

1) Comandos básicos:

1.) Navegue pelo terminal até chegar na pasta Desktop;

```
rhuan@rhuan-VirtualBox:~$ dir
Área\ de\ trabalho  Downloads  Modelos  Público  Vídeos
Documentos          Imagens    Músicas  snap
rhuan@rhuan-VirtualBox:~$ cd Área\ de\ trabalho
rhuan@rhuan-VirtualBox:~/Área de trabalho$
```

2.) Crie um novo diretório na pasta Desktop;

```
rhuan@rhuan-VirtualBox:~/Área de trabalho$ mkdir nova_pasta
rhuan@rhuan-VirtualBox:~/Área de trabalho$ dir
nova_pasta
```

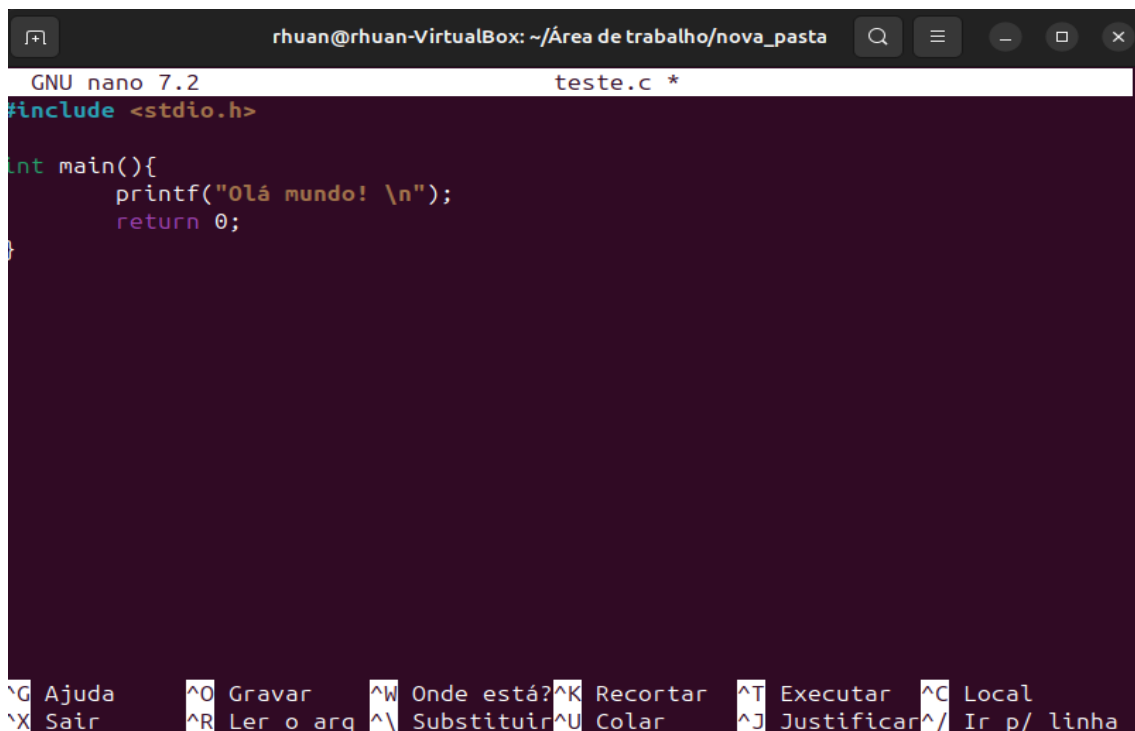
3.) Acesse esse novo diretório;

```
rhuan@rhuan-VirtualBox:~/Área de trabalho$ cd nova_pasta
rhuan@rhuan-VirtualBox:~/Área de trabalho/nova_pasta$ S
```

4.) Crie um arquivo chamado teste.c;

```
rhuan@rhuan-VirtualBox:~/Área de trabalho/nova_pasta$ touch teste.c
rhuan@rhuan-VirtualBox:~/Área de trabalho/nova_pasta$ nano teste.c
```

5.) Digite o seguinte código:



```
rhuan@rhuan-VirtualBox: ~/Área de trabalho/nova_pasta
GNU nano 7.2                               teste.c *
#include <stdio.h>

int main(){
    printf("Olá mundo! \n");
    return 0;
}

^G Ajuda      ^O Gravar     ^W Onde está? ^K Recortar    ^T Executar   ^C Local
^X Sair       ^R Ler o arq  ^\ Substituir ^U Colar      ^J Justificar ^/ Ir p/ linha
```

