

Ambiente Windows

Cada Sistema Operacional possui suas características. Para desenvolvedores, é interessante (ou obrigatório) conhecer essas características e dominar o sistema (mesmo que superficialmente) para poder desenvolver algum software.

Nesta disciplina, optei por trazer o S.O. Windows, mas é recomendável estudar o Linux também, visto que em determinado momento poderá ser necessário implementar algo nele. Por minha experiência, digo que muitas coisas (instalar bibliotecas, software) são mais simples de serem feitas no Linux.

Organização dos Arquivos

Algo interessante, ou melhor, necessário de se saber sobre um SO é a estrutura de diretórios (pastas) padrão do sistema. Muitos conhecem as pastas “Downloads”, “Meus Documentos”, etc. Estes são diretórios criados para cada usuário do SO. Mas fora isso, o SO tem uma estrutura própria para armazenar arquivos importantes para seu funcionamento, instalação de software, etc.

O Windows identifica os discos com letras de acesso. Geralmente, o SO é instalado na partição **C:**. A partição **A:** era associada ao leitor de Disquetes (antigo, não?) e os demais discos ou leitores de CD/DVD/Bluray e drives externos (pendrive, HD externo) à letras aleatórias. Veja: [Por que "C" é a letra padrão para o disco rígido em tantos computadores.](#)

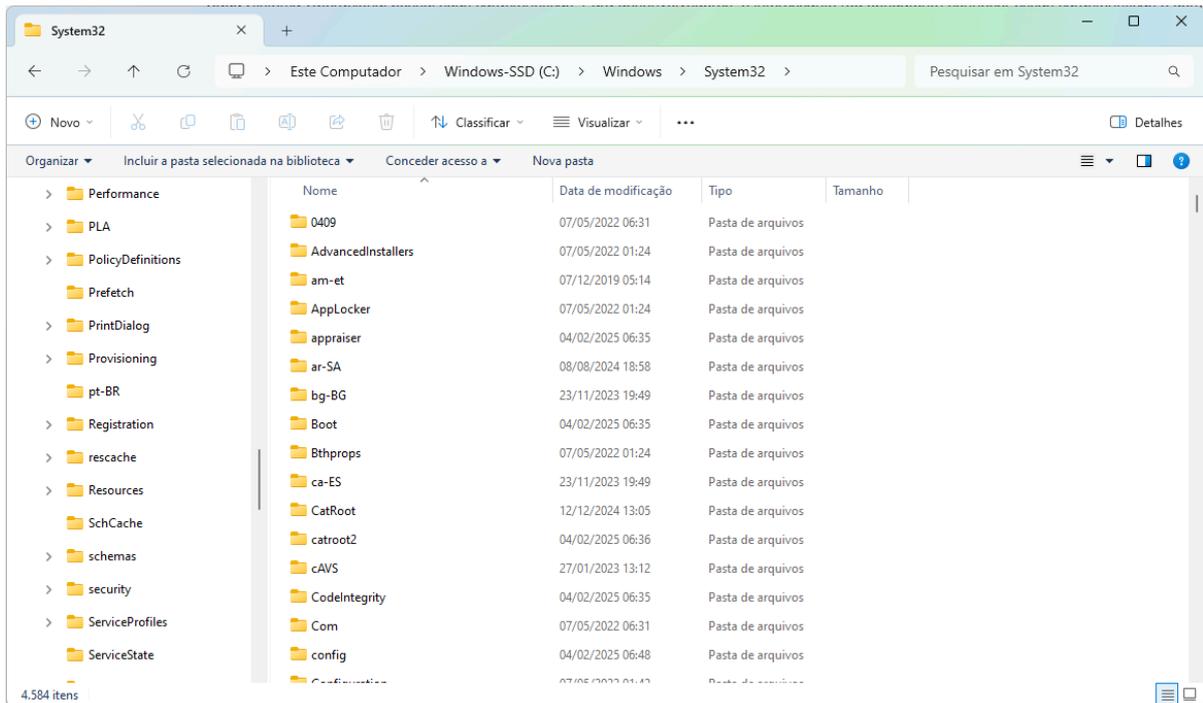
Os arquivos são armazenados com um nome e estão logicamente guardados em diretórios, ou pastas. veja: [Diretório \(computação\) – Wikipédia, a enciclopédia livre](#)

Uma pasta é um arquivo especial que armazena outros arquivos. Logicamente, pois, fisicamente (bits) os arquivos estão armazenados em setores do disco rígido ou mídia.

Todo arquivo em um SO possui um endereço para acessá-lo. Cada SO tem uma maneira de representar os endereços (ou caminhos). No Windows, o caminho começa com a letra da unidade e os diretórios são separados por barras (\). Os caminhos podem apontar para diretórios ou arquivos.

