador.

Nota: Use RealVNC no computador para acessar Raspberry (link abaixo). Parece que TightVNC e outros programas não funcionam bem com Raspberry.

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

Nota: Se não há preocupação com segurança, pode desligar criptografia de VNC. Isto torna a comunicação um pouco mais rápida.

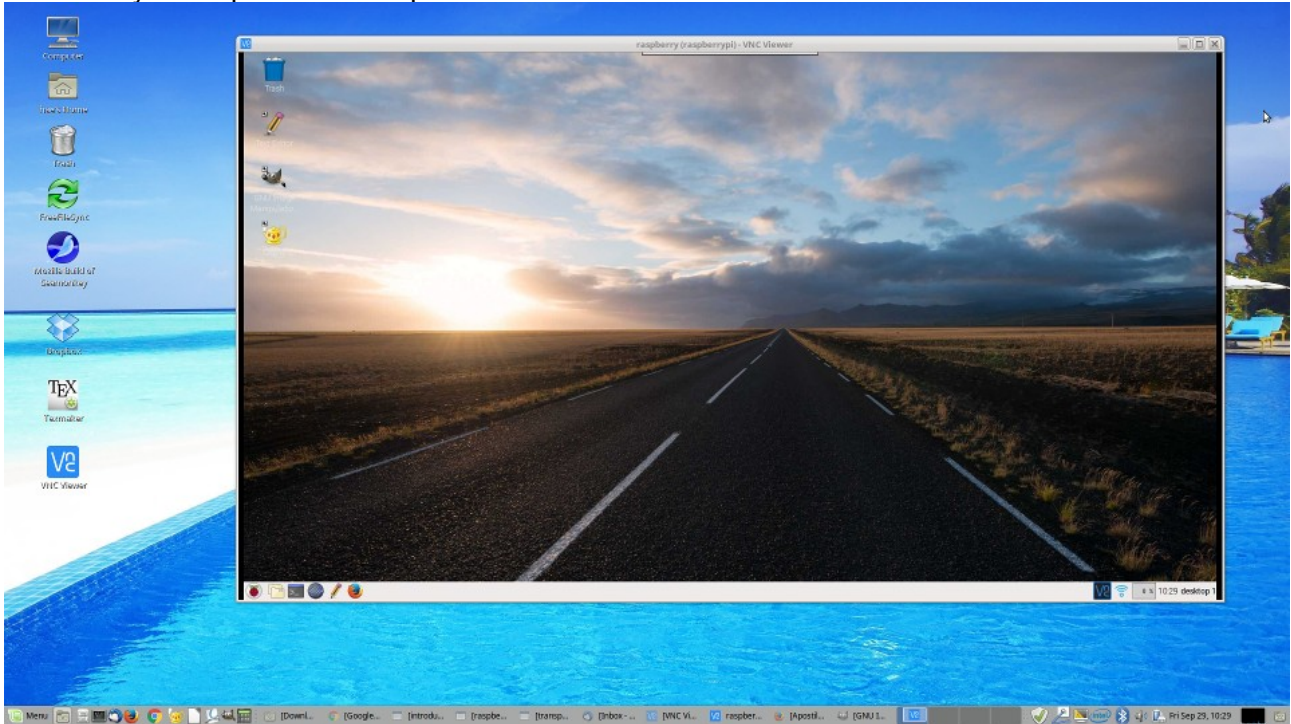


Figura 3: Controlar Raspberry remotamente através de VNC. A tela do Raspberry aparece como uma janela no computador remoto.

**Exercício 1 valendo 5 pontos) Mostre ao professor VNC funcionando. Se não conseguir fazer este item durante a aula, deve trabalhar fora da aula e mostrar este item funcionando na aula seguinte.**

4) Também é possível acessar a linha de comandos de Raspberry através de SSH:

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

tanto de Linux, Windows ou BashWin. O comando seguinte permite acessar remotamente o terminal de comandos do Raspberry:

```
ssh pi@192.168.0.110
```

É possível rodar remotamente os programas GUI do Raspberry através da opção:

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

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

No Linux, isto deve funcionar sem fazer nada extra. Depois de estabelecida a conexão "SSH -X", é possível abrir programas gráficos do Raspberry chamando-os da linha de comando. Por exemplo:

```
pi@raspberrypi$ geany &
```

```
pi@raspberrypi$ xeyes &
```

[Itens em amarelo indicam particularidades do Windows ou do BashWin] No Windows e BashWin, é necessário primeiro instalar e executar o programa Xming (servidor de X11 para Windows). Além disso, deve setar localmente a variável ambiente:

```
export DISPLAY=localhost:0 (BashWin)
set DISPLAY=localhost:0 (Windows)
```

antes de chamar "ssh -X 192.168.0.110". Você pode tornar esses comandos permanentes no Linux (editando .bashrc) e no Windows ("windows explorer > este computador > propriedades > configurações avançadas do sistema" ou usando o programa "edvar" do Cekeikon5). Depois, pode acessar raspberry no Windows por PuTTY ou chamando "ssh -X 192.168.0.110" no prompt de comando (ssh do msys vem junto com Cekeikon5 para Windows). No BashWin, chame diretamente "ssh -X 192.168.0.110".

