

Semana 0 - Aula 01

Conhecendo o Computador e o Terminal

Labenu_



O que vamos ver hoje?

- Começaremos fazendo um **pequeno histórico** sobre a computação
- Veremos uma **analogia** para se explicar **programação**
- Vamos nos familiarizar com um dos programas mais famosos para um programador: o **terminal**



Um pouco de história

Labenu_



Surgimento 🎉

- Na a Segunda Guerra Mundial, **Alan Turing** criou um modelo **teórico** da **Máquina de Turing**
 - Capaz de seguir instruções
 - Realizar cálculos complexos
- Protótipo de máquina que seria capaz de quebrar códigos de guerra alemães
 - **Filme: O jogo da imitação**



Surgimento 🎉



- **Máquina de Turing**
 - Fita escrita que pode ser **lida, escrita** ou **movimentada**
 - Utiliza uma **tabela lógica** para realizar estas operações
- Com essa estrutura, é possível **calcular qualquer coisa** que os computadores atualmente conseguem



Analogia da Cozinha

Labenu_



Analogia da cozinha 🍝

- **Programa** ou **software** é sequência de instruções a serem executadas a fim de produzir um resultado
- **Receita** é sequência de ações a serem realizadas a fim de produzir um prato



Analogia da cozinha 🍪

- **Memória:** armazena programas e dados utilizados por eles
- **Armário:** armazena receitas e ingredientes utilizados por elas



Analogia da cozinha 🍳

- **Processador:** lê os programas, buscando e salvando dados na memória
- **Cozinheiro:** lê a receita; e prepara os pratos, pegando e guardando ingredientes no armário



Computadores hoje

Labenu_



E hoje em dia?

- **Máquinas poderosas**
 - Cálculos complexos
 - Análise de dados
 - Jogos
 - Trabalho (textos, planilhas e apresentações)
- **Simples e intuitivo**, focado em usuários "leigos"
- Máquina que executa **programas**



E hoje em dia?

- **Sistemas Operacionais**
 - **Programas responsáveis** por gerenciar o computador
 - Organiza **vários programas** rodando ao mesmo tempo
 - Gerencia o **sistema de arquivos** e **pastas**



E hoje em dia?

- **Sistemas Operacionais**

- Apresenta **interface** para o computador, o que permite **interação** do usuário
- **Seção especial** do armário (memória) que guarda os materiais administrativos do restaurante: regras de limpeza e organização da cozinha, cardápio, etc.



E hoje em dia?

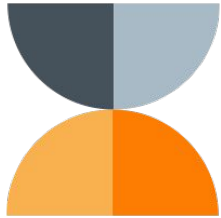
- **Interfaces**

- Todos os **comandos** eram por **texto**, por meio de uma *Command Line Interface* (CLI)
 - Terminal
- Com o tempo, foram desenvolvidas interfaces **mais amigáveis** e **intuitivas**, chamadas de *Graphical User Interface* (GUI)



Pausa para relaxar 🤔

5 min



- O computador é uma máquina lógica. O **processador** é responsável pela execução de **programas**, acessando a **memória** da máquina
 - **Sistemas Operacionais** são programas que criam **interfaces** amigáveis para o usuário
 - **Terminal** é a maneira de criar **comandos** de texto



Terminal

Labenu_



Terminal

- Programa que permite interação com o computador por meio de comandos (**CLI**)
- Existem vários terminais, alguns exemplos abaixo:
 - Bash
 - Zsh
 - CMD
 - PowerShell
 - **GitBash**



Terminal

- Os terminais dependem do **Sistema Operacional**
- **Sistemas Unix (Linux e MacOS)**
 - Herdaram o terminal da época em que não existiam interfaces gráficas
 - **Bash/Zsh**: um terminal muito poderoso



Terminal

- ***Windows***

- Foi desenvolvido com foco na **interface gráfica**
- Os **terminais** disponíveis são, em geral, **menos usuais**
- Usaremos, então, o **GitBash** que simula o bash no *Windows*



Terminal

- **Por que o terminal?**
 - Alguns programas **não** possuem **interface gráfica**
 - Permite **automatização** de tarefas
 - Força **maior entendimento** por parte do usuário
 - É bem mais **rápido** e **eficiente** que a interface gráfica