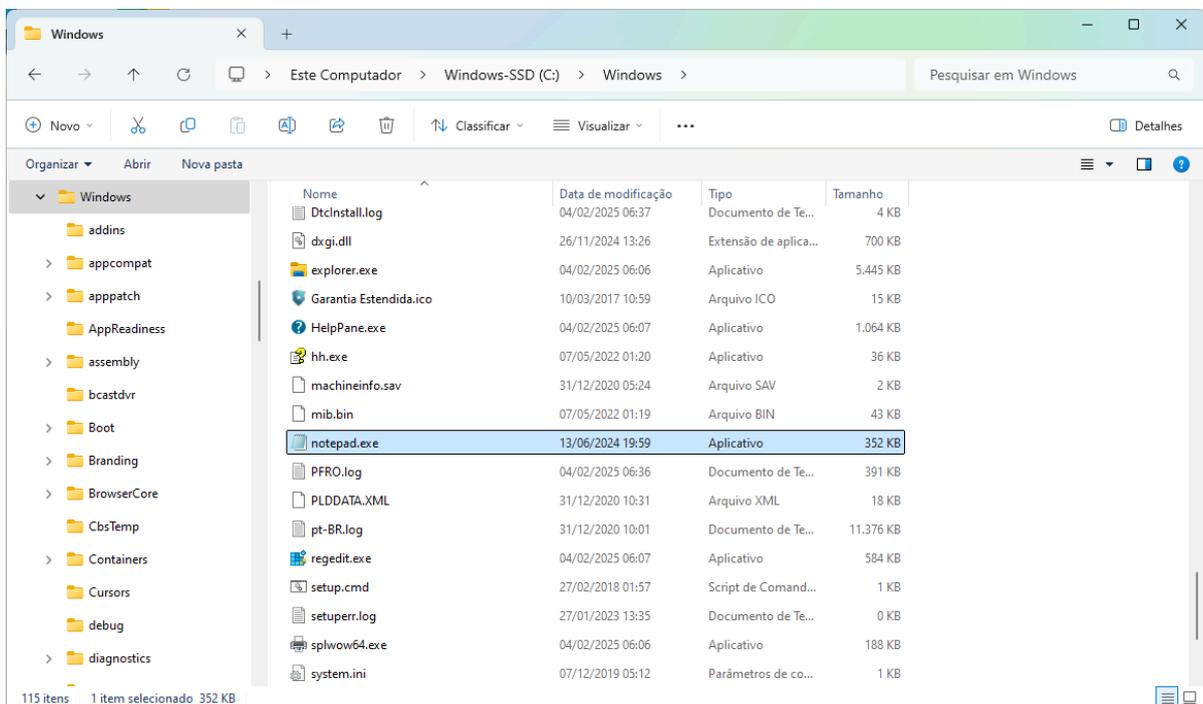
É muito importante compreender como funciona o endereçamento, pois, para compilar, acessar arquivos, incluir cabeçalhos, etc., será necessário apontar para caminhos ou arquivos específicos.

O endereço `C:\Windows\System32` corresponde à pasta `System32`, dentro da pasta `Windows`, que está na ‘raiz’ do Disco (partição) `C`. Podemos ler da esquerda para direita: Entre no disco `C`, depois na pasta `Windows`, depois na pasta `System32`.



Lembrando: este é o caminho que o SO Windows usa. Em sistemas Linux, por exemplo, os caminhos começam com uma barra apenas, e não pela letra do diretório. Por exemplo, para a pasta do usuário, um caminho seria /home/usuario.

Vejamos como exemplo o software Notepad (bloco de notas). O seu arquivo executável fica em C:\Windows



O caminho direto para o executável é C:\Windows\notepad.exe. Para testar, copie e cole em uma janela windows e o software vai abrir. Em alguns computadores, de acordo com a configuração, o '.exe' do arquivo não aparece no navegador de arquivos. Isso pois o

Windows esconde a extensão (após o ponto) dos arquivos que já possuem algum programa para abrir (conhecidos). Veja: [Extensão de nome de ficheiro – Wikipédia, a enciclopédia livre](#)

Um caminho que especifica desde a raiz até um diretório ou arquivo é chamado absoluto. Pense bem; caso um arquivo esteja no disco local C, o caminho absoluto para ele seria C:\.... Mas e se eu não souber se o usuário vai colocar o arquivo no disco C:, no disco D:, em um pendrive... Como posso ter um caminho para um arquivo?

Uma maneira é usar caminhos relativos. Estes indicam um caminho que não parte de uma raiz, mas sim, relativo à pasta atual. veja: [Caminho \(computação\) – Wikipédia, a enciclopédia livre](#)

Por exemplo, considere que eu estou na pasta C:. para acessar o arquivo do notepad, a partir da pasta atual (C:) o caminho relativo seria “Windows\notepad.exe”, ou seja, a partir daqui, entre na pasta Windows e depois abra o arquivo. Isto é útil pois, não importa onde a pasta Windows esteja (C: ou D:), eu consigo ter um caminho para o arquivo

Para descrever um caminho relativo a partir de um diretório, podemos usar o ‘.’ que significa “este diretório”. Por exemplo, se eu estiver na pasta “c:\Windows” e quiser informar o caminho para o “notepad.exe” posso inserir o nome diretamente ou “.notepad.exe”. Sistemas Linux exigem o uso do “./” enquanto no windows não é necessário [no linux a barra ‘/’ é invertida].

Outro operador é o dois pontos finais ‘..’, que significa “no diretório acima”, ou seja, o diretório que comporta o atual. Por exemplo, caso eu esteja na pasta “C:\Windows” o operador “..” é um caminho para o “C:” que é onde a pasta Windows está.

