

Git e Github

Git

Git

É um sistema de versionamento distribuído

Criado por Linus Torvalds em 2005

É usado primariamente para desenvolvimento de software, mas pode ser usado para outros tipos de projeto

Como fazer para usar o Git?

É preciso instalar em seu computador um sistema Git

Alguns sistemas já vem com o Git instalado

É possível manipular o Git por terminal de comando (modo console) ou por um aplicativo gráfico (modo GUI)

Vamos usar somente por terminal de comando – padrão para todas as plataformas.

git --distributed-is-the-new-centralized

🔍 Search entire site...

Git is a [free and open source](#) distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is [easy to learn](#) and has a [tiny footprint with lightning fast performance](#). It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like [cheap local branching](#), convenient [staging areas](#), and [multiple workflows](#).



About

The advantages of Git compared to other source control systems.



Documentation

Command reference pages, Pro Git book content, videos and other material.



Downloads

GUI clients and binary releases for all major platforms.



Community

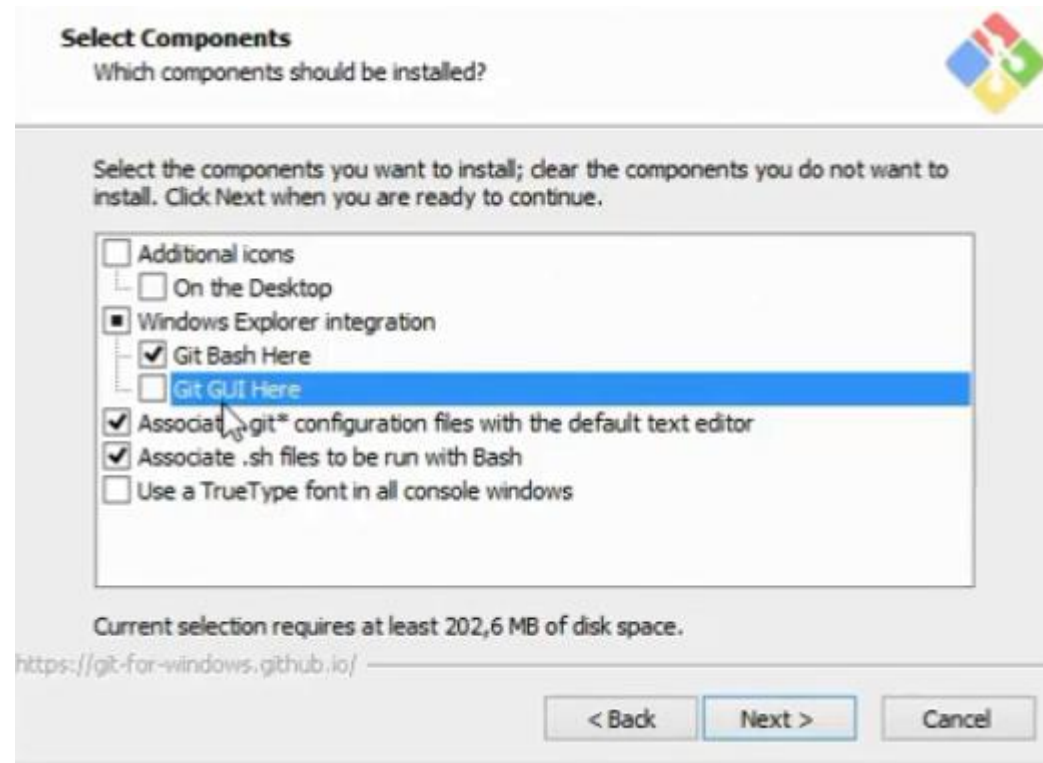
Get involved! Bug reporting, mailing list, chat, development and more.



Instalação Git



Instalação Git



Instalação Git

Marcar esta opção



Adjusting your PATH environment
How would you like to use Git from the command line?

☐ **Use Git from Git Bash only**
This is the safest choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.

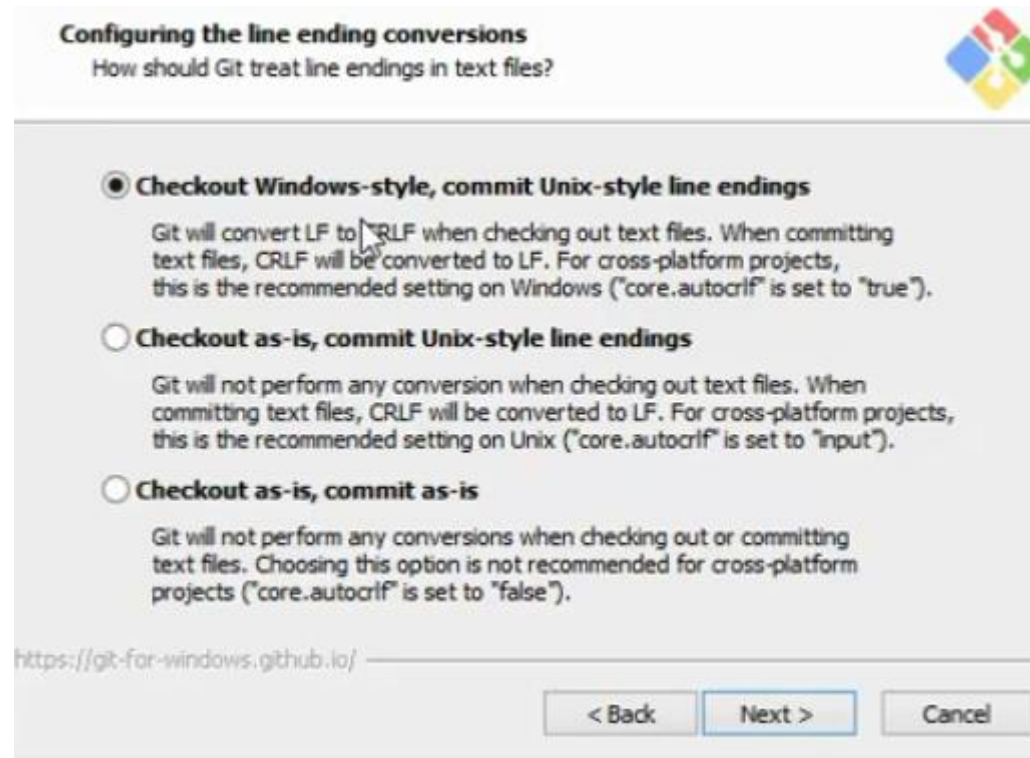
☒ **Use Git from the Windows Command Prompt**
This option is considered safe as it only adds some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from both Git Bash and the Windows Command Prompt.

☐ **Use Git and optional Unix tools from the Windows Command Prompt**
Both Git and the optional Unix tools will be added to your PATH.
Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.

<https://git-for-windows.github.io/>

< Back Next > Cancel

Instalação Git



Instalação Git



Instalação Git

Desmarcar esta opção



Configuring extra options
Which features would you like to enable?

☒ **Enable file system caching**
File system data will be read in bulk and cached in memory for certain operations ("core.fscache" is set to "true"). This provides a significant performance boost.

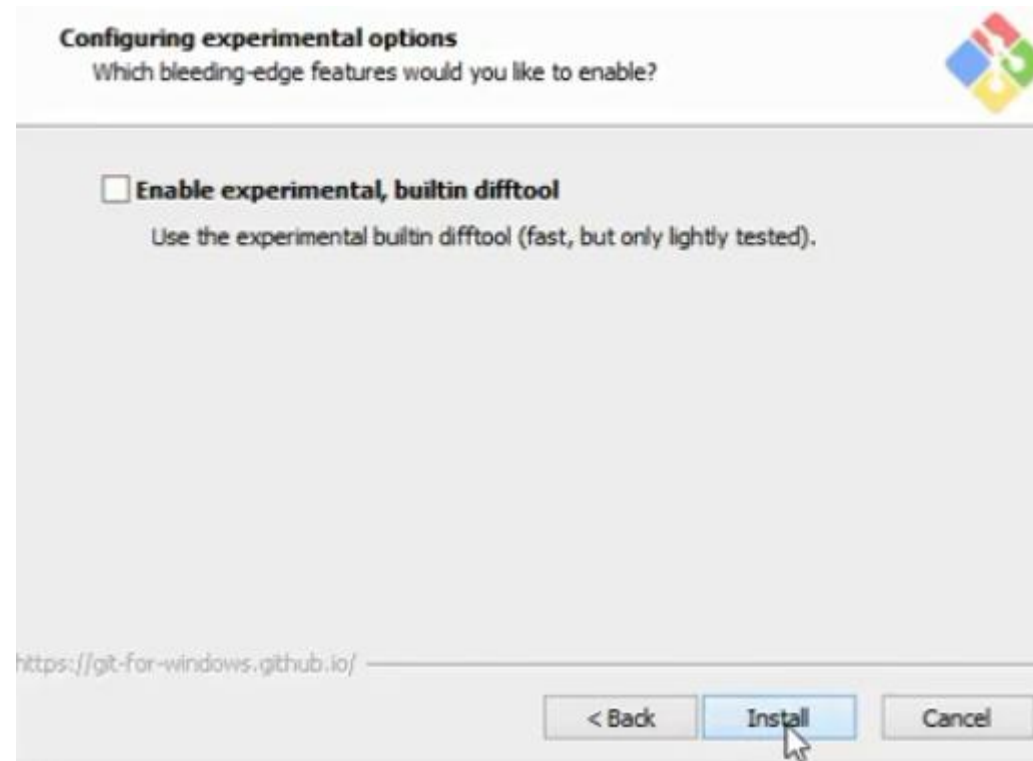
☐ **Enable Git Credential Manager**
The [Git Credential Manager](#) for Windows provides secure Git credential storage for Windows, most notably multi-factor authentication support for Visual Studio Team Services and GitHub. (requires .NET framework v4.5.1 or later).

☐ **Enable symbolic links**
Enable [symbolic links](#) (requires the SeCreateSymbolicLink permission). Please note that existing repositories are unaffected by this setting.

<https://git-for-windows.github.io/>

< Back Next > Cancel

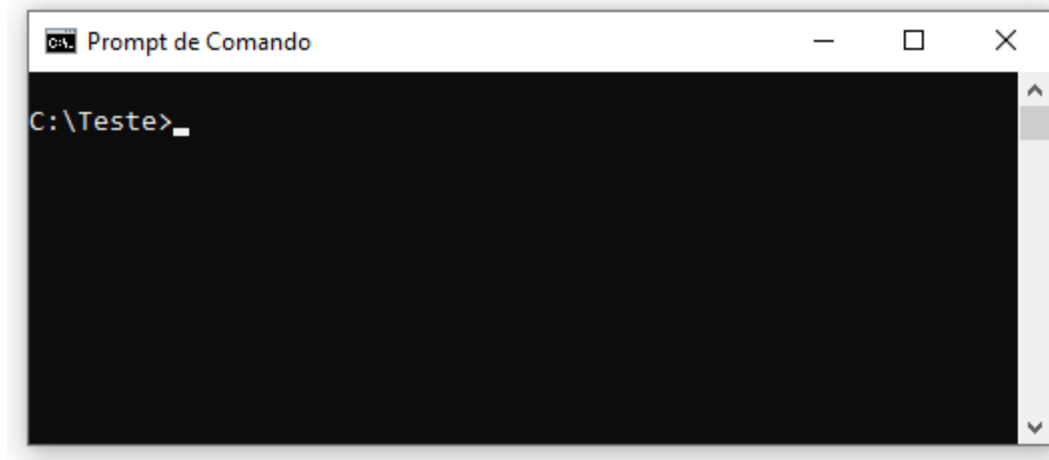
Instalação Git



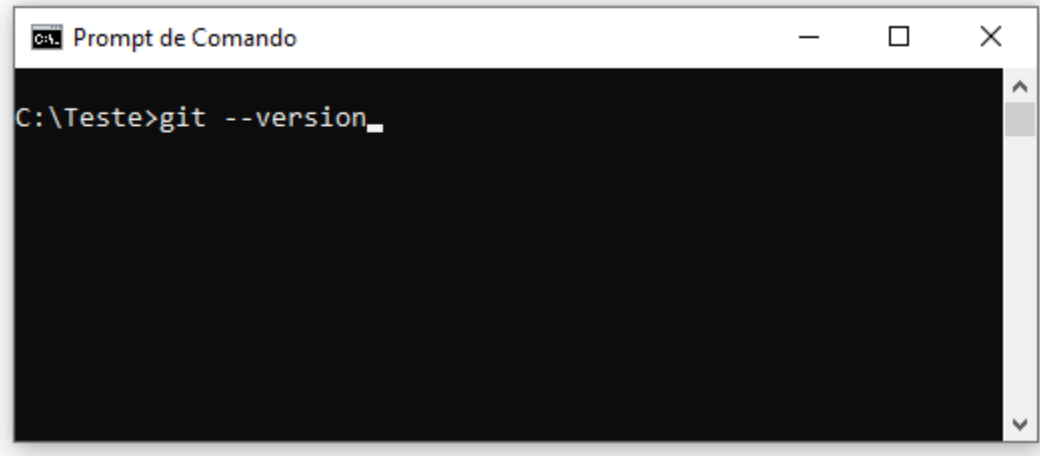
Instalação Git



Testando a instalação Git



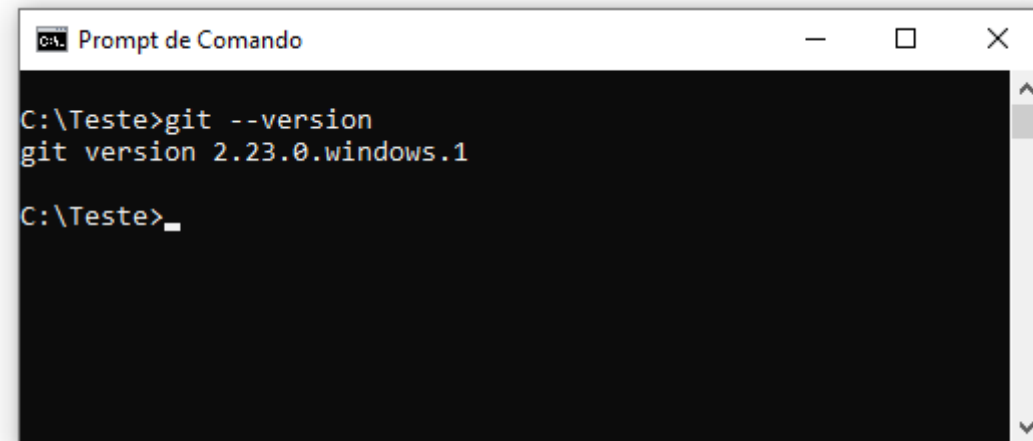
Testando a instalação Git



```
C:\Teste>git --version_
```

The image shows a Windows Command Prompt window titled "Prompt de Comando". The window has a standard Windows title bar with minimize, maximize, and close buttons. The command prompt shows the current directory as "C:\Teste" and the command "git --version_" is being entered. The cursor is at the end of the command line.