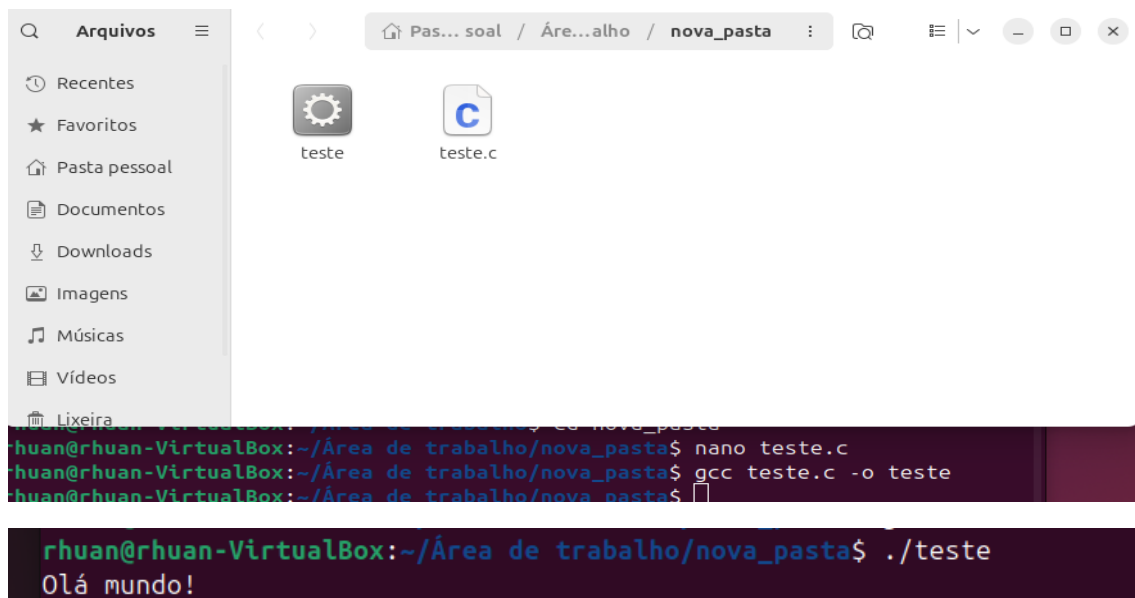
6.) Salve o arquivo;

```
ADRI 1+1 ~/Área de trabalho/nova_pasta

#include <stdio.h>

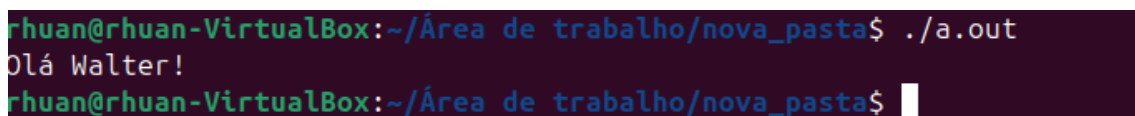
int main(){
    printf("Olá mundo! \n");
    return 0;
}
```

7.) No terminal, execute os seguintes comandos:



8.) Explique o funcionamento dos comandos da linha 7. Altere o código de 5 para outro programa e veja o funcionamento.

R: O comando “gcc teste.c -o teste” traduz o código-fonte “teste.c” para um arquivo executável. O compilador gera automaticamente um arquivo chamado “teste” e o “./teste” executa esse código compilado.



9.) Crie um novo arquivo chamado teste.py



10.) Digite o seguinte código:

```
GNU nano 7.2
print("Olá mundo")
```

11.) No terminal, execute os seguintes comandos:

```
rhuan@rhuan-VirtualBox:~/Área de trabalho/nova_pasta$ python3 teste.py
```

12.) O que ocorre?

```
Olá mundo
rhuan@rhuan-VirtualBox:
```

13.) No terminal digite apenas o comando python. O que acontece? Qual a diferença deste caso para o do item 7?

```
rhuan@rhuan-VirtualBox:~/Área de trabalho/nova_pasta$ python3
Python 3.12.3 (main, Feb  4 2025, 14:48:35) [GCC 13.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>
```

R: O terminal mostra a versão instalada do Python e ele abre o interpretador python, no qual posso escrever e executar os códigos diretamente dele. Diferente no caso 7, pois a linguagem C é compilada, então o terminal tem que criar um arquivo executável do código fonte e executar ele para iniciar o código.

2) Comandos básicos:

Crie uma nova pasta e realize o seguinte procedimento:

1.) Digite o comando git init para iniciar o versionamento;

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
```

- 2.) Digite o comando git status, para verificar o estado do versionamento. O que aparece?

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ git status
No ramo master

No commits yet

nada para enviar (crie/copie arquivos e use "git add" para registrar)
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$
```

- 3.) Crie um novo arquivo (como por exemplo um dos criados na seção anterior).

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ nano teste.py
```

- 4.) Use novamente o comando git status. Houve alguma diferença?

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ git status
No ramo master

No commits yet

Arquivos não monitorados:
  (utilize "git add <arquivo>..." para incluir o que será submetido)
    teste.py

nada adicionado ao envio mas arquivos não registrados estão presentes (use "git add" to registrar)
```

- 5.) Use o comando git add . (o ponto faz parte do comando). Utilize novamente o git status. Houve alteração?

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ git status
No ramo master

No commits yet

Mudanças a serem submetidas:
  (utilize "git rm --cached <arquivo>..." para não apresentar)
    new file:   teste.py
```

- 6.) Utilize o comando git commit -m" onde você descreva a operação que você fez.

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ git commit -m "Meu primeiro commit do Lab1"
[master (root-commit) b3c08c6] Meu primeiro commit do Lab1
1 file changed, 1 insertion(+)
create mode 100644 teste.py
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$
```

- 7.) Use novamente o git status e verifique a alteração

```
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$ git status
No ramo master
nothing to commit, working tree clean
rhuan@rhuan-VirtualBox:~/Documentos/Nova_pasta$
```