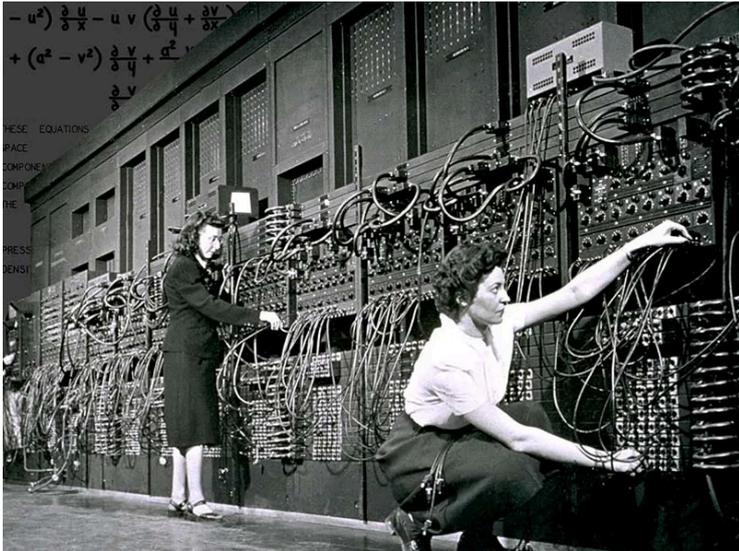
Usando a interface de arquivos (Windows Explorer) os caminhos acabam não sendo tão úteis. Mas ao configurar projetos de programação ou usar terminais, conhecer a estrutura de diretórios é essencial.

Isso ficará mais visível usando um terminal.

Usando o Terminal do Windows

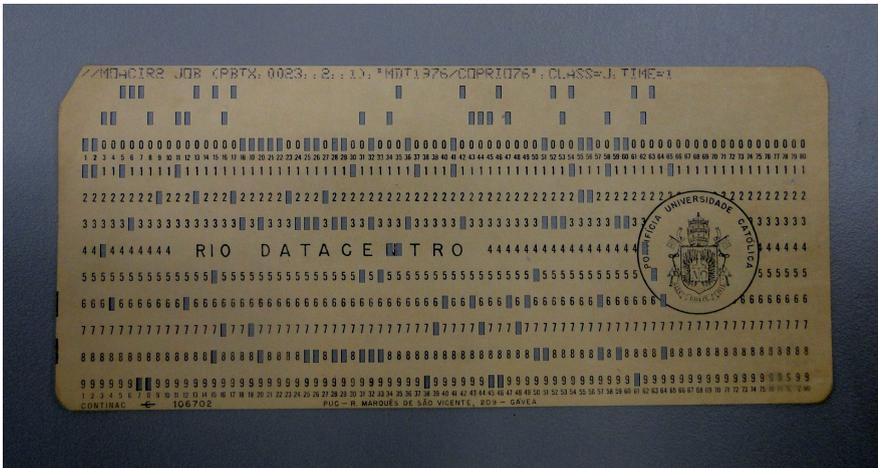
Nossa interação com computadores evoluiu bastante com o passar do tempo.

Válvulas



(ENIAC, CNN Brasil)

Cartões Perfurados



(Unoeste)



(Un. Lisboa)

Texto

```
C:\>chkdsk
Volume Serial Number is 3E76-4B58

2,146,467,840 bytes total disk space
 131,072 bytes in 2 hidden files
  32,768 bytes in 1 directories
 7,405,568 bytes in 124 user files
2,138,898,432 bytes available on disk

    32,768 bytes in each allocation unit
    65,505 total allocation units on disk
    65,274 available allocation units on disk

655,360 total bytes memory
602,704 bytes free

Instead of using CHKDSK, try using SCANDISK. SCANDISK can reliably detect
and fix a much wider range of disk problems. For more information,
type HELP SCANDISK from the command prompt.

C:\>_
```

(Higher

Intellect)

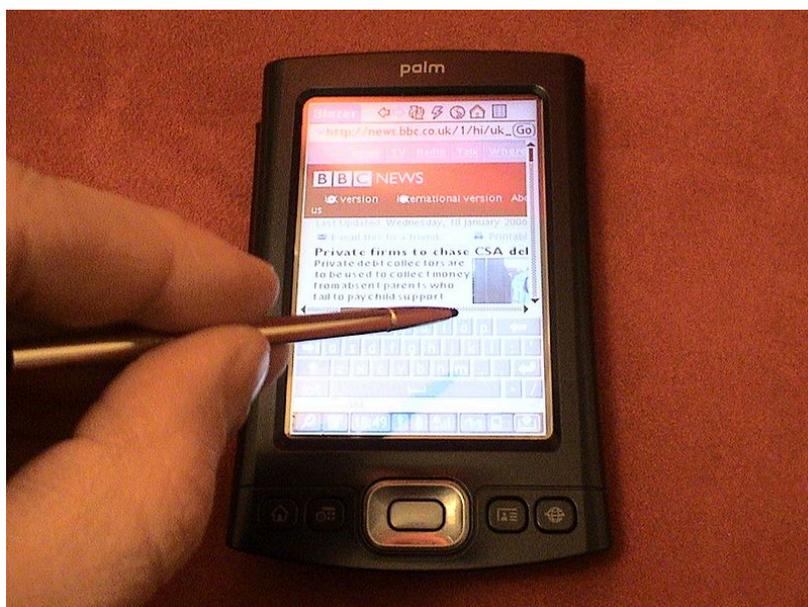


Xerox (Wikipedia)