Comandos

Labenu_



Comandos

- **O que são comandos?**

- Comandos nada mais são do que um programa que executa uma ação
- Ele pode receber **argumentos** e **opções**
 - **argumentos**: são valores que mudam de acordo com o que o usuário quer
 - **opções**: são argumentos específicos do comando



Comandos

- **Exemplo**

- Visualizar as primeiras 5 linhas de um arquivo chamado *meu-arquivo*

```
head -n 5 meu-arquivo.txt
```



Comandos

- **Comandos básicos**

- whoami

- Imprime o nome do usuário na tela

```
whoami # retorna o nome do usuário atual ex: 'seuUsuario'
```



Comandos

- **Comandos básicos**

- echo

- Imprime algo no terminal



```
echo "Hello World" # imprime Hello World no terminal
```



Comandos

- **Comandos básicos**

- pwd

- Sigla para *print working directory*
 - Mostra o diretório/pasta em que estamos



```
pwd # retorna a pasta que o terminal está atualmente ex: '/c/Users/seuUsuario'
```



Comandos

- Comandos básicos

- cd

- Sigla para *change directory*
 - Muda o diretório/pasta em que estamos



```
cd ./minha-pasta # troca o diretório atual para a subpasta 'minha-pasta'  
cd - # volta para a última pasta  
cd # vai para a pasta "home" do usuário atual  
cd ../ # vai para a pasta acima da atual
```



Comandos

- Comandos básicos

- ls

- O nome vem de *list*
 - Lista os arquivos e diretórios do diretório em que estamos

```
ls # retorna nome de arquivos e pastas presentes na pasta atual
ls -a # retorna nome de arquivos e pastas, incluindo os ocultos (cujo nome começa com `.`)
ls -l # retorna informações de arquivos e pastas, incluindo tamanho, proprietário e outras informações
ls -la # soma dos dois modificadores anteriores
```



Comandos

- **Comandos básicos**

- touch
 - Cria um novo arquivo



```
touch index.html # criará um arquivo chamado index.html na pasta atual
```



Comandos

- **Comandos básicos**

- mkdir

- Sigla para *make directory*
 - Cria um novo diretório



```
mkdir minha-pasta # cria uma pasta chamada 'minha-pasta' no diretório atual
```



Comandos

- Comandos básicos

- rm

- Vem da palavra **remove**
 - Possibilita apagar arquivos
 - Ele apaga de uma maneira **IRREVERSÍVEL** e **SEM PEDIR CONFIRMAÇÃO**



```
rm ./meu-arquivo-gigante.txt # remove imediatamente o arquivo 'meu-arquivo-gigante.txt'
rm -r ./minha-pasta # remove Recursivamente todos os arquivos e sub-pastas da 'minha-pasta'
```



Comandos

- Comandos básicos

- mv

- Vem da palavra ***move***
 - Permite mover arquivos de um diretório para outro
 - Pode ser usado para renomear arquivos

```
mv ./meu-arquivo-gigante.txt ./minha-sub-pasta # move 'meu-arquivo-gigante' para 'minha-sub-pasta'
mv ./meu-arquivo-gigante.txt ./meu-gigante.txt # renomeia 'meu-arquivo-gigante.txt' para 'meu-gigante.txt'
```



Comandos

- Comandos básicos

- cp

- Vem da palavra **copy**
 - Copia arquivos de um diretório para outro

```
cp ./meu-arquivo-gigante.txt ./minha-sub-pasta # copia 'meu-arquivo-gigante' para 'minha-sub-pasta'
```



Comandos

- Comandos básicos

- cat

- Vem da palavra *con**cat***
 - Ele concatena tudo que está no arquivo e imprime no terminal



```
cat meu-arquivo-gigante.txt # imprime o conteúdo do arquivo 'meu-arquivo-gigante.txt'
```



Comandos

- **Comandos básicos**

- head

- Imprime as 10 primeiras linhas de um arquivo
 - A opção -n permite indicar quantas linhas queremos ver



```
head meu-arquivo-gigante.txt # imprime as 10 primeiras linhas do arquivo 'meu-arquivo-gigante.txt'  
head -n 20 meu-arquivo-gigante.txt # imprime as 20 primeiras linhas do arquivo 'meu-arquivo-gigante.txt'
```