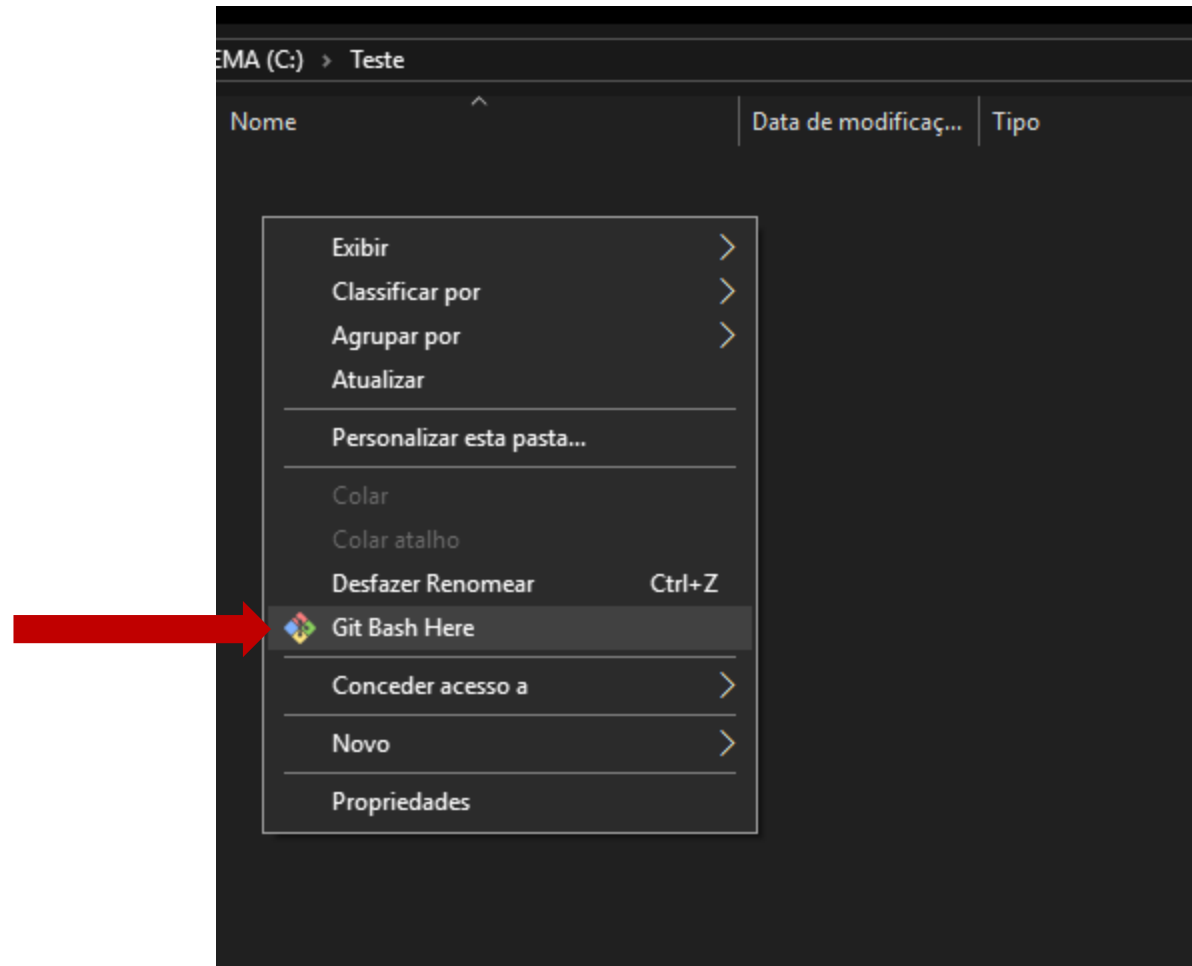
Testando a instalação Git



```
C:\Teste>git --version
git version 2.23.0.windows.1
C:\Teste>
```

The image shows a Windows Command Prompt window titled "C:\Teste Prompt de Comando". The window has a standard Windows title bar with minimize, maximize, and close buttons. The command prompt shows the command `git --version` being executed, which returns the output `git version 2.23.0.windows.1`. The prompt then returns to `C:\Teste>` with a cursor.

Testando a instalação Git



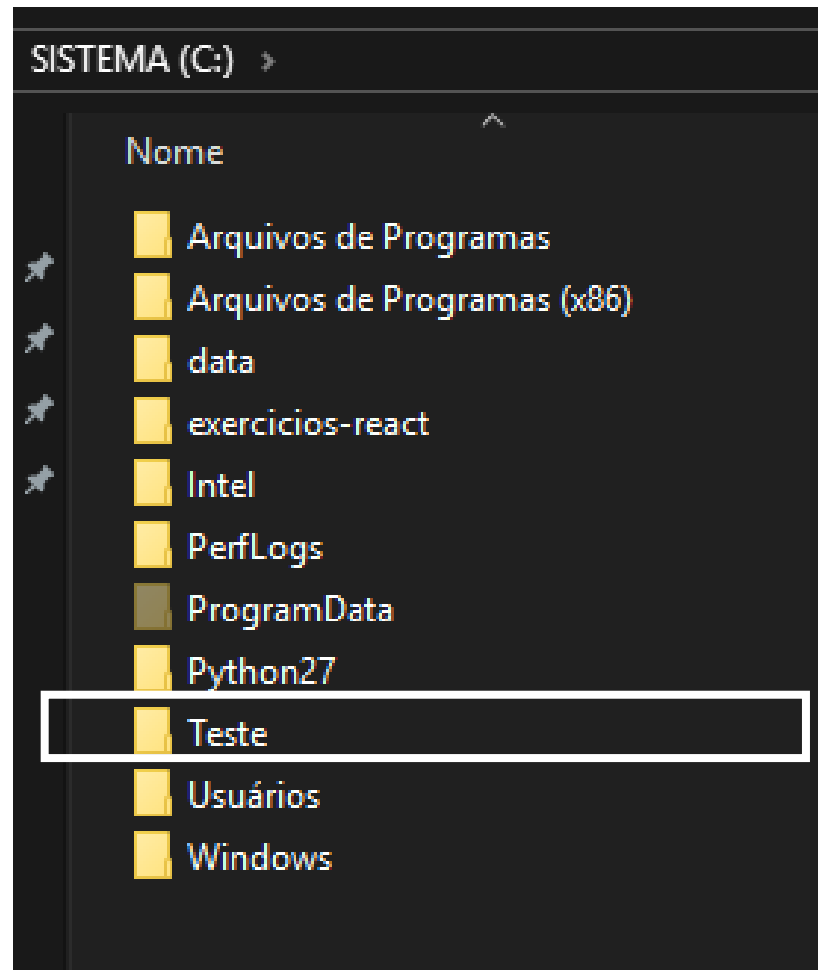
Como criar um novo repositório Git

O que é um repositório Git?

Pasta onde está o projeto que desejamos fazer o controle de versão.

Para cada projeto diferente deverá ser criado um Repositório Git.

Como criar um novo repositório Git

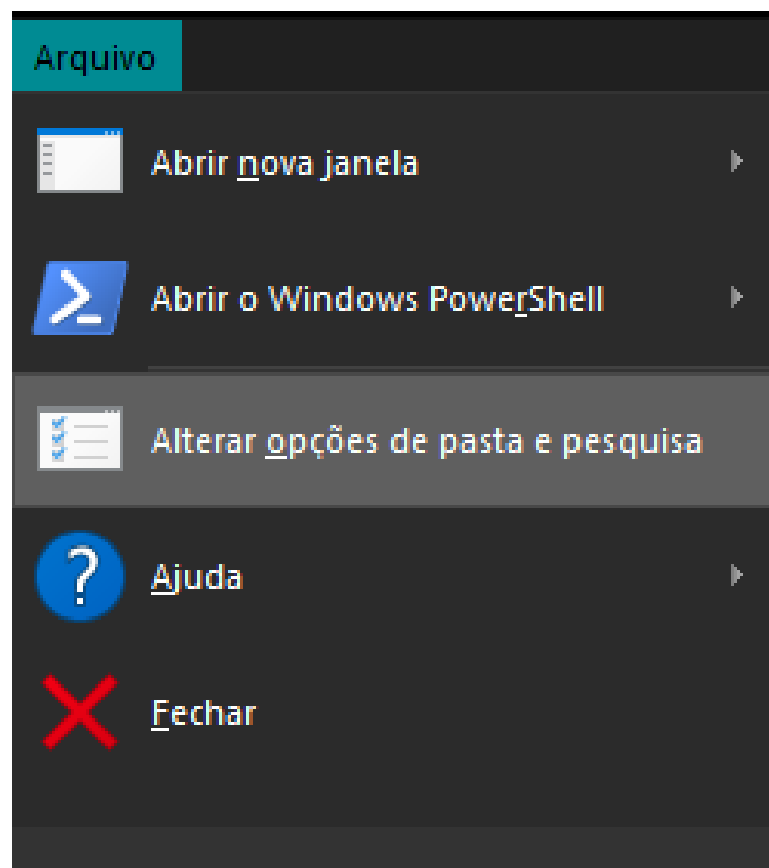


Como criar um novo repositório Git

1) Recomendável: ative a visualização de arquivos ocultos e extensões

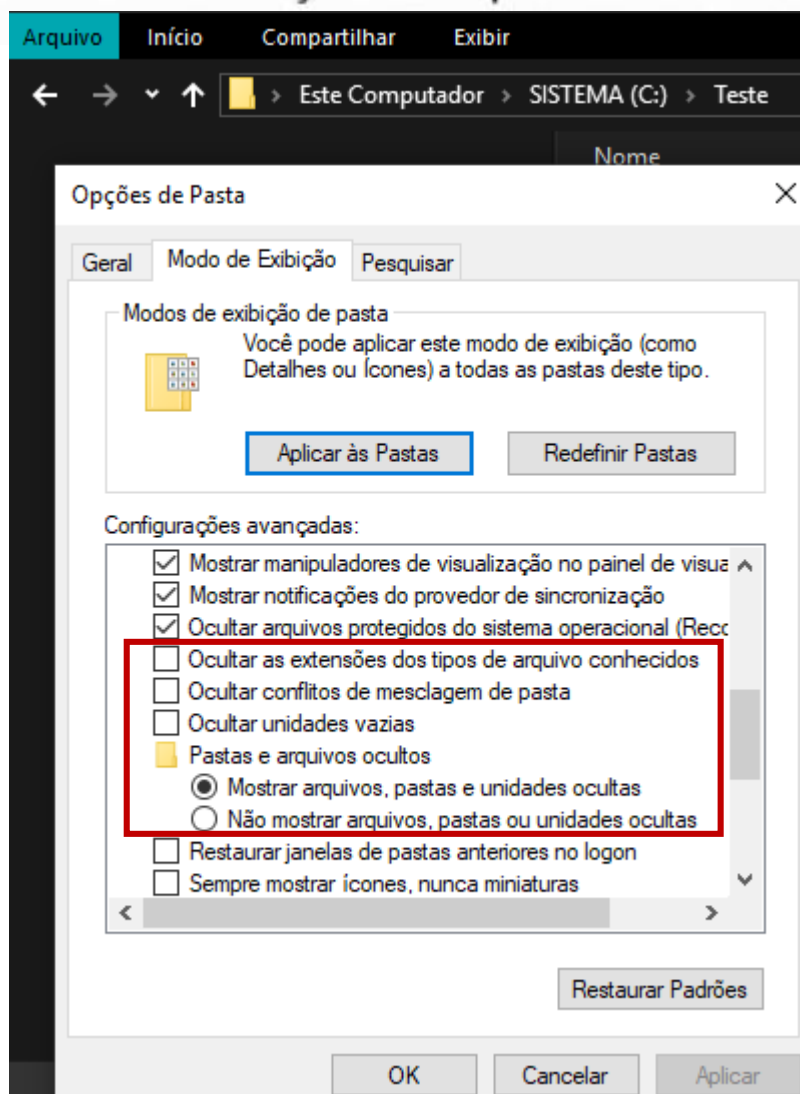
Como criar um novo repositório Git

1) Recomendável: ative a visualização de arquivos ocultos e extensões



Como criar um novo repositório Git

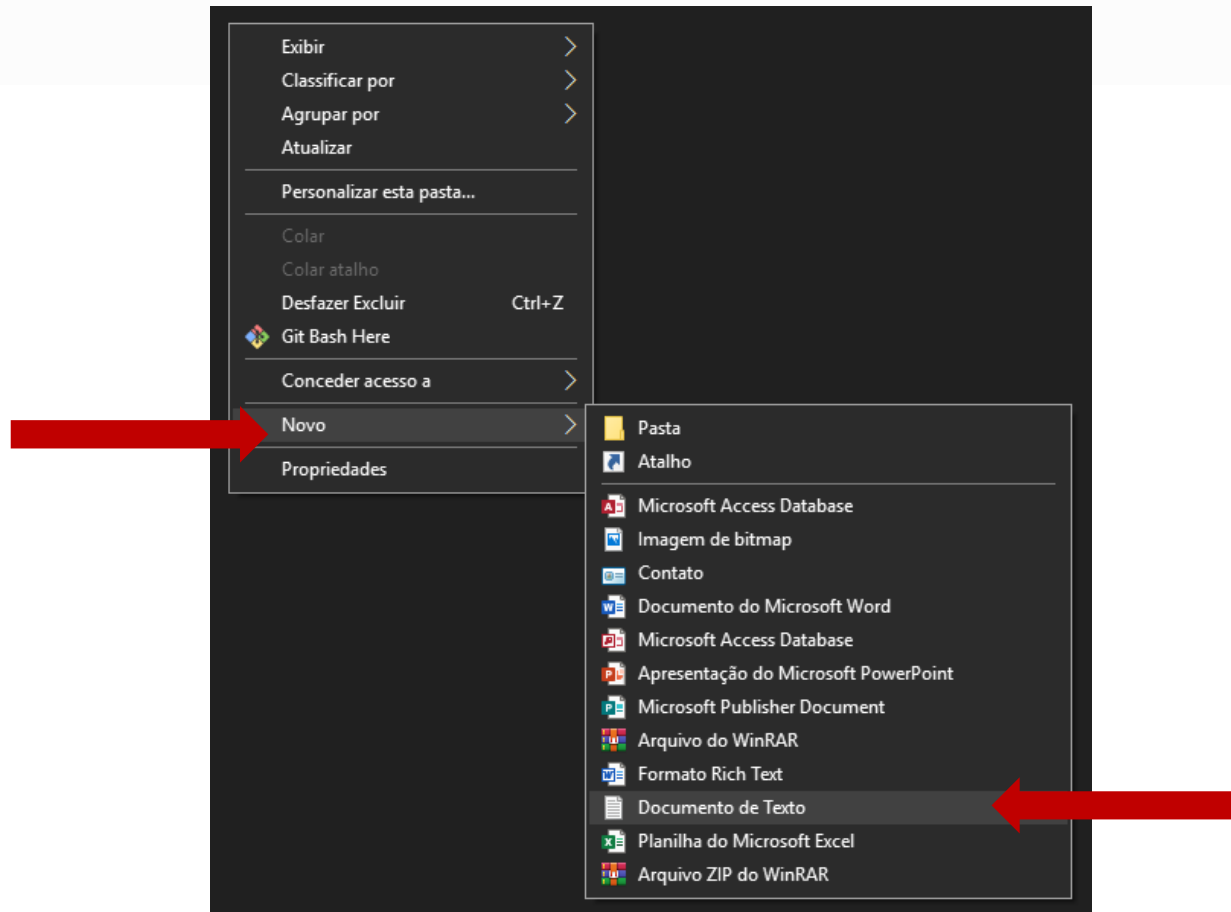
1) Recomendável: ative a visualização de arquivos ocultos e extensões



Como criar um novo repositório Git

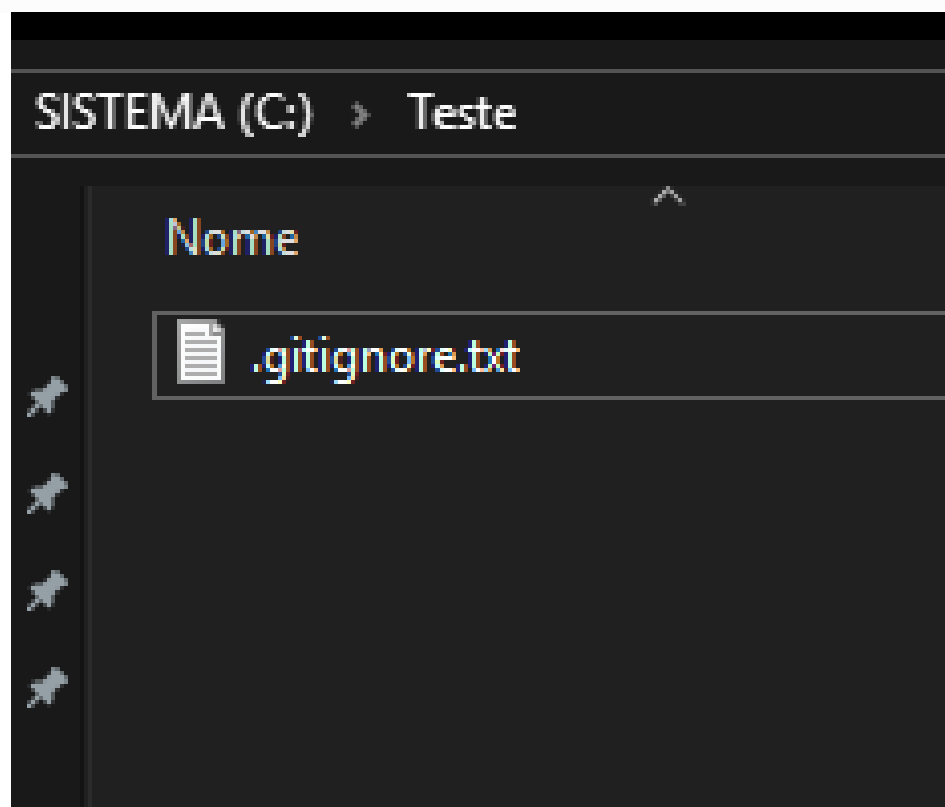
1) Recomendável: ative a visualização de arquivos ocultos e extensões

2) Inclua um arquivo .gitignore na pasta inicial do repositório, contendo as especificações apropriadas para seu tipo de projeto



Como criar um novo repositório Git

- 1) Recomendável: ative a visualização de arquivos ocultos e extensões
- 2) Inclua um arquivo .gitignore na pasta inicial do repositório, contendo as especificações apropriadas para seu tipo de projeto



O que é o arquivo **.gitignore.txt**?