5) Use sshfs para montar um diretório remoto.

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

Nota: Isto só pode ser feito em Linux (não pode ser feito em BashWin ou em Windows).

Exemplo: Crie um diretório pi no /home/usuario do computador remoto.

```
>cd ~
>mkdir pi
```

Depois, dê o seguinte comando no computador:

```
>sshfs pi@192.168.0.110: /home/usuario/pi [troque usuario pelo username do seu computador]
```

Depois disso, se você acessar o diretório /home/usuario/pi do computador, estará acessando a estrutura de diretório do Raspberry. Para desmontar o volume:

```
>fusermount -u /home/usuario/pi [troque usuario pelo username do seu computador]
```

[Para 2019: Tornar sshfs item obrigatório.]

6) Copie arquivos entre computador e Raspberry usando scp, rsync ou FreeFileSync:

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

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

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

Nota: FreeFileSync é muito bom para manter os diretórios de Raspberry e computador sincronizados. Há FreeFileSync pré-compilado para Linux x86 e Windows. No Raspberry (ARM), infelizmente precisa compilar a partir da fonte. Porém, você pode sempre fazer sincronismo chamando FreeFileSync do computador. Dá para instalar FreeFileSync no Linux Mint através do "gerenciador de aplicativos".

Iremos desenvolver um sistema servidor-cliente (para ser executado no Raspberry e no computador). É importante que você crie um ambiente em que se pode trabalhar com dois computadores ao mesmo tempo. Caso contrário, provavelmente irá perder muito tempo chaveando entre um computador e outro. Figura 4 mostra um ambiente de desenvolvimento, onde na coluna esquerda há o editor de texto e janela de comandos do computador (BashWin) e na coluna direita há o editor de texto e janela de comandos do Raspberry.



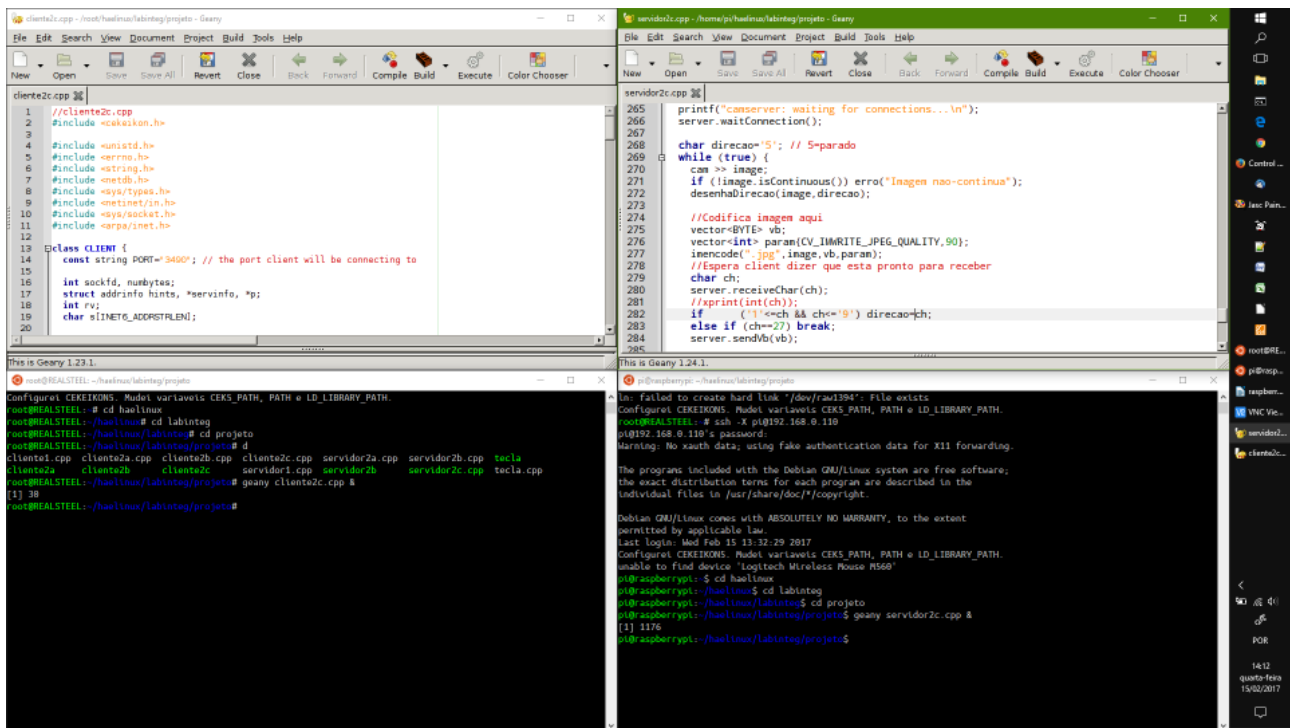


Figura 4: Um ambiente para trabalhar ao mesmo tempo em computador (coluna da esquerda) e Raspberry (coluna da direita).