Mouse,



(apple2history)

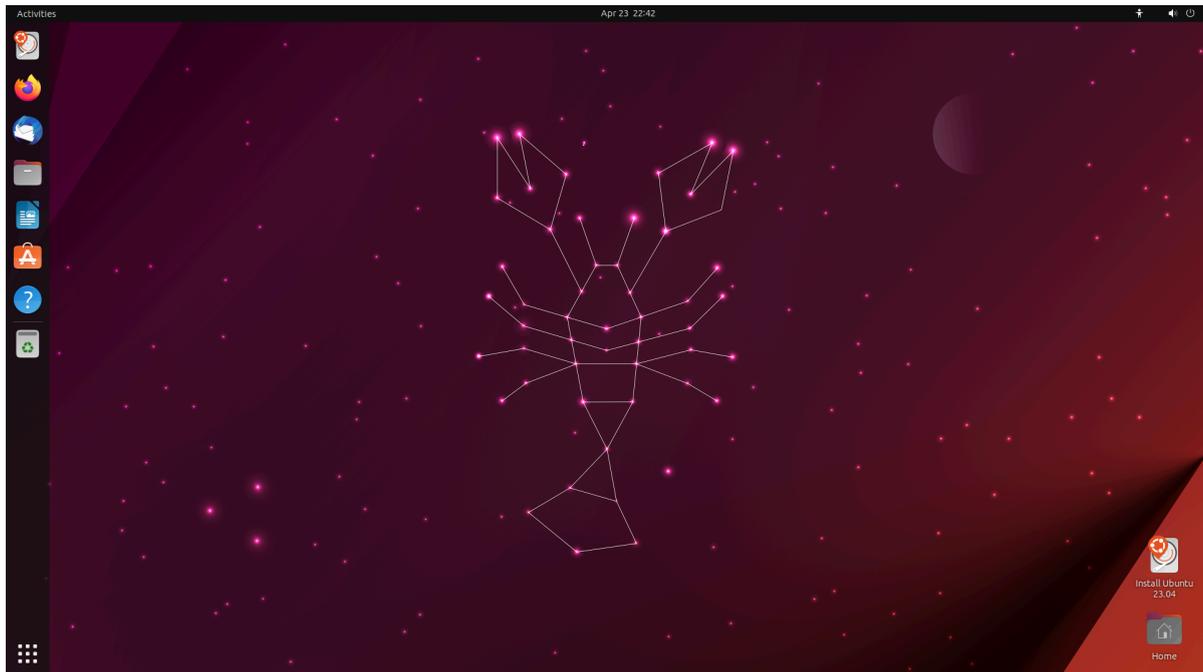


(touch) PALM

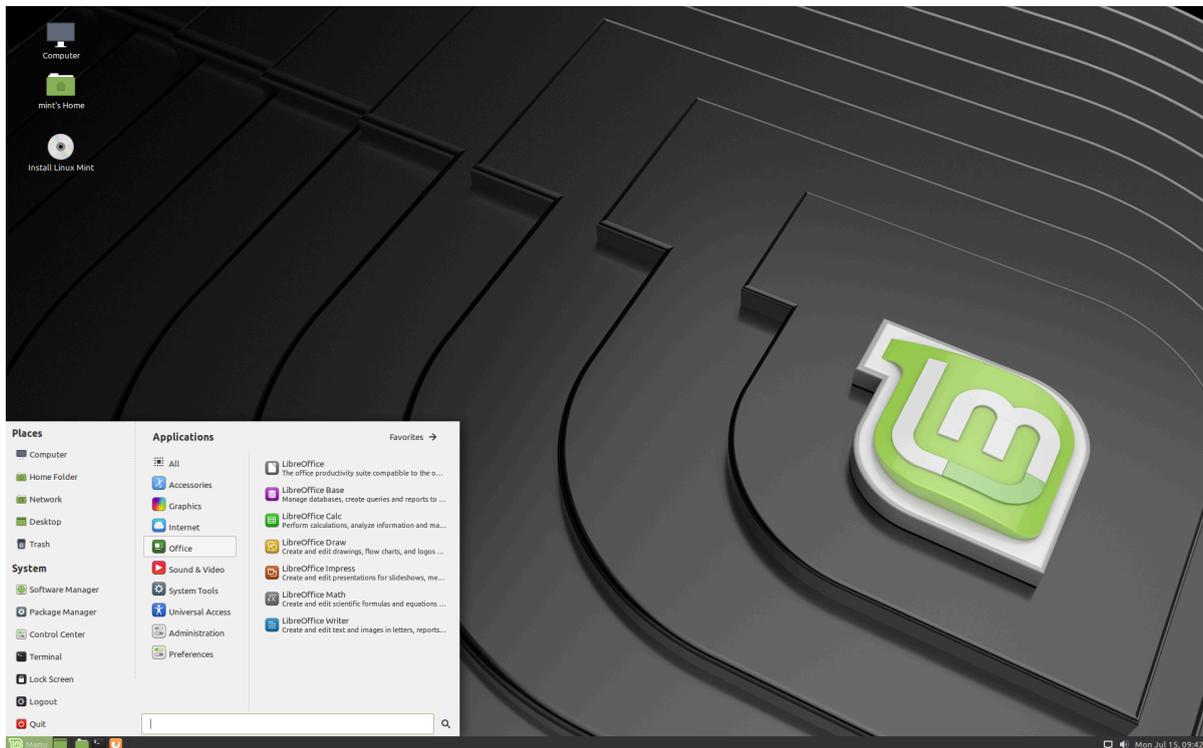
(Wikipedia)

Sugestão: Pirates of Silicon Valley / Piratas da Inforática (https://pt.wikipedia.org/wiki/Pirates_of_Silicon_Valley)

Linux é mais difícil pois só usa o terminal? Não, possui interface gráfica, assim como Windows.