Um arquivo que lista o que o Git deve **ignorar** quando você estiver trabalhando no repositório, de forma que seja possível evitar a adição de arquivos indesejados no repositório sem dificuldades, tornando os arquivos indesejados totalmente invisíveis ao Git.

O que é o arquivo **.gitignore.txt**?

Um arquivo que lista o que o Git deve **ignorar** quando você estiver trabalhando no repositório, de forma que seja possível evitar a adição de arquivos indesejados no repositório sem dificuldades, tornando os arquivos indesejados totalmente invisíveis ao Git.

O que for informado no arquivo **.gitignore.txt** o git não vai controlar a versão.

O que é o arquivo **.gitignore.txt**?

Quais arquivos da linguagem c que não devem ser versionados?

O que é o arquivo **.gitignore.txt**?

Quais arquivos da linguagem c que não devem ser versionados?



gitignore linguagem c



O que é o arquivo **.gitignore.txt**?

Quais arquivos da linguagem c que não devem ser versionados?

A screenshot of a Google search interface. The search bar contains the text "gitignore linguagem c". Below the search bar, there are tabs for "Todas", "Vídeos", "Imagens", "Shopping", "Notícias", "Mais", "Configurações", and "Ferramentas". The search results show approximately 6,220 results in 0.36 seconds. The first result is titled "GIT - Ignore arquivos já adicionados ao repositório | blog ..." with a URL "https://www.marceloweb.info > git-ignore-arquivos-ja-adicionados-ao-repo...". The second result is titled ".gitignore - Saiba como ignorar arquivos no Git facilmente ..." with a URL "https://fjorgemota.com > gitignore-ou-como-ignorar-arquivos-no-git". The third result is highlighted with a red box and is titled "gitignore/C.gitignore at master · github/gitignore · GitHub" with a URL "https://github.com > github > gitignore > blob > C.giti...". Below the highlighted result, there is a section titled "Itens também pesquisados" with suggestions like "gitignore c++ example", "how .gitignore works", "gitignore template", and "configure gitignore github".

Google

gitignore linguagem c

Todas Vídeos Imagens Shopping Notícias Mais Configurações Ferramentas

Aproximadamente 6.220 resultados (0,36 segundos)

GIT - Ignore arquivos já adicionados ao repositório | blog ...
<https://www.marceloweb.info > git-ignore-arquivos-ja-adicionados-ao-repo...>
5 de jan. de 2018 - Se o objetivo é destravar todos os arquivos que estão no **.gitignore**, primeiro commit as mudanças pendentes, depois execute o seguinte ...

.gitignore - Saiba como ignorar arquivos no Git facilmente ...
<https://fjorgemota.com > gitignore-ou-como-ignorar-arquivos-no-git>
28 de jan. de 2016 - Conheça o **.gitignore**, um arquivo que vai facilitar sua vida com o uso de ...
Linguagem de programação escolhida;; Framework usado; ...

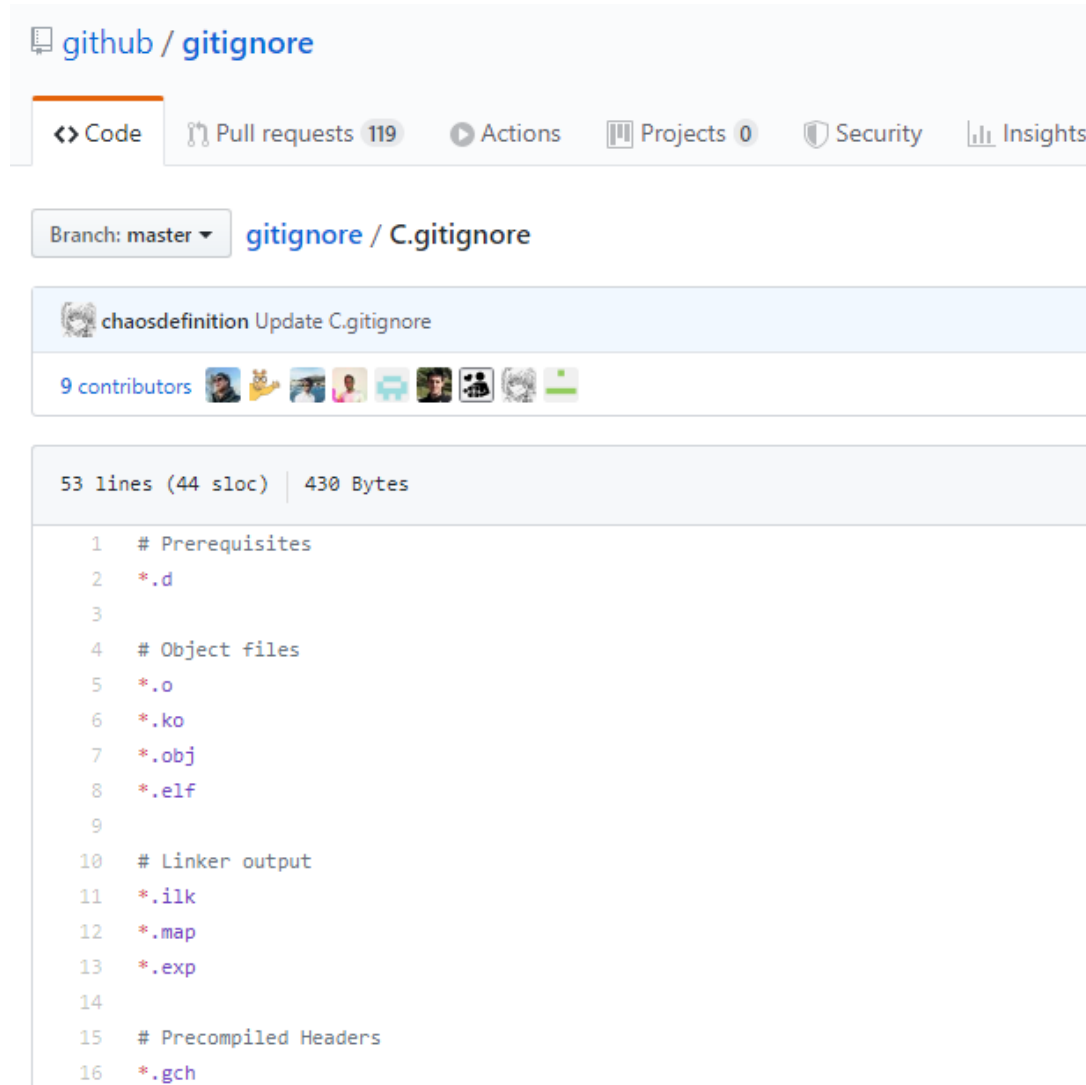
gitignore/C.gitignore at master · github/gitignore · GitHub
<https://github.com > github > gitignore > blob > C.giti...> Traduzir esta página
A collection of useful **.gitignore** templates. Contribute to github/**gitignore** development by creating an account on GitHub.
Não encontrados: linguagem

Itens também pesquisados

gitignore c++ example how .gitignore works
gitignore template configure gitignore github

O que é o arquivo **.gitignore.txt**?

Quais arquivos da linguagem c que não devem ser versionados?



The screenshot shows the GitHub interface for the 'gitignore' repository. The file 'C.gitignore' is selected, showing its content. The file is 430 bytes and contains 53 lines (44 sloc). The content lists various file types to be ignored, including prerequisites, object files, linker output, and precompiled headers.

```
1 # Prerequisites
2 *.d
3
4 # Object files
5 *.o
6 *.ko
7 *.obj
8 *.elf
9
10 # Linker output
11 *.ilk
12 *.map
13 *.exp
14
15 # Precompiled Headers
16 *.gch
```

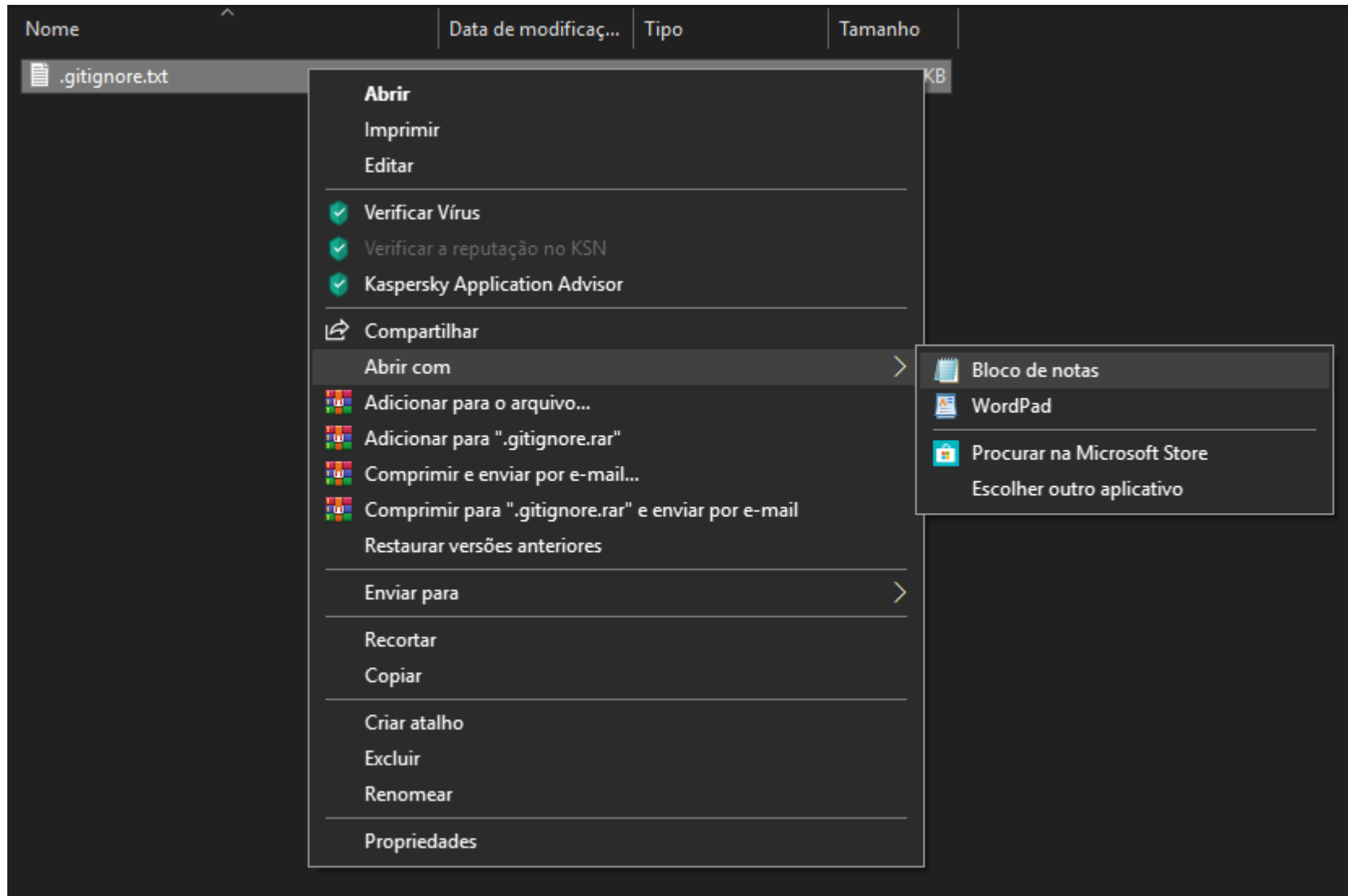
O que é o arquivo **.gitignore.txt**?

Copiar e colar DENTRO do arquivo **.gitignore.txt**.

```
github.com/github/gitignore/blob/master/C.gitignore
16 *.gch
17 *.pch
18
19 # Libraries
20 *.lib
21 *.a
22 *.la
23 *.lo
24
25 # Shared objects (inc. Windows DLLs)
26 *.dll
27 *.so
28 *.so.*
29 *.dylib
30
31 # Executables
32 *.exe
33 *.out
34 *.app
35 *.i*86
36 *.x86_64
37 *.hex
38
39 # Debug files
40 *.dSYM/
41 *.su
42 *.idb
43 *.pdb
44
45 # Kernel Module Compile Results
46 *.mod*
47 *.cmd
48 *.tmp_versions/
49 *.modules.order
50 *.Module.symvers
51 *.Mkfile.old
52 *.kms.conf
```


O que é o arquivo **.gitignore.txt**?

Copiar e colar DENTRO do arquivo **.gitignore.txt**.



O que é o arquivo **.gitignore.txt**?

Copiar e colar DENTRO do arquivo **.gitignore.txt**.