Comandos

- **Comandos básicos**

- tail

- Imprime as 10 últimas linhas de um arquivo
 - A opção -n permite indicar quantas linhas queremos ver



```
tail meu-arquivo-gigante.txt # imprime as 10 últimas linhas do arquivo 'meu-arquivo-gigante.txt'  
tail -n 20 meu-arquivo-gigante.txt # imprime as 20 últimas linhas do arquivo 'meu-arquivo-gigante.txt'
```



Comandos

- **Comandos básicos**

- grep

- Permite buscar o conteúdo de um arquivo



```
grep Future4 ./lista-de-empresas.txt # Busca pela palavra Future4 no arquivo lista-de-empresas.txt e  
imprime toda a linha encontrada
```

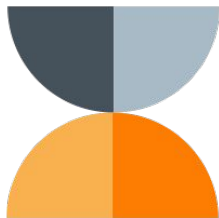


Pausa para relaxar 🤔

10 min

Vimos os seguintes comandos:

- *whoami*
- *echo*
- *pwd*
- *cd*
- *ls*
- *touch*
- *mkdir*
- *rm*
- *mv*
- *cp*
- *cat*
- *head*
- *tail*
- *grep*




Vamos Praticar!

Labenu_



Comandos

- Ex 1: Navegar até a pasta *aula1*



```
pwd # imprime o diretório atual  
ls # mostra todos os arquivos da pasta atual  
cd aula1 # entra na pasta aula1
```



Comandos

- Ex 2: Criar um novo diretório, chamado *minha-pasta* e um novo arquivo, *meu-arquivo*, ambos na pasta *aula1*



```
touch meu-arquivo # cria um arquivo chamado meu-arquivo  
mkdir minha-pasta # cria uma pasta chamada minha-pasta
```



Comandos

- Ex 3: Copiar o arquivo *meu-arquivo* para a pasta *minha-pasta*




```
cp meu-arquivo minha-pasta # copia meu-arquivo para minha-pasta
```



Comandos

- Ex 4: Renomear o arquivo *meu-arquivo* para *meu-arquivo2*

```
 mv meu-arquivo meu-arquivo2 # renomeia o meu-arquivo para meu-arquivo2
```



Comandos

- Ex 5: Mover o arquivo *meu-arquivo2* para a *minha-pasta*, com o nome *meu-arquivo3* com só um comando



```
mv meu-arquivo2 minha-pasta/meu-arquivo3 # move o arquivo para dentro da minha-pasta, e renomeia para meu-arquivo3
```



Comandos

- Ex 6: Entrar na *minha-pasta* e apagar o arquivo *meu-arquivo3*



```
cd minha-pasta  
rm meu-arquivo3 # apaga o arquivo meu-arquivo3
```



Comandos

- **Ex 7: Voltar para a pasta *aula1* e apagar a *minha-pasta***



```
cd ..  
rm minha-pasta # gera um erro, pois é uma pasta e não um arquivo  
rm -r minha-pasta # apaga a pasta recursivamente, ou seja, a pasta e tudo que está dentro
```



Comandos

- Ex 8: Imprimir na tela o conteúdo do arquivo *notas*

```
cd Labenu  
cat notas.txt
```



Comandos

- Ex 9: Imprimir as 10 primeiras e as 5 últimas linhas do arquivo notas

```
head notas  
tail -n 5 notas
```



Comandos

- Ex 10: Descobrir a nota de uma pessoa específica

```
grep "nome" notas.txt
```



Comandos

- **Ex 11: Descobrir nomes e notas de 2 estudantes seguintes à pessoa do exercício anterior**

```
grep -A 2 "seu-nome" notas
```



Resumo

Labenu_



Resumo



- O computador é uma máquina lógica. O processador é responsável pela execução de programas, acessando a memória quando necessário
- Sistemas Operacionais são programas que criam interfaces amigáveis para o usuário
- Terminal é uma maneira de criar comandos de texto



Resumo

- Vimos os seguintes comandos:
 - *whoami*
 - *echo*
 - *man*
 - *pwd*
 - *cd*
 - *ls*
 - *touch*
 - *mkdir*
 - *rm*
 - *mv*
 - *cp*
 - *cat*
 - *head*
 - *tail*
 - *grep*



Dúvidas? 🧐

Labenu_



Labenu_



Obrigad@!