Ubuntu (Wikipedia)

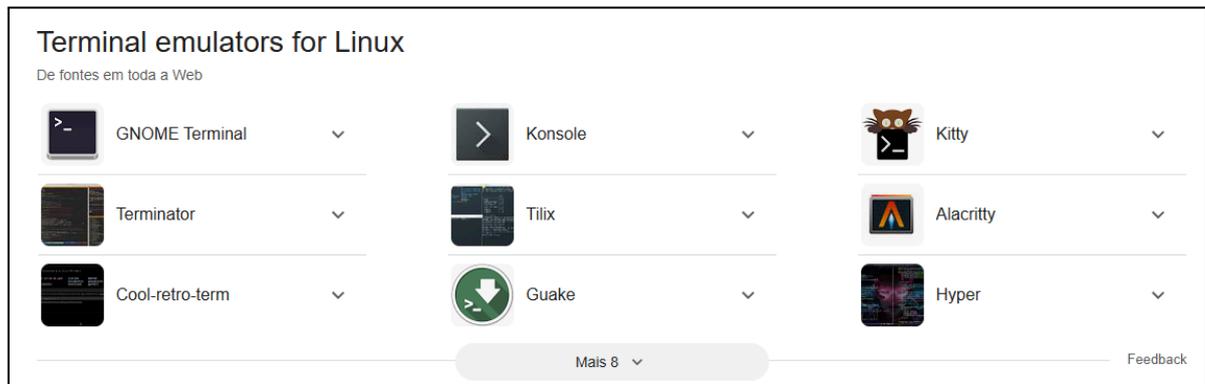


Linux Mint

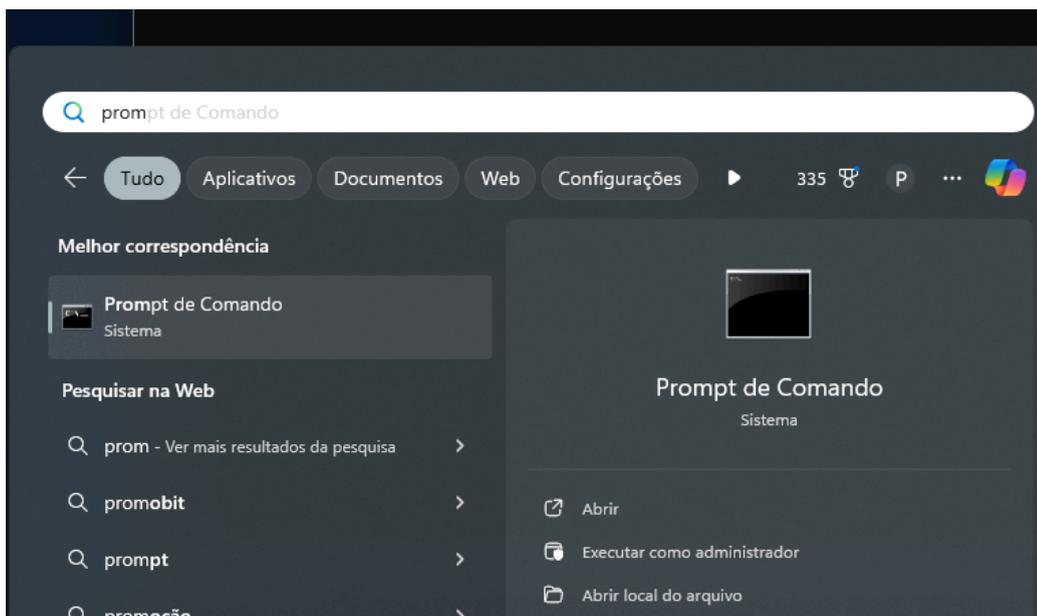
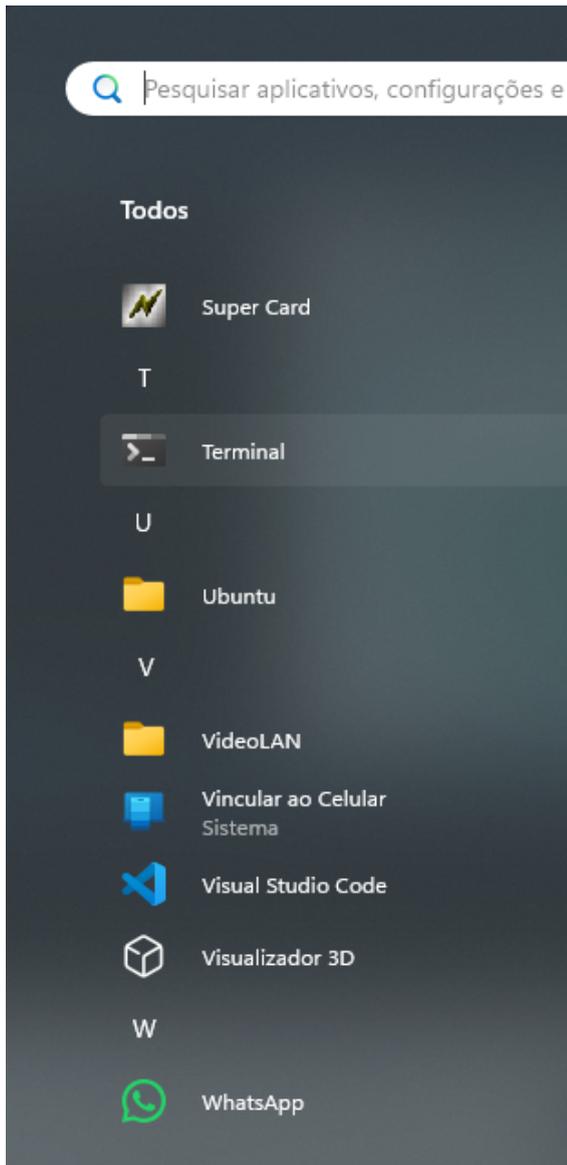
Por que usar o terminal?