 .gitignore.txt - Bloco de notas

Arquivo Editar Formatar Exibir Ajuda

```
# Prerequisites
```

```
*.d
```

```
# Object files
```

```
*.o
```

```
*.ko
```

```
*.obj
```

```
*.elf
```

```
# Linker output
```

```
*.ilk
```

```
*.map
```

```
*.exp
```

```
# Precompiled Headers
```

```
*.gch
```

```
*.pch
```

```
# Libraries
```

```
*.lib
```

```
*.a
```

```
*.la
```

```
*.lo
```

```
# Shared objects (inc. Windows DLLs)
```

```
*.dll
```

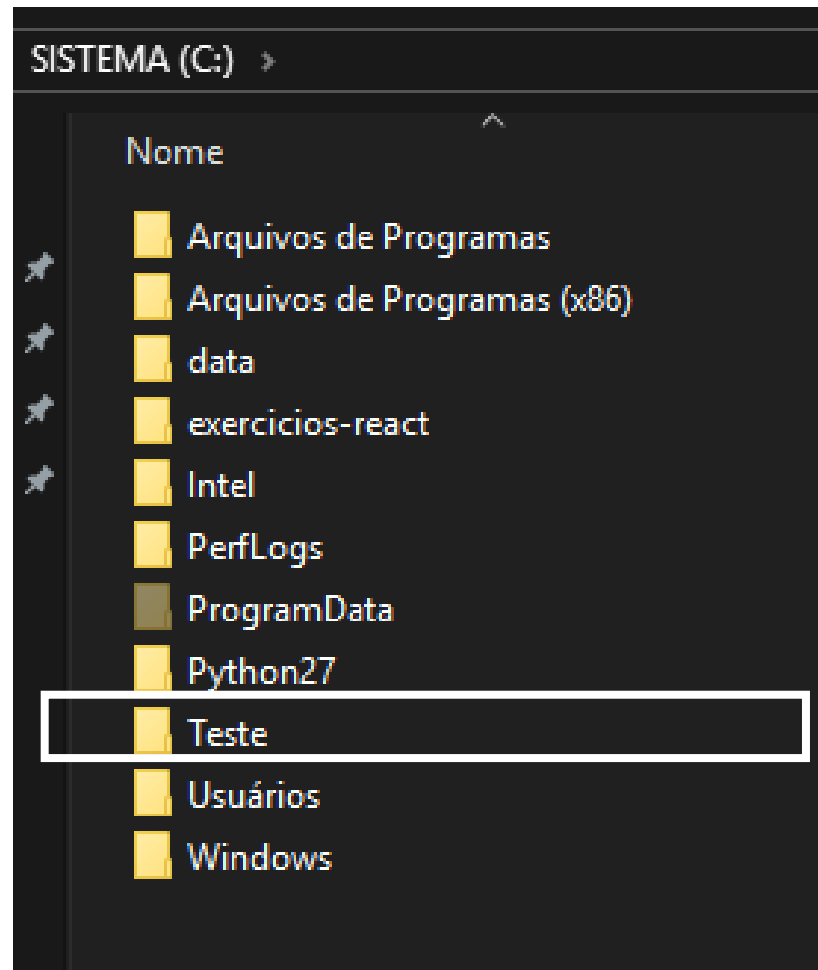
```
*.so
```

```
*.so.*
```

```
*.dylib
```

Controlando as versões

Criar uma função que receba as coordenadas de dois pontos (A e B) e retorne distância entre esses dois pontos. O algoritmo deve ler as coordenadas X e Y de cada um dos pontos e mostrar essa distância.



Controlando as versões

distancia.c

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {

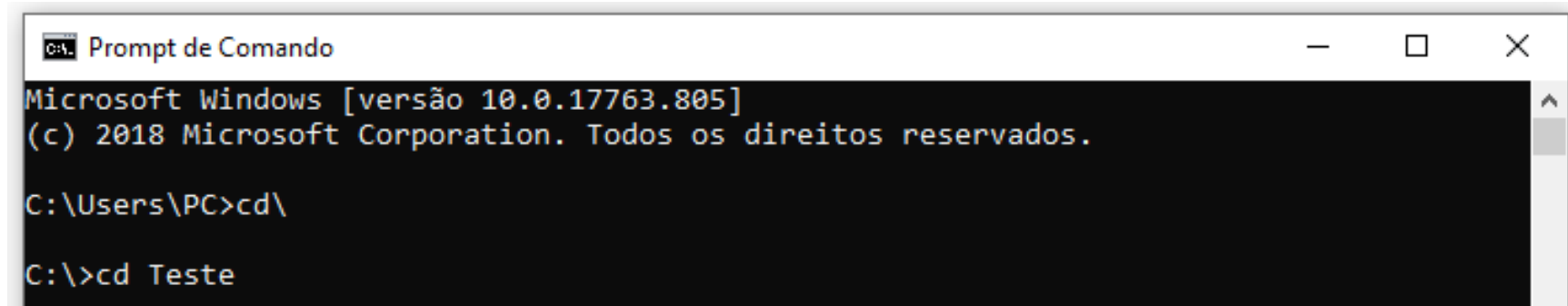
}
```

Como criar um novo repositório Git

- 1) Recomendável: ative a visualização de arquivos ocultos e extensões
- 2) Inclua um arquivo .gitignore na pasta inicial do repositório, contendo as especificações apropriadas para seu tipo de projeto
- 3) No git bash execute o comando:

git init

Controlando as versões



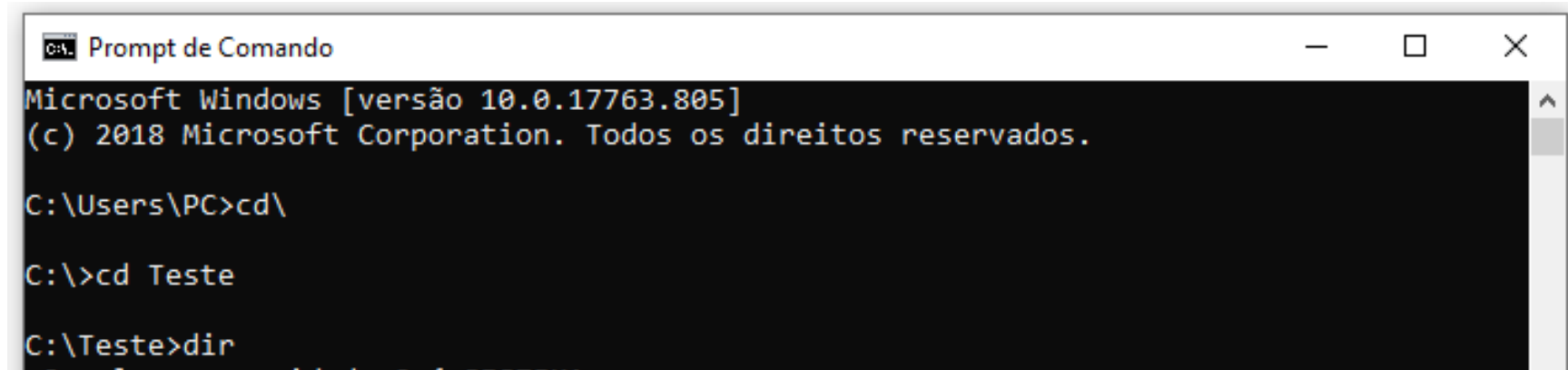
```
C:\> Prompt de Comando

Microsoft Windows [versão 10.0.17763.805]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\PC>cd\

C:\>cd Teste
```

Controlando as versões



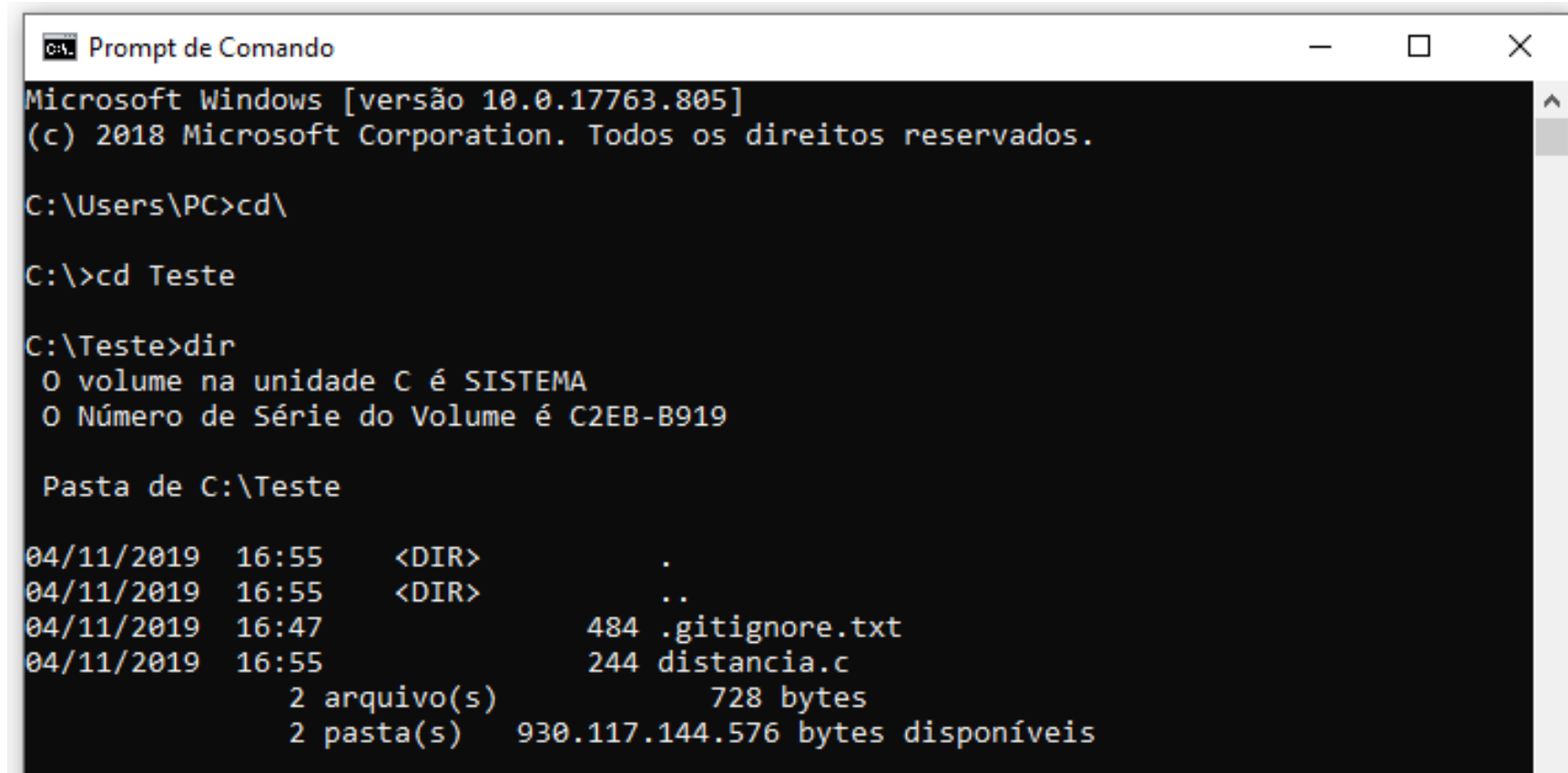
```
C:\> Prompt de Comando
Microsoft Windows [versão 10.0.17763.805]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\PC>cd\

C:\>cd Teste

C:\Teste>dir
```

Controlando as versões



```
C:\> Prompt de Comando

Microsoft Windows [versão 10.0.17763.805]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\PC>cd\

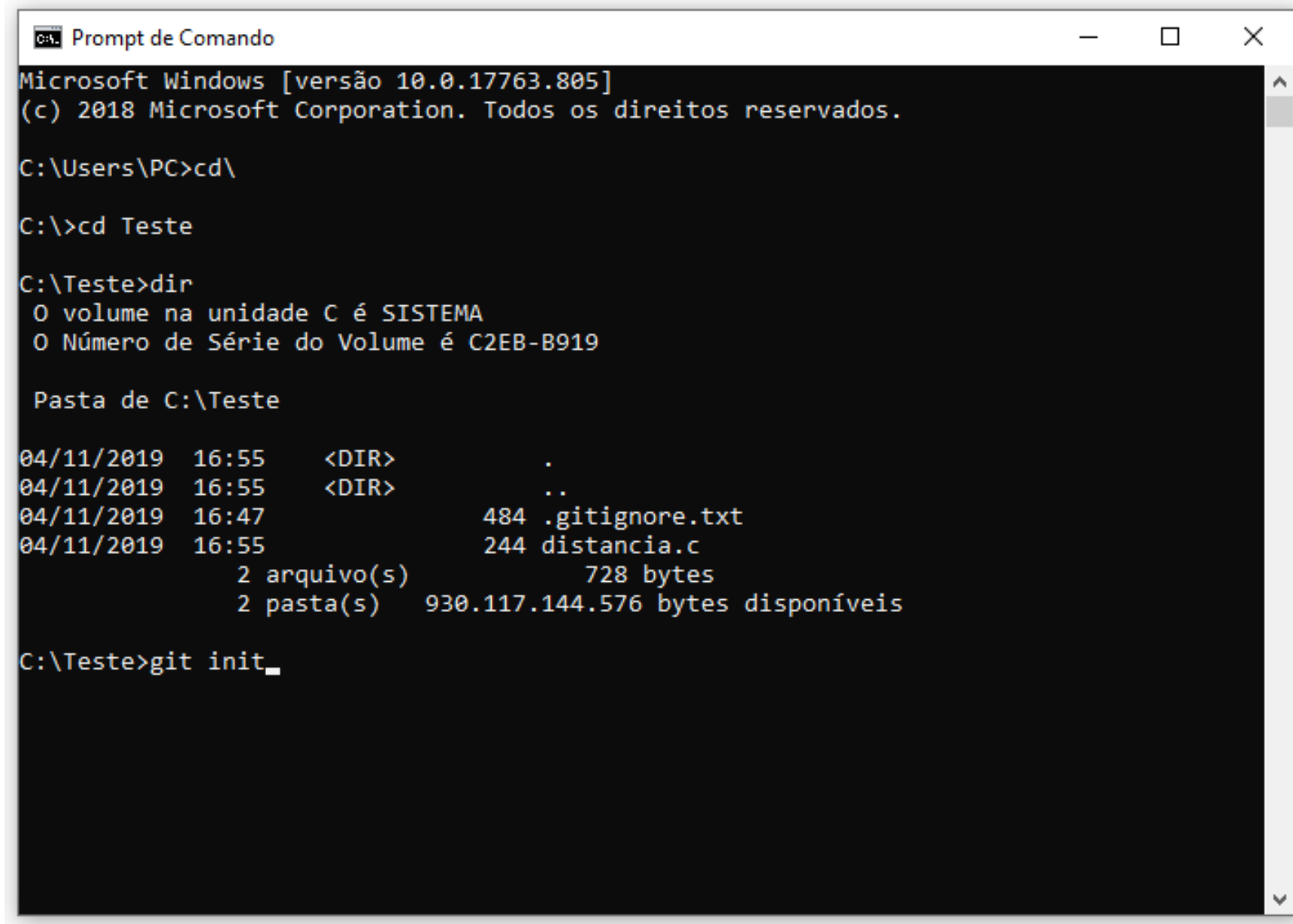
C:\>cd Teste

C:\Teste>dir
O volume na unidade C é SISTEMA
O Número de Série do Volume é C2EB-B919

Pasta de C:\Teste

04/11/2019  16:55    <DIR>          .
04/11/2019  16:55    <DIR>          ..
04/11/2019  16:47             484 .gitignore.txt
04/11/2019  16:55             244 distancia.c
                2 arquivo(s)             728 bytes
                2 pasta(s)  930.117.144.576 bytes disponíveis
```

Controlando as versões

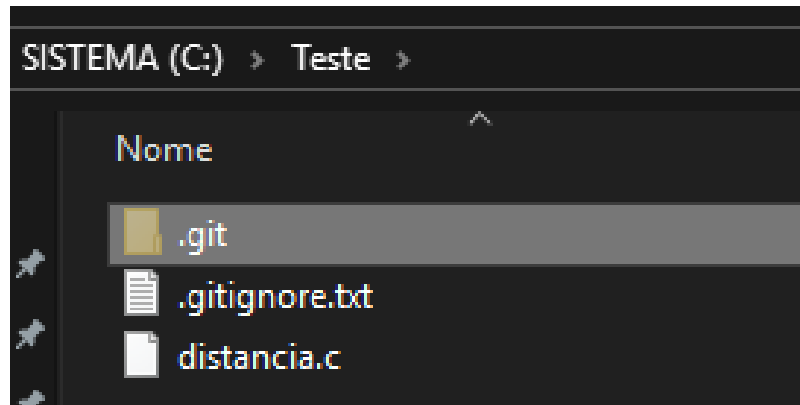
A screenshot of a Windows Command Prompt window titled "Prompt de Comando". The window has a standard Windows title bar with minimize, maximize, and close buttons. The command history is as follows:
1. Microsoft Windows [versão 10.0.17763.805]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.
2. C:\Users\PC>cd\
3. C:\>cd Teste
4. C:\Teste>dir
The output of the 'dir' command is:
O volume na unidade C é SISTEMA
O Número de Série do Volume é C2EB-B919

Pasta de C:\Teste

04/11/2019 16:55 <DIR> .
04/11/2019 16:55 <DIR> ..
04/11/2019 16:47 484 .gitignore.txt
04/11/2019 16:55 244 distancia.c
2 arquivo(s) 728 bytes
2 pasta(s) 930.117.144.576 bytes disponíveis
5. C:\Teste>git init_

Controlando as versões

