

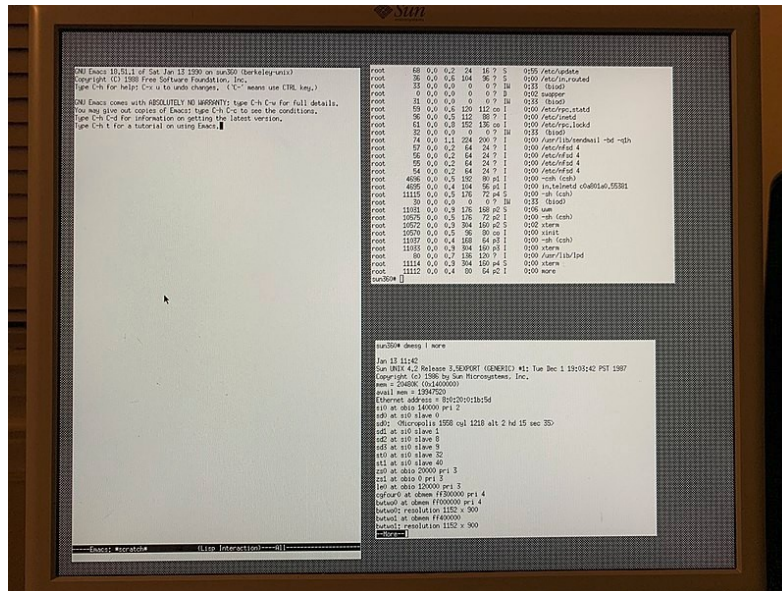
**Esperar até 19:10**

# Bash/Terminal

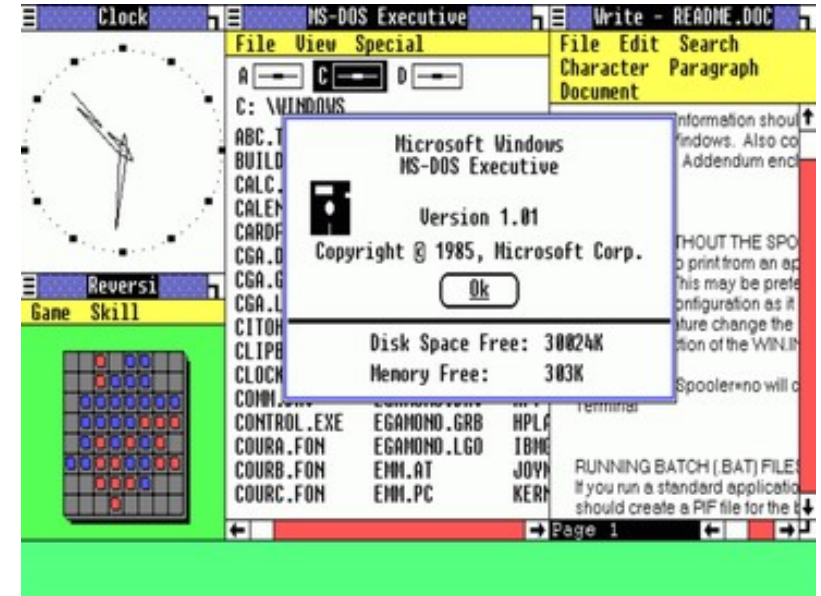
# O que é um terminal



# Interfaces Gráficas



## X Windows (X11)



# Microsoft Windows 1.0

# Shell vs Terminal

- Shell: Interface que interage com o sistema operacional
  - Exemplo: Interface gráfica
  - Por motivos históricos, geralmente se refere a programas de terminal
- Terminal: Interface baseada em apenas texto
  - Não muito usada no Windows
  - Muito importante em sistemas POSIX (MacOS, Linux, FreeBSD)

# Bash (Bourne Again SHell)

- Linguagem de scripting
- Usada por padrão no Ubuntu (e no WSL2)
- Pode ser utilizada para automatizar comandos
- Sintaxe muito “feia”

# Alguns comandos em Bash

- `ls`: Lista todos arquivos na pasta atual (`dir` no windows)
- `cd pasta`: Muda o terminal para a pasta escolhida
- `mv arquivo pasta`: Move um arquivo para a pasta
- `rm arquivo`: deleta um arquivo

# Linguagem de Scripting

- Podem automatizar sequências de comandos
- Ligar programas desconectados
- Rodar experimentos ou testes
- Bash (.sh) vs Batch (.bat)



# Demonstração Arquivo .sh

# “Tudo é um Arquivo”

- Abstração POSIX
- Interfaces de entrada ou saída são abstraídas com arquivo
  - Input do usuário (stdin), drivers de dispositivos, soquetes de rede
  - Tela do terminal (stdout)
- Facilita o desenvolvimento de programas

# Redirecionamento de entrada/saída

- Saída de um programa para um arquivo
- Arquivo para a entrada de um programa
- `./tp > saida.txt`
- `./tp < entrada.txt`
- `./tp < entrada.txt > saida.txt`

# Redirecionamento de Erros

- stderr: Uma segunda saída “secreta”
- Geralmente imprime na tela igual ao stdout
- Todos erros são impressos aqui
- Alguns programas também (valgrind)
- `./tp 2> erros.txt`

# **Demonstração de Redirecionamento**

# Pipes

- Redirecionam a saída de um programa direto pra outro
- Evita criar arquivos intermediários
- `./gerar_entrada | ./tp`

# Variáveis

- Sintaxe estranha: \$variavel
- O sistema tem algumas especiais
  - \$PATH, \$HOME, \$USER
- Valor pode ser inserido no meio de um script
- `echo $USER`

# Inserção de saída

- Insere a saída de um comando direto em outro comando
- Útil para passar argumentos de linha de comando
- `./tp $(./gera_argumentos)`



# Demonstração Variáveis

# Loops

- Rodar a mesma sequencia de comandos repetidas vezes
- Mesmo comando para sequencia de arquivos

```
for entrada in entrada1 entrada2 entrada3  
do  
    ./tp < $entrada  
done
```

# Demonstração Loop

# É isto

## Dúvidas?