Mais simples; alguns softwares não foram implementados com interface gráfica e quem precisa usá-los já está acostumado, por exemplo, o 'ping' no windows.

O terminal é um programa que acompanha o SO. O Windows tem o CMD e a partir do Windows 10 começou a ter outro, o PowerShell, um terminal um pouco mais elaborado. O Linux, dependendo da distribuição, possui um terminal.

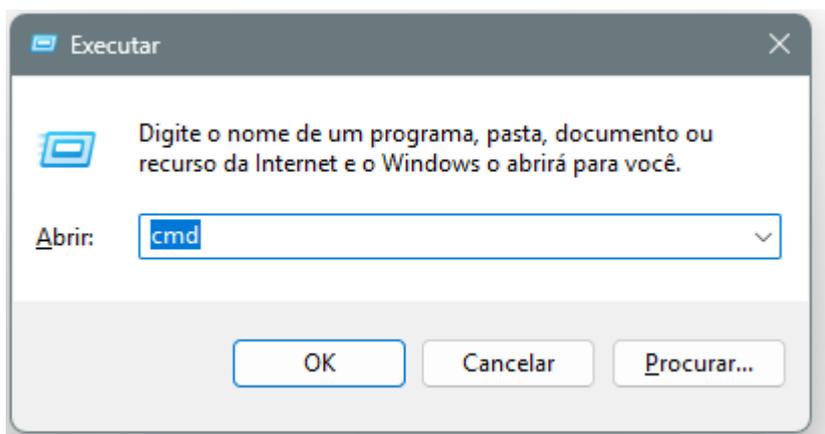


No windows, podemos encontrar no menu iniciar.



Varia em cada versão, mas basta “pesquisar” por “terminal” ou “prompt”.

Ainda, o windows permite executar processos apertando a tecla “Windows”+R.



Assim um dos terminais (CMD ou PowerShell) abre.

```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [versão 10.0.22631.4751]
(c) Microsoft Corporation. Todos os direitos reservados.
C:\Users\panca>
```

```
Windows PowerShell x + v
O Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Instale o PowerShell mais recente para obter novos recursos e aprimoramentos! https://aka.ms/PSWindows
PS C:\Users\panca>
```

“Navegando” usando o Terminal

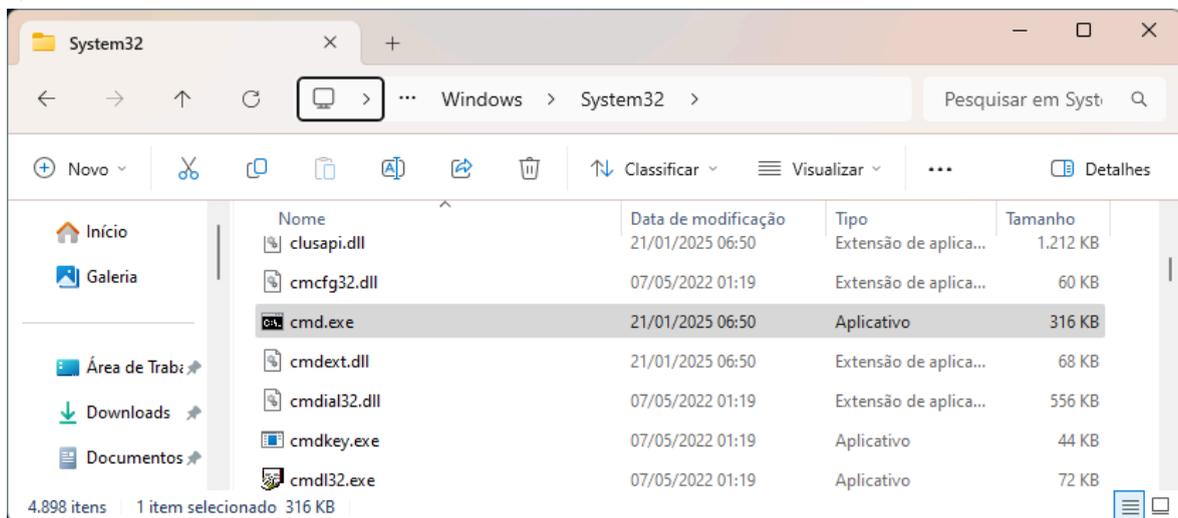
Ao abrir o terminal, ele estará em um diretório padrão, geralmente o diretório do usuário. Digitando “HELP” e apertando ENTER, aparecem os comandos disponíveis. Digitando “HELP” seguido de um comando específico, aparece a ajuda do comando. Por exemplo, “HELP CD”.