```
C:\Teste>git init  
Initialized empty Git repository in C:/Teste/.git/  
  
C:\Teste>_
```



Configurando a identificação do usuário no Git

- `git config --global user.name "Seu Nome"`
- `git config --global user.email "Seu Email"`

 Node.js command prompt

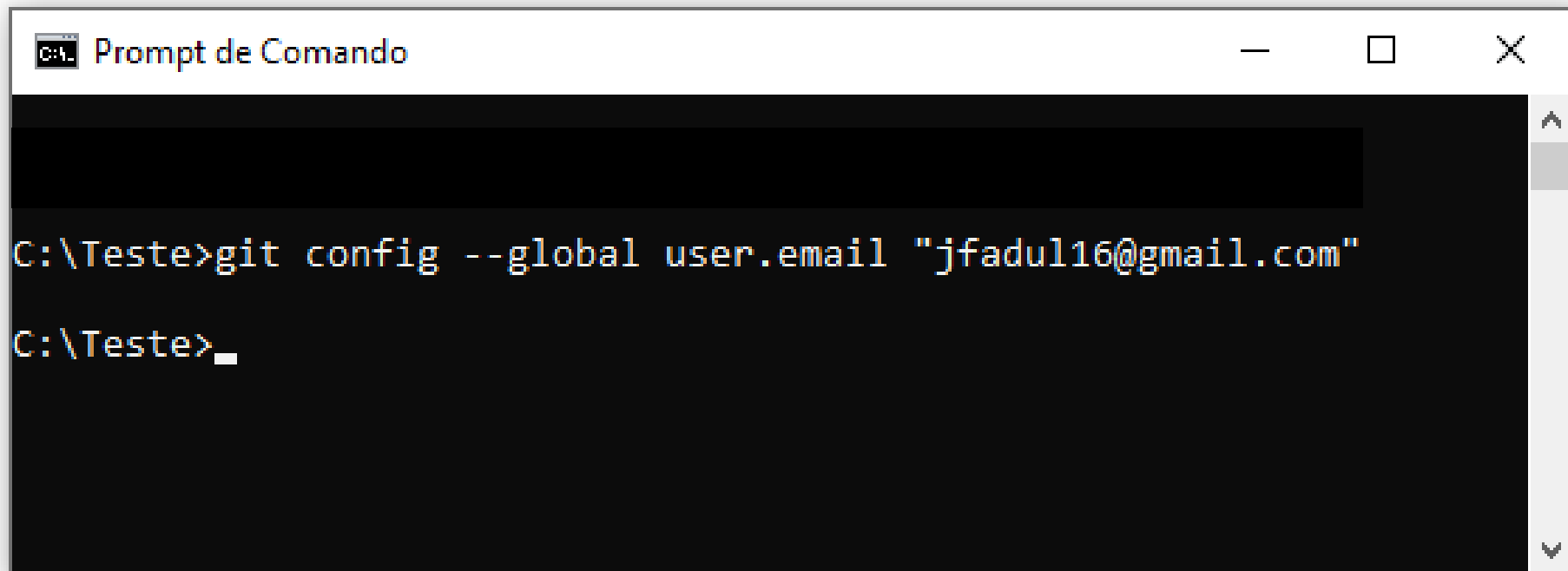
```
C:\Teste>git config --global user.name "jfadul"
```

```
C:\Teste>
```

ATENÇÃO: é importante você colocar o mesmo email que você vai usar para se cadastrar no Github (veremos mais tarde porquê)

Configurando a identificação do usuário no Git

- `git config --global user.name "Seu Nome"`
- `git config --global user.email "Seu Email"`



```
C:\Teste>git config --global user.email "jfadul16@gmail.com"
C:\Teste>
```

ATENÇÃO: é importante você colocar o mesmo email que você vai usar para se cadastrar no Github (veremos mais tarde porquê)

Como salvar uma "versão" do seu projeto

A ação de salvar uma "versão" do projeto chama-se "commit"

Como salvar uma "versão" do seu projeto

A ação de salvar uma "versão" do projeto chama-se "commit"

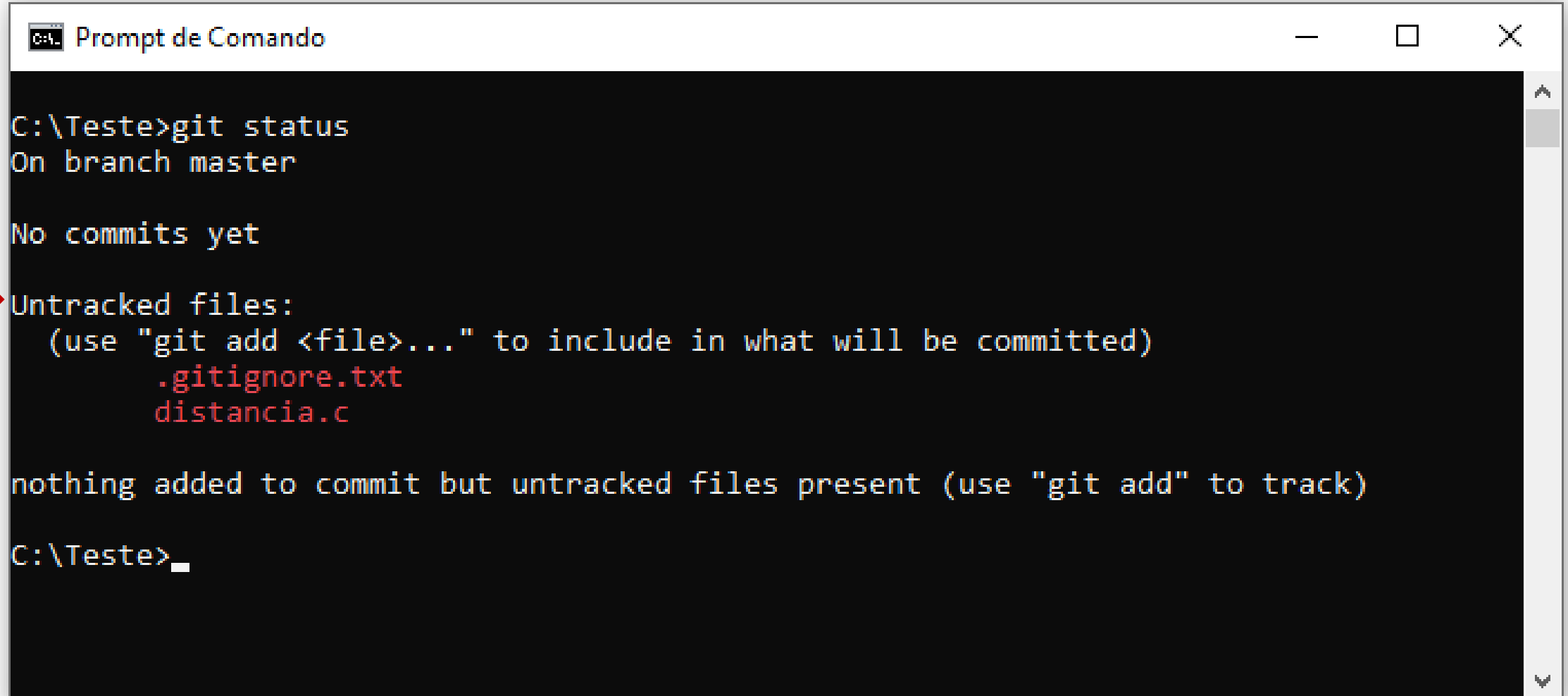
Primeiro vamos verificar como está nosso repositório:

```
git status
```

vermelho: arquivos não rastreados (untracked)

verde: arquivo prontos para serem "commitados" (stage)

Como salvar uma "versão" do seu projeto



```
Prompt de Comando

C:\Teste>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore.txt
        distancia.c

nothing added to commit but untracked files present (use "git add" to track)

C:\Teste>_
```

Como salvar uma "versão" do seu projeto

Feito em duas etapas:

Como salvar uma "versão" do seu projeto

Feito em duas etapas:

1) Coloque os arquivos modificados na área de "stage":

```
git add .
```



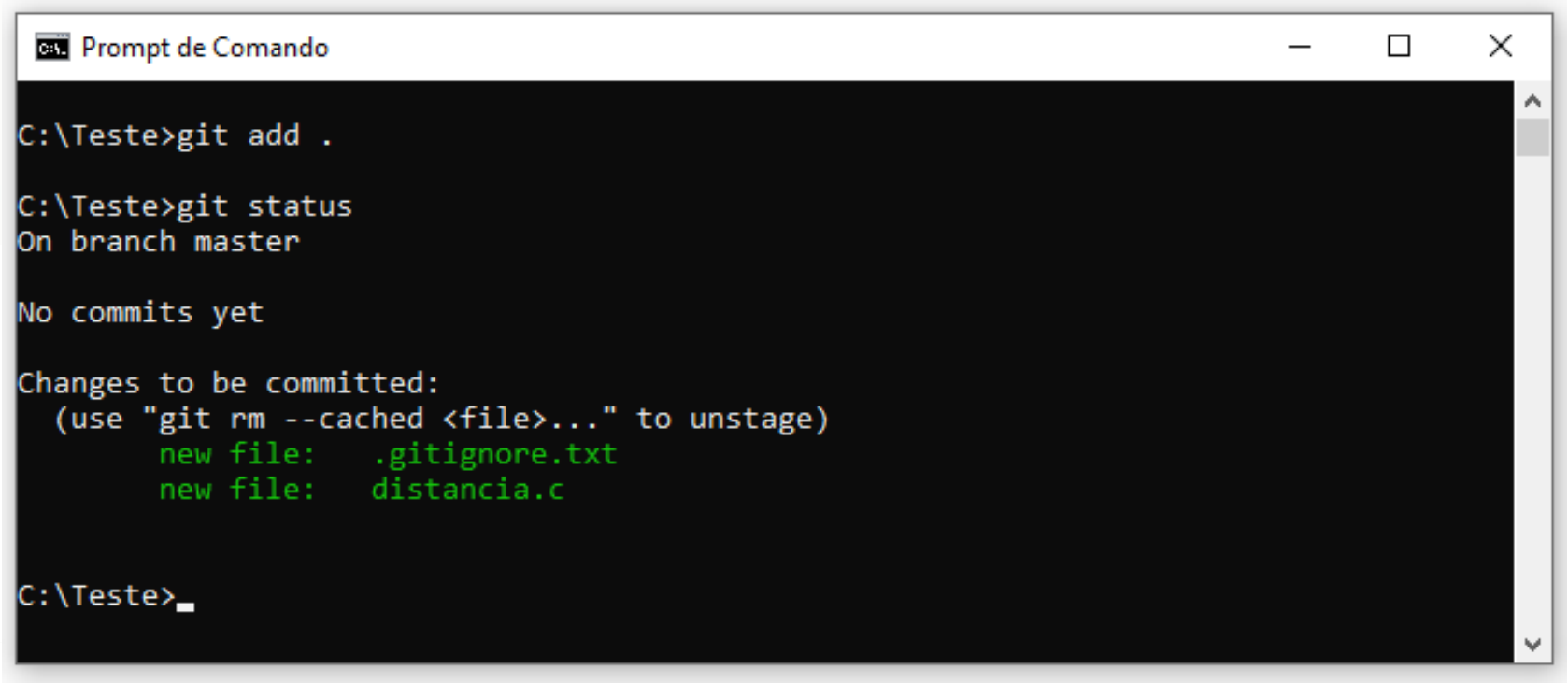
Todos os arquivos.

Como salvar uma "versão" do seu projeto

Feito em duas etapas:

1) Coloque os arquivos modificados na área de "stage":

git add .



```
C:\Teste>git add .

C:\Teste>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   .gitignore.txt
        new file:   distancia.c

C:\Teste>_
```

Como salvar uma "versão" do seu projeto

Feito em duas etapas:

1) Coloque os arquivos modificados na área de "stage":

```
git add .
```

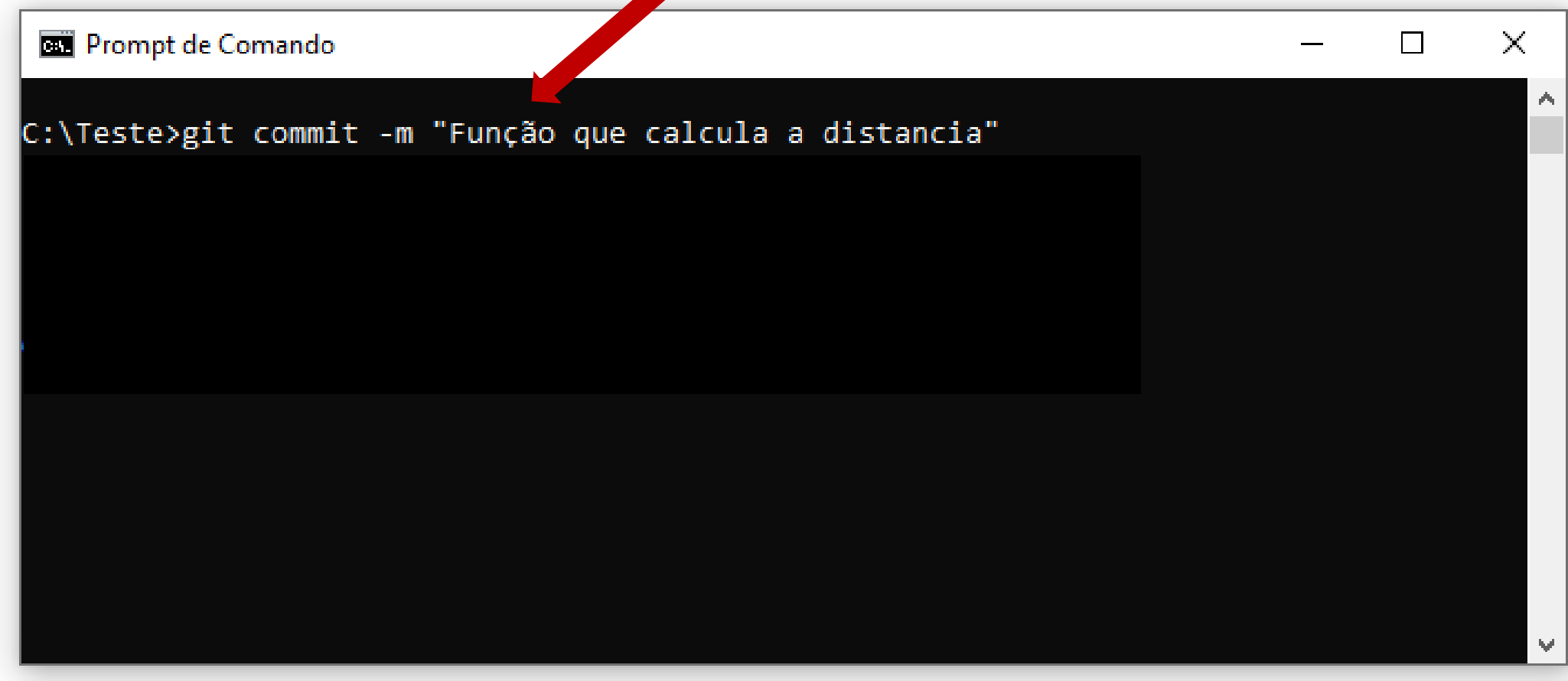
2) Realize o commit:

```
git commit -m "Descrição do commit"
```

2) Realize o commit:

```
git commit -m "Descrição do commit"
```

Sem acentuação

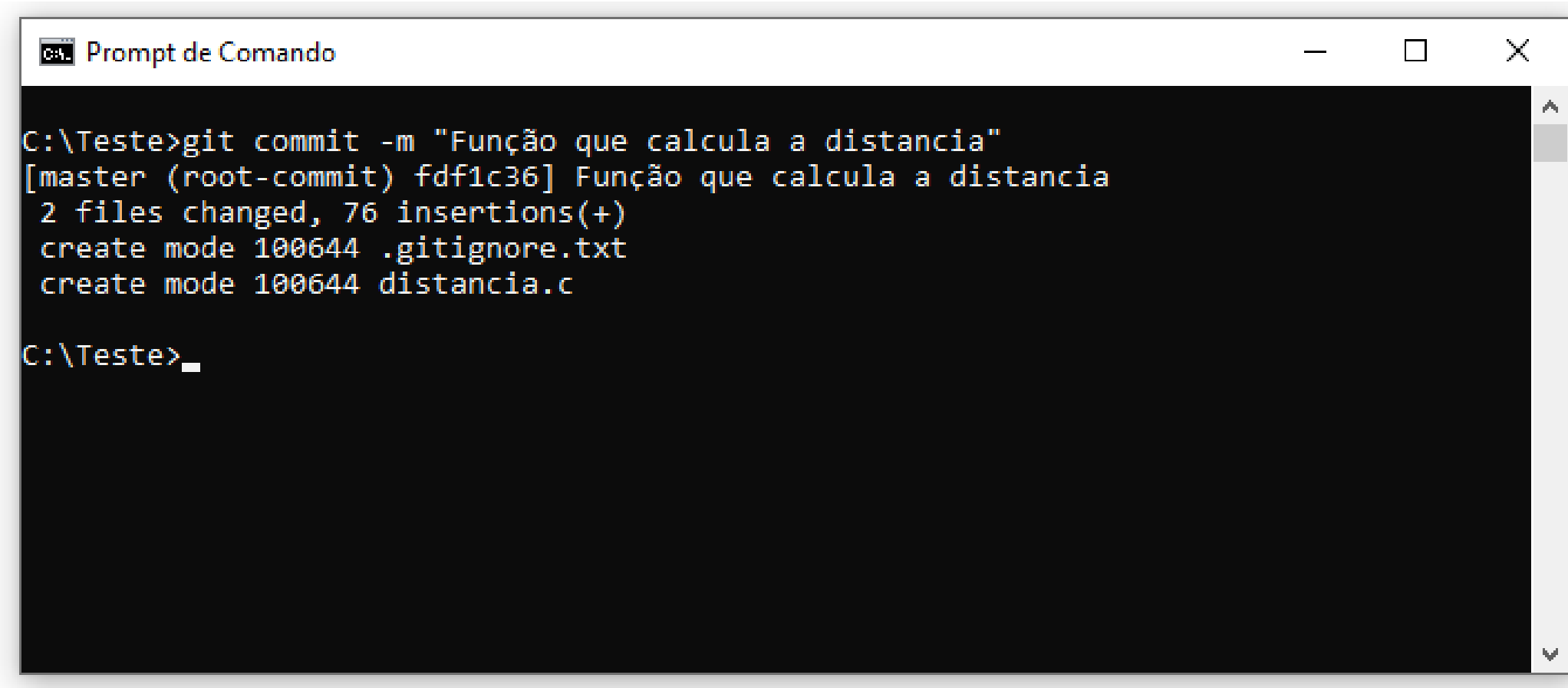


The screenshot shows a Windows Command Prompt window titled "Prompt de Comando". The command entered is `C:\Teste>git commit -m "Função que calcula a distancia"`. A red arrow points from the text "Sem acentuação" to the command, highlighting the correction of the accent in the commit message.

```
C:\Teste>git commit -m "Função que calcula a distancia"
```

2) Realize o commit:

```
git commit -m "Descrição do commit"
```



The image shows a Windows Command Prompt window titled "Prompt de Comando". The window has standard Windows window controls (minimize, maximize, close) in the top right corner. The command prompt shows the following text:

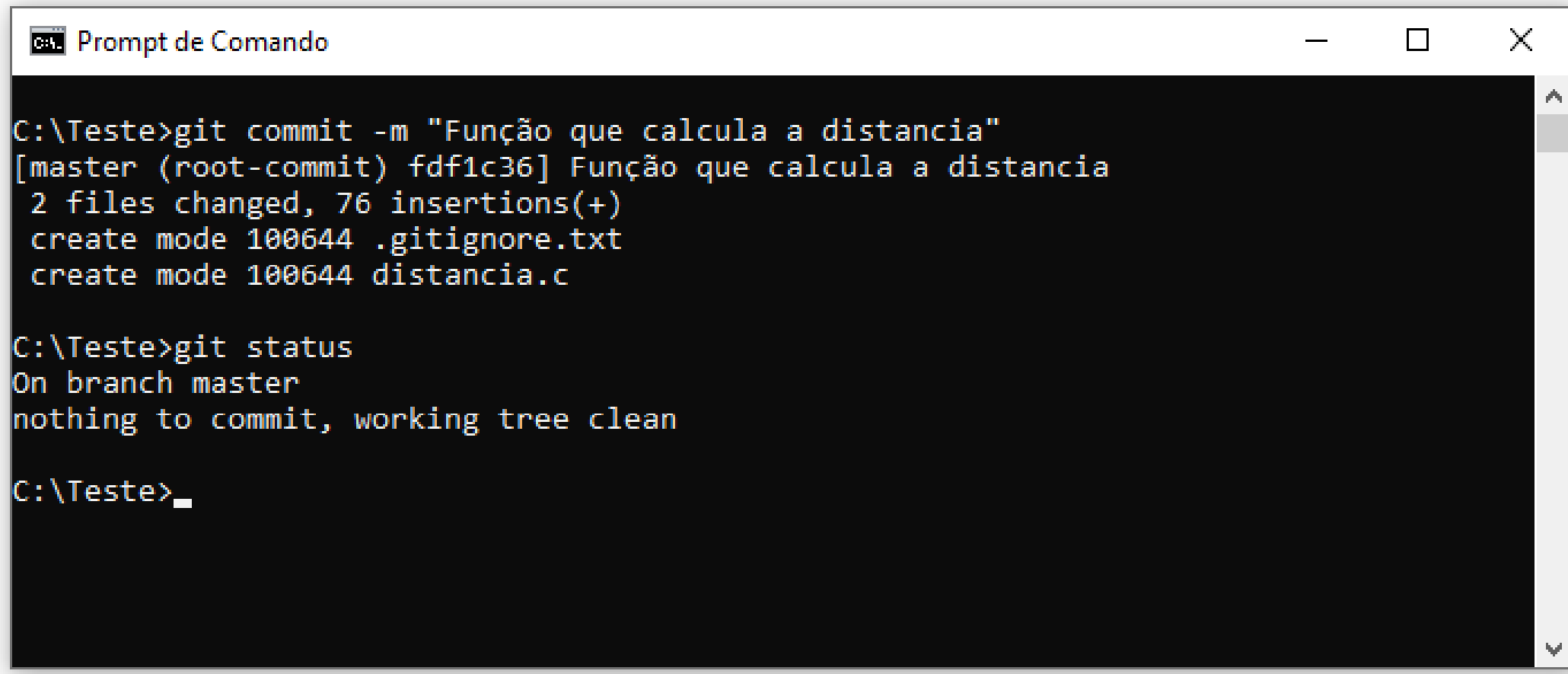
```
C:\Teste>git commit -m "Função que calcula a distancia"
[master (root-commit) fdf1c36] Função que calcula a distancia
 2 files changed, 76 insertions(+)
 create mode 100644 .gitignore.txt
 create mode 100644 distancia.c

C:\Teste>_
```

The output of the command shows the commit was successful, creating a new root commit on the master branch. It lists the files changed and the number of insertions. The prompt ends with an underscore character.

2) Realize o commit:

```
git commit -m "Descrição do commit"
```



```
cmd Prompt de Comando

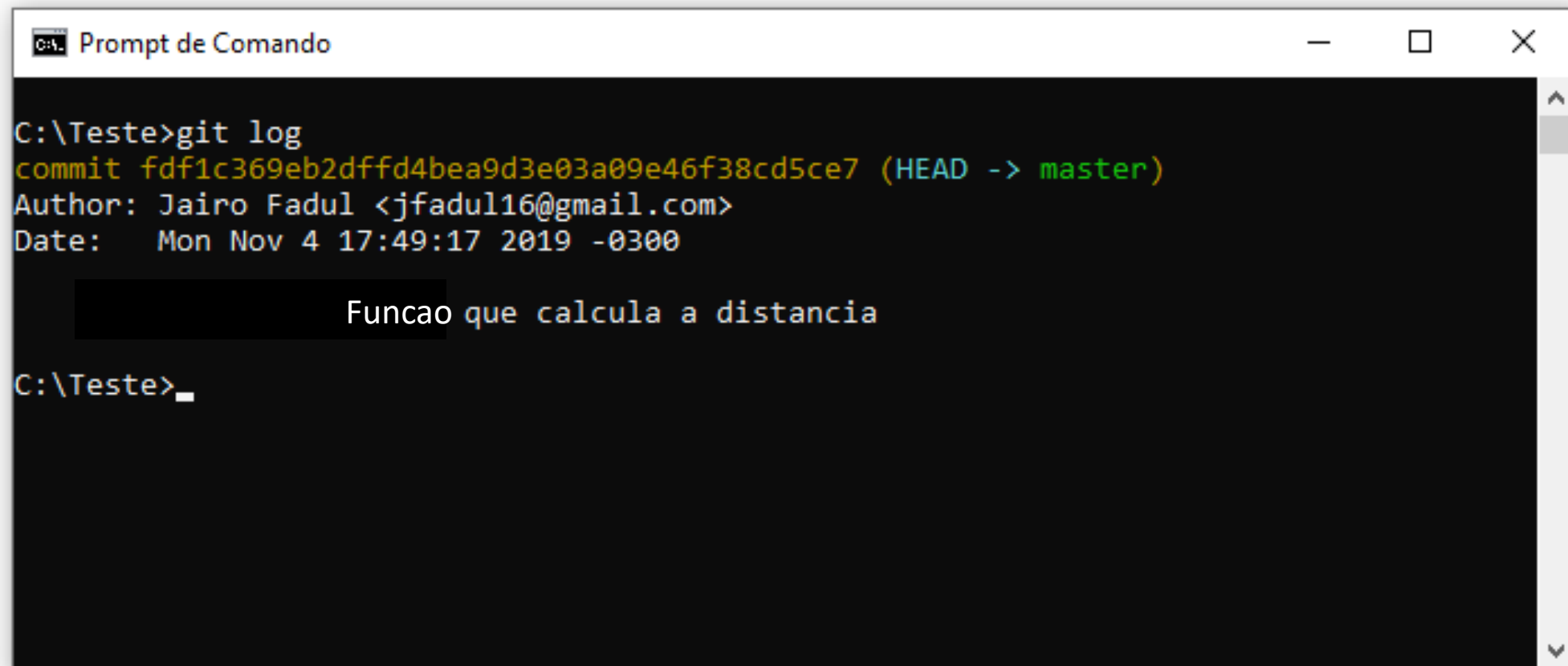
C:\Teste>git commit -m "Função que calcula a distancia"
[master (root-commit) fdf1c36] Função que calcula a distancia
 2 files changed, 76 insertions(+)
 create mode 100644 .gitignore.txt
 create mode 100644 distancia.c

C:\Teste>git status
On branch master
nothing to commit, working tree clean

C:\Teste>_
```

Para verificar o que foi feito

`git log`



```
C:\Teste>git log
commit fdf1c369eb2dffd4bea9d3e03a09e46f38cd5ce7 (HEAD -> master)
Author: Jairo Fadul <jfadul16@gmail.com>
Date:   Mon Nov 4 17:49:17 2019 -0300

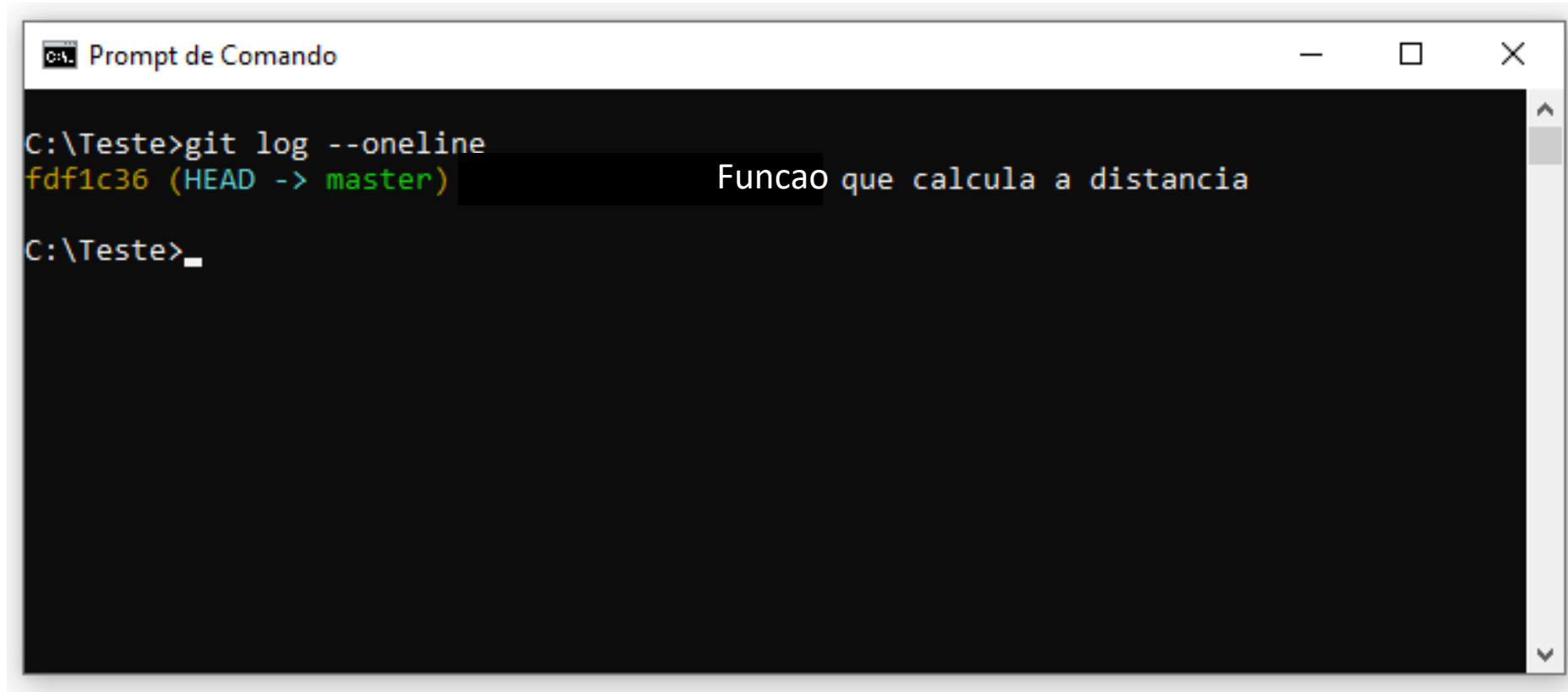
    Funcao que calcula a distancia

C:\Teste>
```

Para verificar o que foi feito

```
git log
```

```
git log --oneline
```



```
Prompt de Comando

C:\Teste>git log --oneline
fdf1c36 (HEAD -> master) Funcao que calcula a distancia