**Exercício 2 valendo 5 pontos) Mostre ao professor um ambiente como o da figura 4 funcionando. Se não conseguir fazer este item durante a aula, deve trabalhar fora da aula e mostrar este item funcionando na aula seguinte.**

Existem duas formas de criar ambiente como o da figura 4 (usando um editor GUI, digamos Geany):

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 no localmente no computador e acessando o arquivo de Raspberry remotamente:

```
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). Usando opção (1), o editor fica lento.

É útil criar um esquema para manter os diretórios de projeto de computador e Raspberry sincronizados. Uso FreeFileSync para manter os diretórios dos dois computadores atualizados. Veja figura 5.



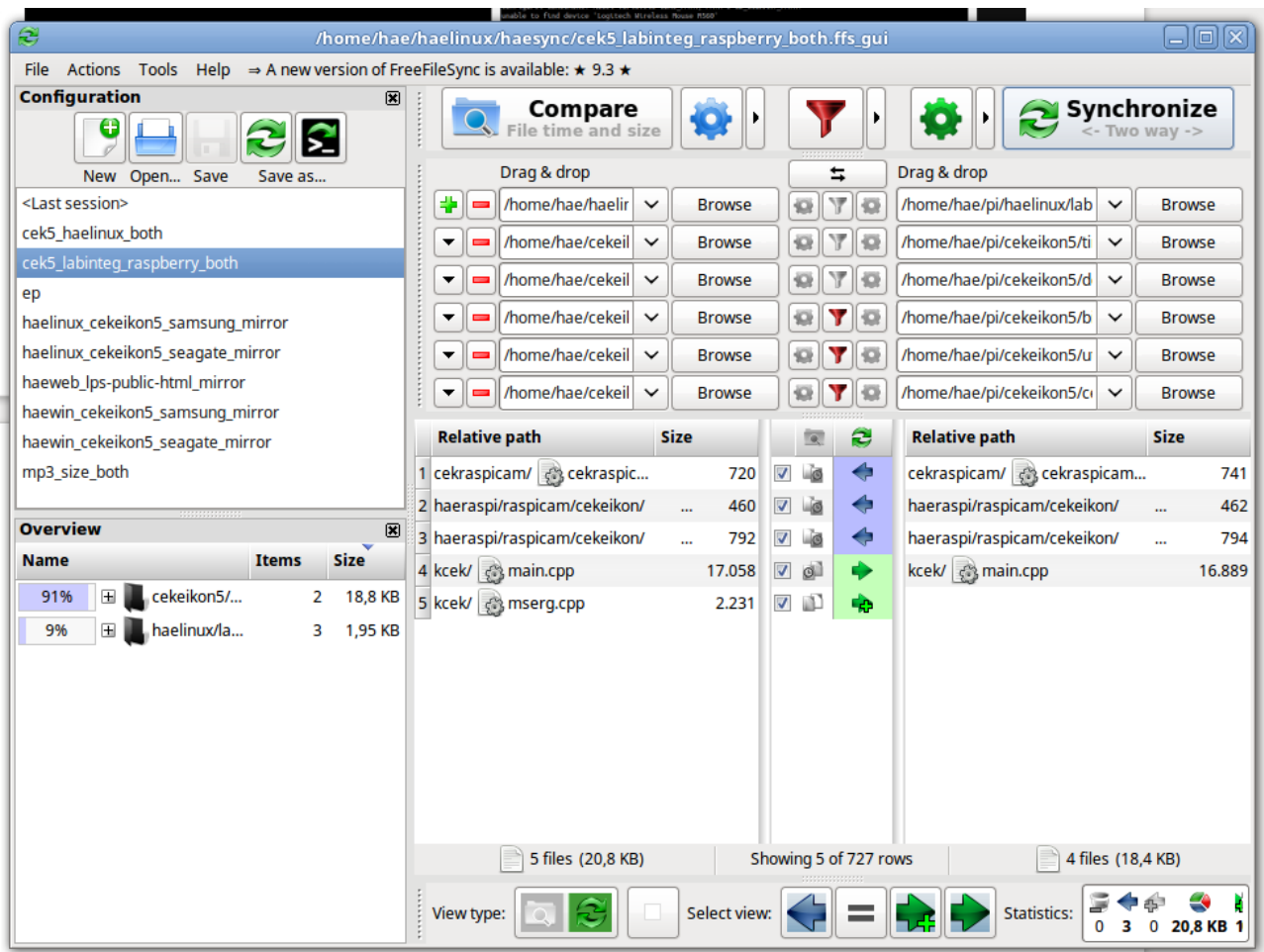


Figura 5: FreeFileSync detecta arquivos diferentes nos dois computadores e copia só o que for necessário para manter os diretórios sincronizados.

7) Raspberry não possui relógio permanente. Para acertar o relógio:

```
>sudo apt install ntpdate
```

Rodar quando iniciar:

```
>sudo ntpdate-debian 192.168.160.254
```

Ou

```
>sudo ntpdate br.pool.ntp.org
```