Comando	Efeito
CD [caminho absoluto relativo]	Muda o diretório atual
DIR [caminho]	Mostra os arquivos dentro do diretório atual ou de um passado por parâmetro
MKDIR [nome]	Cria um diretório

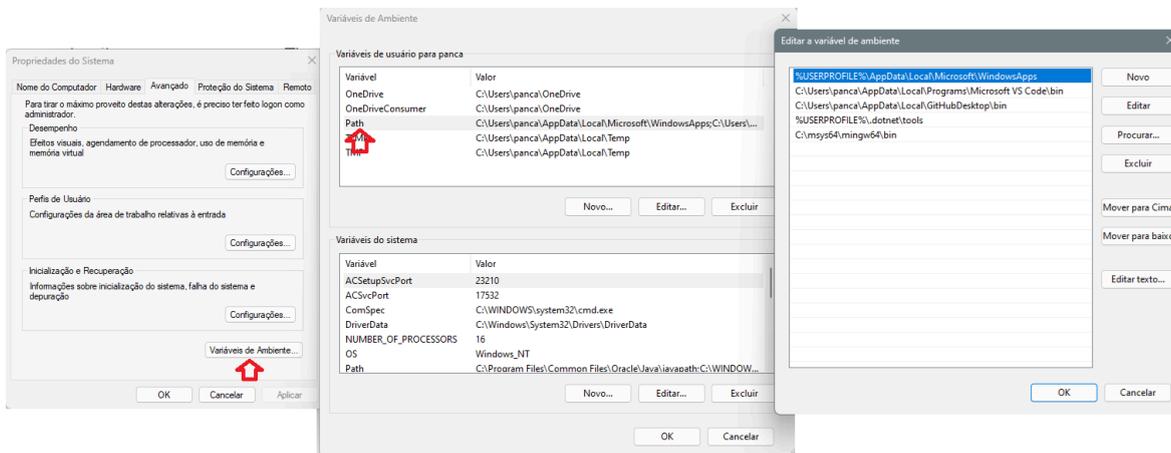
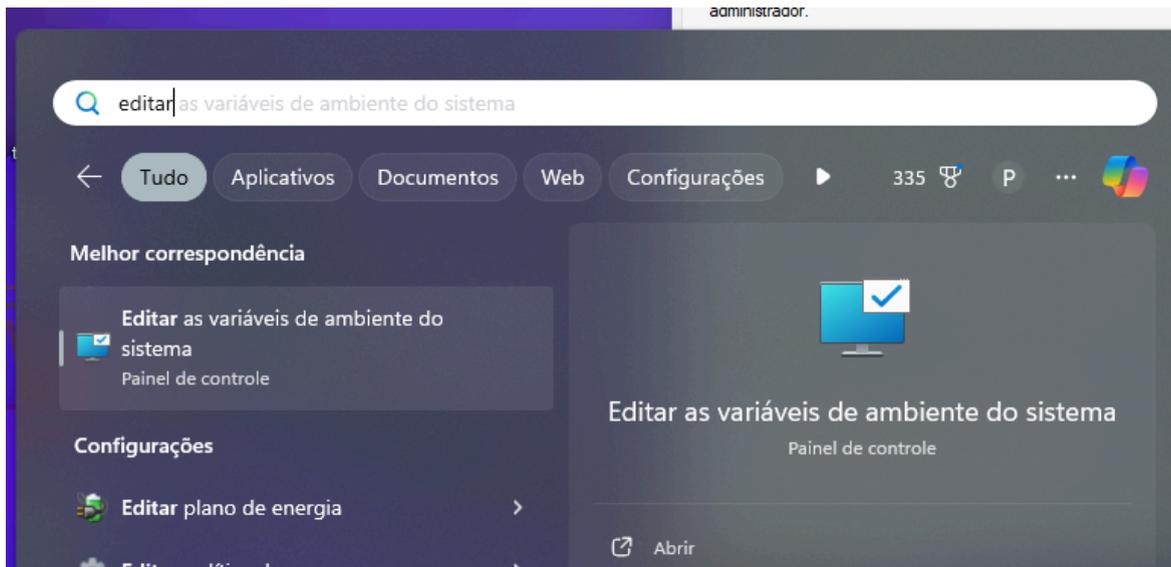
REN [nome] [novo_nome]	Renomeia arquivo ou diretório
MOVE [origem] [destino]	Movê um arquivo de um caminho (origem) para outro (destino)
ping >> arq.txt	o '>>' envia a saída do terminal para um arquivo. Apenas para teste
cls	Limpa a tela
color [fundo fonte]	Altera a cor
type [arquivo.txt]	Mostra o conteúdo de um arquivo texto
DEL [arquivo]	Remove permanentemente um arquivo. Cuidado!
rmdir [diretório]	Remove permanentemente um diretório. O diretório precisa estar vazio.