C:\Teste>_
```

Nova
alteração

distancia.c

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

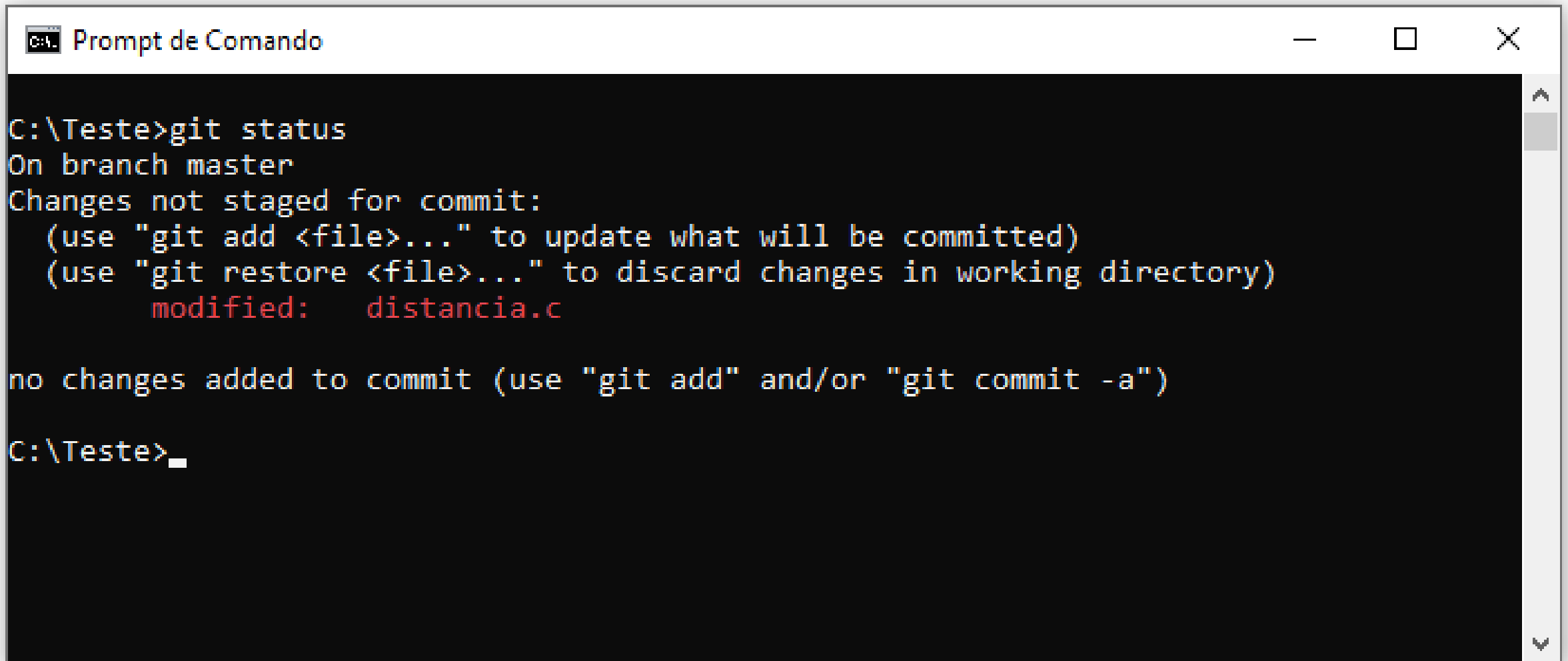
// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {

    // definição das variáveis
    int x1,y1,x2,y2;
    float distancia;

}
```


Controlando a nova alteração



```
C:\Teste>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   distancia.c

no changes added to commit (use "git add" and/or "git commit -a")

C:\Teste>_
```

Controlando a nova alteração

cmd Prompt de Comando

```
C:\Teste>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   distancia.c

no changes added to commit (use "git add" and/or "git commit -a")
```

Controlando a nova alteração

Prompt de Comando

```
C:\Teste>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   distancia.c

no changes added to commit (use "git add" and/or "git commit -a")

C:\Teste>git add .
```

Controlando a nova alteração

C:\> Prompt de Comando

```
C:\Teste>git commit -m "Definicao das variaveis"  
[master 2f7c7f0] Definicao das variaveis  
1 file changed, 3 insertions(+), 1 deletion(-)
```

Controlando a nova alteração

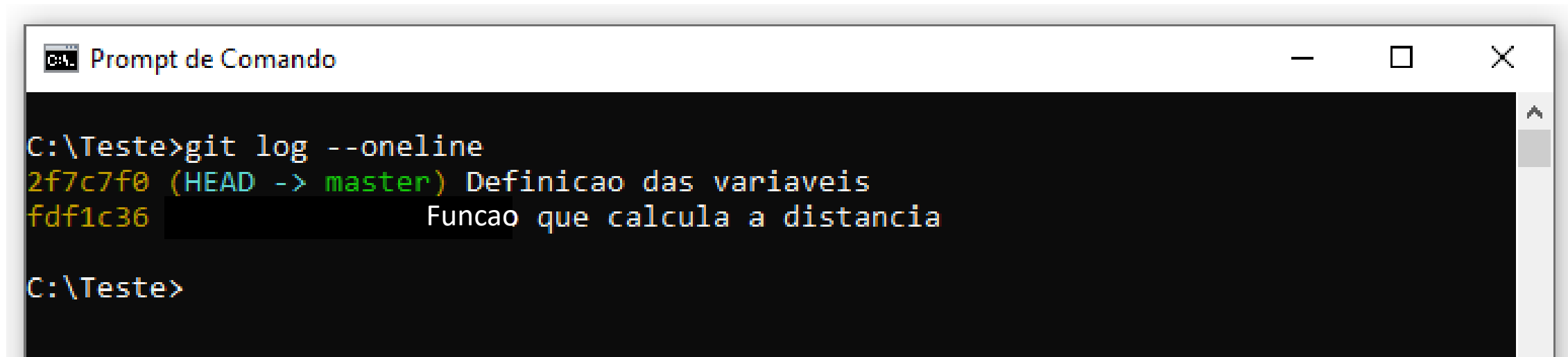
C:\> Prompt de Comando

```
C:\Teste>git commit -m "Definicao das variaveis"
[master 2f7c7f0] Definicao das variaveis
 1 file changed, 3 insertions(+), 1 deletion(-)

C:\Teste>git status
On branch master
nothing to commit, working tree clean

C:\Teste>_
```

Controlando a nova alteração



```
C:\Teste>git log --oneline
2f7c7f0 (HEAD -> master) Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>
```

The image shows a Windows Command Prompt window titled "Prompt de Comando". The window has a standard title bar with minimize, maximize, and close buttons. The command prompt shows the execution of the command `git log --oneline` in the directory `C:\Teste`. The output displays two commit entries: `2f7c7f0 (HEAD -> master) Definicao das variaveis` and `fdf1c36 Funcao que calcula a distancia`. The commit hashes are shown in yellow, the branch name in green, and the commit messages in white. The prompt returns to `C:\Teste>` after the command is executed.

Salvando versões - **Resumo**

Salvando versões - **Resumo**

- `git status`

Salvando versões - **Resumo**

- **untracked**
(não rastreados)

Salvando versões - **Resumo**

git add .

- **untracked** ->
(não rastreados)

Salvando versões - **Resumo**

git add .

- **untracked** -> **staged**
(não rastreados) (pronto)

Salvando versões - **Resumo**

git add .

git commit -m "msg"

- **untracked** -> **staged**
(não rastreados) (pronto)

Salvando versões - **Resumo**



Salvando versões - **Resumo**

git add .

git commit -m "msg"

- **untracked** (não rastreados) \rightarrow **staged** (pronto) \rightarrow **committed** (salvo)
- `git status`

Salvando versões - **Resumo**

- | | | | | |
|--------------------|------------------------|---------------|----------------------------------|------------------|
| | <code>git add .</code> | | <code>git commit -m "msg"</code> | |
| • untracked | -> | staged | -> | committed |
| (não rastreados) | | (pronto) | | (salvo) |
-
- | | |
|----------------------------------|--|
| • <code>git status</code> | <code>git status</code> |
| • <code>git log</code> | <code>git add .</code> |
| • <code>git log --oneline</code> | <code>git commit -m "leitura das coordenadas"</code> |
| | <code>git log --oneline</code> |

Controlando as versões

distancia.c

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {

    // definição das variáveis
    int x1,y1,x2,y2;
    float distancia;

    // leitura das coordenadas dos pontos
    printf("Informe as coordenadas do ponto A ");
    scanf("%i %i", &x1, &y1);
    printf("Informe as coordenadas do ponto B ");
    scanf("%i %i", &x2, &y2);

}
```


Salvando versões


```
git status
```

```
git add .
```

```
git commit -m "leitura das coordenadas"
```

```
git log --oneline
```

Salvando versões

 Node.js command prompt

```
C:\Teste>git log --oneline
ad31e7d (HEAD -> master) leitura das coordenadas
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>_
```

E se eu quiser desfazer o último commit?

Desejo
alterar a
mensagem

distancia.c

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {

    // definição das variáveis
    int x1,y1,x2,y2;
    float distancia;

    // leitura das coordenadas dos pontos
    printf("Informe as coordenadas para o primeiro ponto: ");
    scanf("%i %i", &x1, &y1);
    printf("Informe as coordenadas para o segundo ponto: ");
    scanf("%i %i", &x2, &y2);

}
```

E se eu quiser desfazer o último commit?

Remover o último commit INCLUSIVE as alterações nos arquivos (**AÇÃO PERIGOSA**):

```
git reset --hard HEAD~1
```

E se eu quiser desfazer o último commit?

```
C:\Teste>git reset --hard HEAD~1  
HEAD is now at 2f7c7f0 Definicao das variaveis  
C:\Teste>_
```

E se eu quiser desfazer o último commit?

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


// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {
    // definicao das variaveis
    int x1,y1,x2,y2;
    float distancia;

}
```

A alteração desaparece do editor,
automaticamente.

E se eu quiseser desfazer o último commit?

 Node.js command prompt

```
C:\Teste>git status
On branch master
nothing to commit, working tree clean

C:\Teste>git log --oneline
2f7c7f0 (HEAD -> master) Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>_
```


Alterar a
mensagem

distancia.c

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {

    // definição das variáveis
    int x1,y1,x2,y2;
    float distancia;

    // leitura das coordenadas dos pontos
    printf("Informe as coordenadas para o primeiro ponto: ");
    scanf("%i %i", &x1, &y1);
    printf("Informe as coordenadas para o segundo ponto: ");
    scanf("%i %i", &x2, &y2);

}
```

Salvando versões

```
git status
```

```
git add .
```

```
git commit -m "leitura das coordenadas"
```

```
git log --oneline
```

Completar o programa

distancia.c

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {

    // definição das variáveis
    int x1,y1,x2,y2;
    float distancia;

    // leitura das coordenadas dos pontos
    printf("Informe as coordenadas para o primeiro ponto: ");
    scanf("%i %i", &x1, &y1);
    printf("Informe as coordenadas para o segundo ponto: ");
    scanf("%i %i", &x2, &y2);

    // calculo da distancia entre os pontos
    distancia = calculaDistancia(x1,y1,x2,y2);

    // mostra o resultado
    printf("Distancia: %f", distancia);
}
```

Salvando versões

```
git status
```

```
git add .
```

```
git commit -m "calcula a distancia e mostra o resultado"
```

```
git log --oneline
```

Salvando versões

```
Node.js command prompt
C:\Teste>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   distancia.c

no changes added to commit (use "git add" and/or "git commit -a")

C:\Teste>git add .

C:\Teste>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   distancia.c

C:\Teste>git commit -m "calcula a distancia e mostra o resultado"
[master 90b3067] calcula a distancia e mostra o resultado
 1 file changed, 11 insertions(+), 9 deletions(-)

C:\Teste>git log --oneline
90b3067 (HEAD -> master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>
```

E se eu quiser somente dar uma olhada em uma versão anterior?

```
C:\Teste>git log --oneline
90b3067 (HEAD -> master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia
C:\Teste>
```

E se eu quiser somente dar uma olhada em uma versão anterior?

```
C:\Teste>git log --oneline
90b3067 (HEAD -> master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia
C:\Teste>
```

Quero ver como
o projeto estava
neste ponto.

E se eu quiser somente dar uma olhada em uma versão anterior?

Node.js command prompt

```
C:\Teste>git status
On branch master
nothing to commit, working tree clean


C:\Teste>_
```

Para isso o projeto não deve ter nenhuma alteração pendente.

E se eu quiser somente dar uma olhada em uma versão anterior?

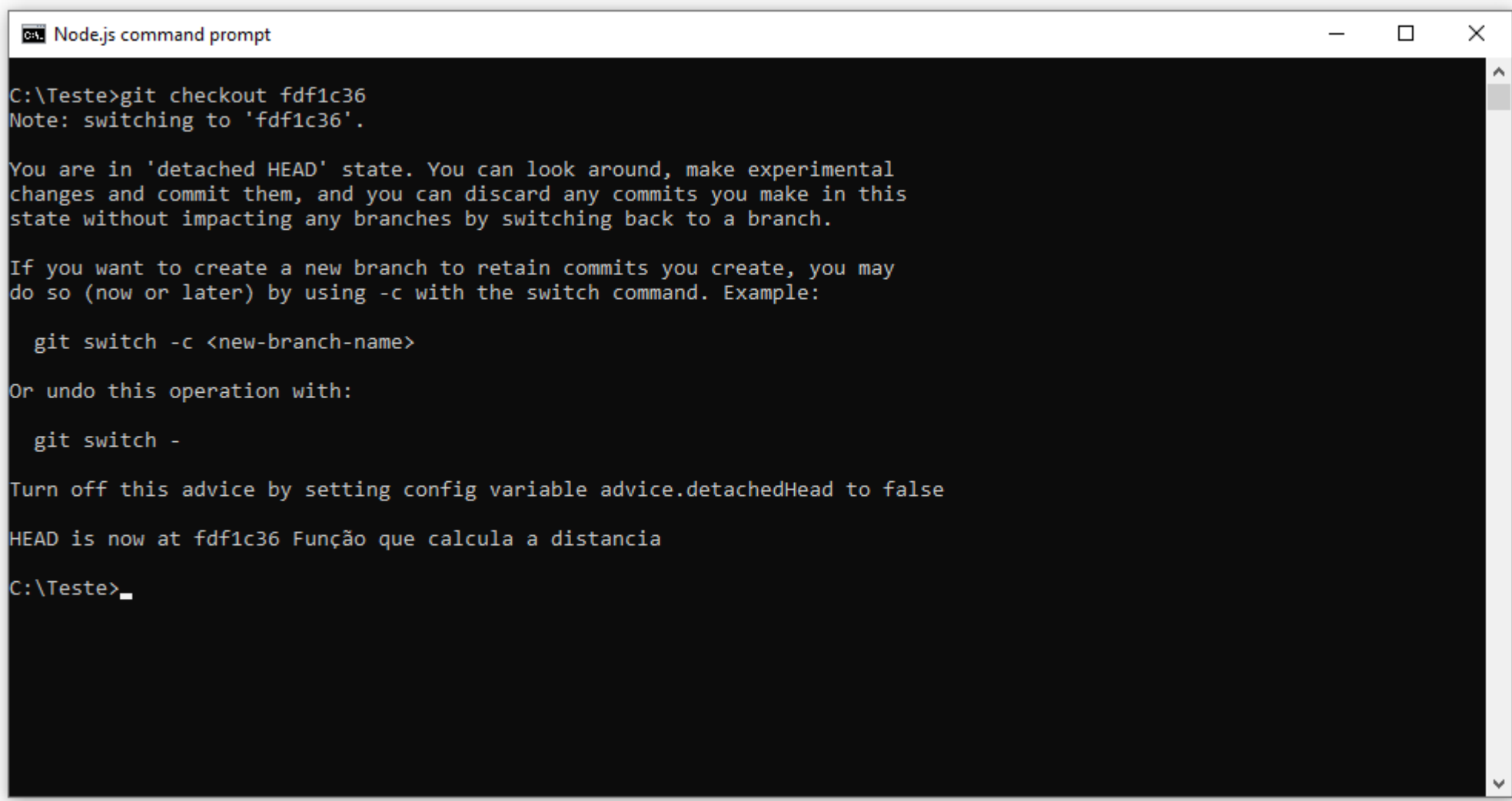
1) Navegar entre commits, alterando os arquivos temporariamente:

`git checkout <código do commit>`



```
C:\Teste>git log --oneline
90b3067 (HEAD -> master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia
C:\Teste>
```

E se eu quiser somente dar uma olhada em uma versão anterior?