Como o sistema sabe como executar determinados comandos?

Alguns são implementados dentro do prompt de comando. Outros, são programas instalados em locais específicos, nos quais o SO procura. Por exemplo, o comando CMD (pelo executar ou chamado por um terminal) corresponde ao programa CMD.EXE, na pasta system32 do windows:



O sistema procura em diretórios específicos. É possível adicionar diretórios com programas para que o Windows procure, adicionado às variáveis do ambiente.



No Linux, geralmente, os instaladores de pacotes já adicionam os programas e componentes nos locais que o SO procura. Mas também é possível adicionar locais de busca.

Compilando um programa exemplo

Primeiramente, crie um arquivo "teste.c" e digite o seguinte conteúdo:

```
#include <stdlib.h>
#include <stdio.h>

int main() {
    printf("Hello, World!");
}
```

```
    exit(0);  
}
```

Abra um terminal, verifique se o comando “gcc” é reconhecido. Uma mensagem de erro deve aparecer:

```
C:\Users\panca\Documents\teste3>gcc  
gcc: fatal error: no input files  
compilation terminated.
```

Informando que faltaram arquivos de entrada. Caso o comando não seja reconhecido, outra mensagem irá aparecer:

```
C:\Users\panca\Documents\teste3>gcc  
gcc' não é reconhecido como um comando interno  
ou externo, um programa operável ou um arquivo em lotes.
```

Caso o comando não seja reconhecido, é necessário instalar o compilador e configurar as variáveis. Em seu computador, siga os passos de configuração de ambiente: [Criando Ambiente de Desenvolvimento](#)

Em computadores compartilhados (por exemplo os da universidade) pode haver restrições de acesso a usuários, ou seja, seu usuário não pode fazer alterações. Se não for possível inserir no PATH, usar o caminho completo ao gcc.exe, por exemplo :

```
C:\Users\panca\Documents\teste3>C:\msys64\mingw64\bin\gcc.exe  
gcc.exe: fatal error: no input files  
compilation terminated.
```

Agora, no prompt de comandos, “navegue” até o local onde o arquivo está salvo. Digite o comando para compilar:

```
gcc -o teste.exe teste.c
```

Se tudo estiver correto, nada aparecerá escrito. Caso o arquivo possua erros de digitação, podem aparecer erros:

```
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 int main(){ asdfaa
5     printf("Hello, World!");
6
7     exit(0);
8 }
```

```
C:\Users\panca\Desktop\sdl-test>gcc -o teste.exe teste.c
teste.c: In function 'main':
teste.c:4:13: error: unknown type name 'asdfaa'
   4 | int main(){ asdfaa
     |             ^~~~~~
teste.c:5:12: error: expected declaration specifiers or '...' before string constant
   5 |     printf("Hello, World!");
     |            ^~~~~~~~~~~~~~~~~
C:\Users\panca\Desktop\sdl-test>
```

Digitando DIR, um arquivo teste.exe deverá aparecer no diretório. Este é o programa já compilado!

```
Pasta de C:\Users\panca\Desktop\sdl-test

06/02/2025  10:59    <DIR>          .
06/02/2025  08:21    <DIR>          ..
04/02/2025  06:47    <DIR>          .vscode
04/02/2025  06:40                1.075 hello.c
06/02/2025  10:59                103 teste.c
06/02/2025  10:59          257.097 teste.exe
              3 arquivo(s)          258.275 bytes
              3 pasta(s) 66.227.970.048 bytes disponíveis

C:\Users\panca\Desktop\sdl-test>
```

Para executar o programa, basta digitar "teste.exe" ou ".\teste.exe"

```
C:\WINDOWS\system32\cmd. x + v
C:\Users\panca\Desktop\sdl-test>.\teste.exe
Hello, World!
C:\Users\panca\Desktop\sdl-test>
```

Exercícios

1. Leia o material acima e reproduza os exemplos. Chame o professor caso tenha dúvidas.
2. Em meu computador, eu possuo duas partições no disco: uma está associada à letra C: e outra à letra D:. Ao inserir um pendrive, qual letra será associada (pelo sistema) a ele? Explique os motivos.
3. Gere a árvore de diretórios de sua pasta de usuário.
4. Faça download do arquivo e compile os códigos-fonte. Use apenas o prompt de comando. Mostre a sequência de comandos utilizada. Não mova arquivos dos diretórios. Execute os arquivos finais (.exe). [Link](#)
5. Crie uma pasta com arquivos do tipo imagem, texto (.txt) e arquivos pdf - procure na internet exemplos. Pelo menos 30 arquivos. Coloque todos dentro de pastas específicas, ou seja, imagens dentro da pasta 'imagens', .txt em uma pasta "textos" e pdf em uma pasta "PDF". Escreva um arquivo de comandos (.bat) com comandos para mover os arquivos de dentro das pastas para dentro da pasta 'destino', também separados por tipo, renomeando cada um em ordem sequencial (1.jpg, 2.jpg, ...). No final, a estrutura deverá ficar assim:

```
C:\USERS\PANCA\DESKTOP\EXERCICIO
├── destino
│   ├── imagens
│   ├── PDF
│   └── textos
├── imagens
├── PDF
└── textos
exercico.bat
```

[Download do Template](#)