```
Node.js command prompt
C:\Teste>git checkout fdf1c36
Note: switching to 'fdf1c36'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at fdf1c36 Função que calcula a distancia
C:\Teste>
```

E se eu quiser somente dar uma olhada em uma versão anterior?

Node.js command prompt

```
C:\Teste>git log --oneline
```

```
fdf1c36 (HEAD) Funcao que calcula a distancia
```

```
C:\Teste>_
```

E se eu quiser somente dar uma olhada em uma versão anterior?

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {
```

O editor é atualizado com a versão desejada, automaticamente.

```
}
```

E se eu quiser somente dar uma olhada em uma versão anterior?

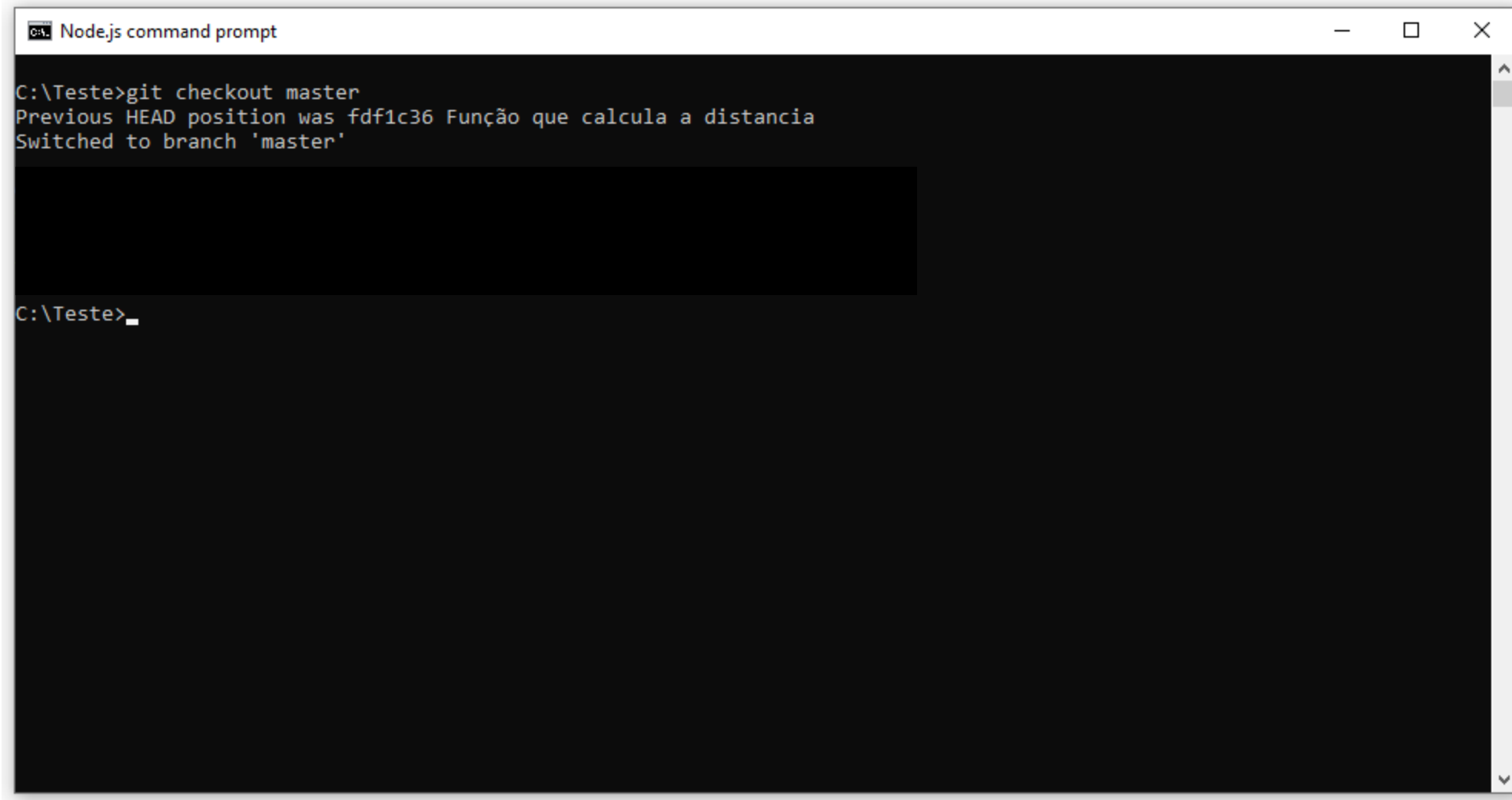
2) Voltar para o último commit:

`git checkout <nome do branch>`

```
C:\Teste>git log --oneline
90b3067 (HEAD -> master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>
```

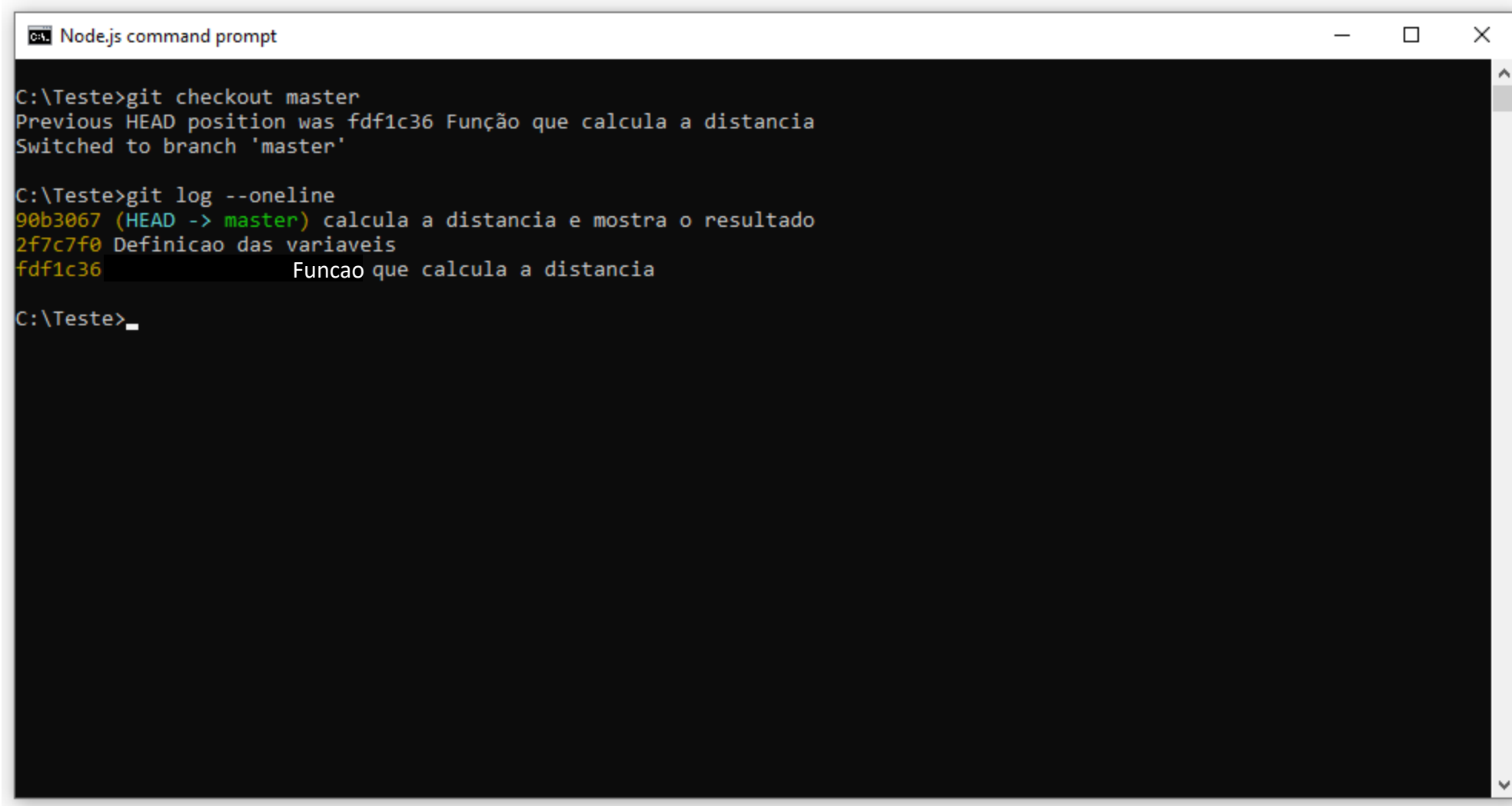
E se eu quiser somente dar uma olhada em uma versão anterior?



```
Node.js command prompt
C:\Teste>git checkout master
Previous HEAD position was fdf1c36 Função que calcula a distancia
Switched to branch 'master'

C:\Teste>_
```

E se eu quiser somente dar uma olhada em uma versão anterior?



```
Node.js command prompt

C:\Teste>git checkout master
Previous HEAD position was fdf1c36 Função que calcula a distancia
Switched to branch 'master'

C:\Teste>git log --oneline
90b3067 (HEAD -> master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>_
```

Github

Visão geral do Github

- É um serviço de armazenamento remoto de repositórios Git
- Interface com usuário via web
- Padrão da indústria para armazenamento de projetos de código aberto
- Maior hospedeiro de código fonte do mundo
- Planos pagos para repositórios privados
- É uma "rede social" de repositórios Git. Dica: currículo!



Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 40 million developers.

Username

Email

Password

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.



Sign in to GitHub


Username or email address

Password


[Forgot password?](#)


Sign in

New to GitHub? [Create an account.](#)

Repositories 

Find a repository...

 [jfadul/exercicios-react](#)

 [jfadul/curso-git](#)


Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you [watch](#) and people you [follow](#).

Explore GitHub

💡 ProTip! The feed shows you events from people you [follow](#) and repositories you [watch](#).



🔔 Subscribe to your news feed

 © 2019 GitHub, Inc.



[Blog](#)

[API](#)



[Terms](#)

 **GitHub Sponsors is out of beta** 

We're now generally available in 30 countries, with more coming soon.

 **Securing software together** 

Introducing new ways to identify and prevent security vulnerabilities across your code base.

 **Welcome to the new dashboard. Get closer to the stuff you care about most.** 

Explore repositories

logaretm/vee-validate
✔ Template Driven Validation Framework for Vue.js
JavaScript ★ 7k

rust-lang/rust
Empowering everyone to build reliable and efficient software.
Rust ★ 40.2k

libgit2/libgit2
A cross-platform, linkable library implementation of Git that you can use in your application.
C ★ 7.2k

Explore more →



😊 Set status

jfadul

Edit profile

✉ jfadul16@gmail.com

Overview

Repositories 2

Projects 0

Packages 0

Stars 0

Followers 0

Following 0

Popular repositories

Customize your pins

[curso-git](#)

Teste de comandos Git/GitHub

● HTML

[exercicios-react](#)

Projeto de teste de aplicações React/Redux.

● HTML

10 contributions in the last year

Contribution settings ▾



Contribution activity

November 2019

2019

2018

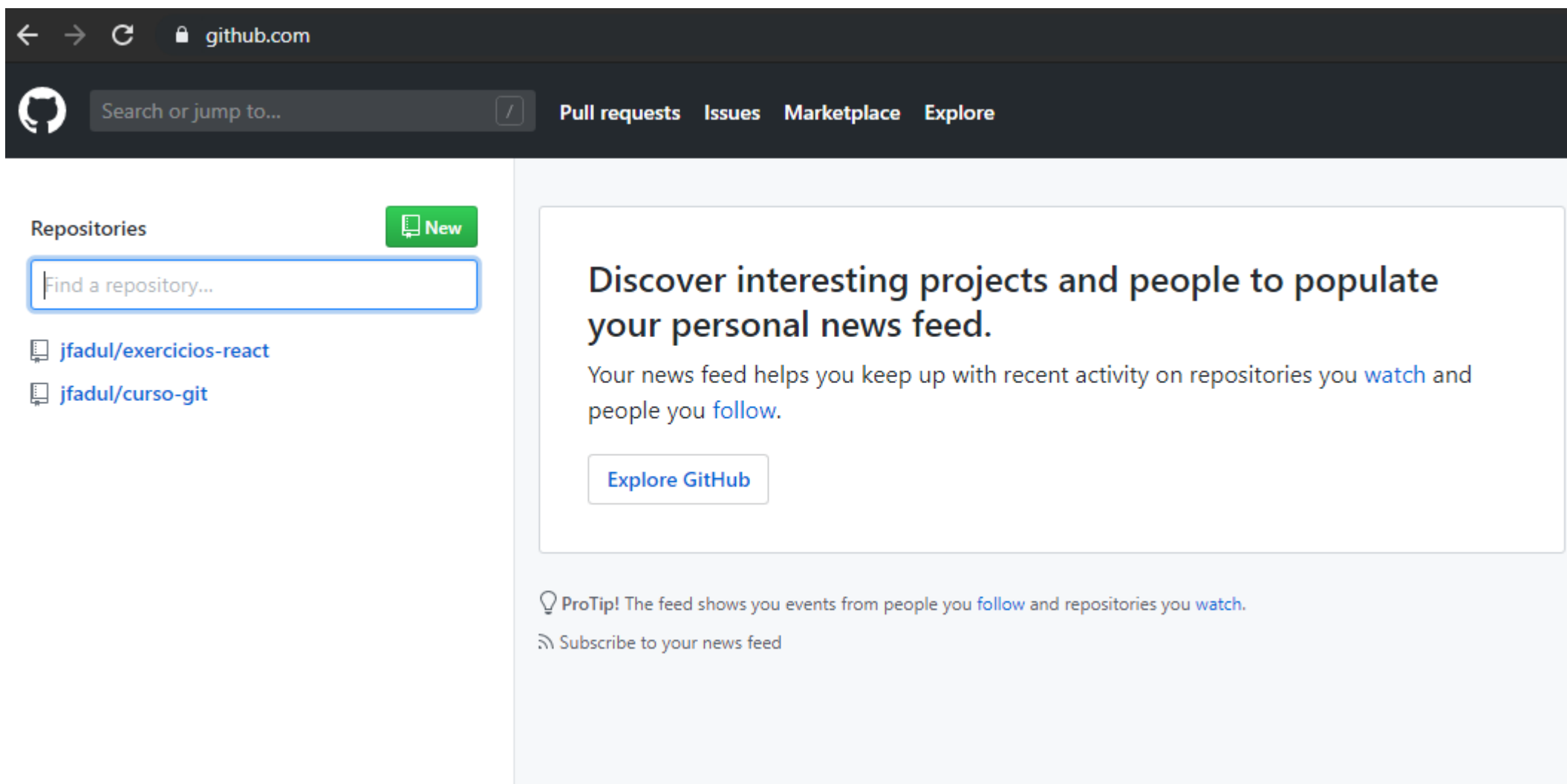
Criando um repositório remoto no Github

- Faça login no Github

Criando um repositório remoto no Github

- Faça login no Github
- Crie um repositório vazio (sem readme, sem gitignore e sem licença)

Criando um repositório remoto no Github



The image is a screenshot of the GitHub homepage. At the top is a dark navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. Below the navigation bar, the left sidebar features a 'Repositories' section with a 'New' button and a search box. Two repositories are listed: 'jfadul/exercicios-react' and 'jfadul/curso-git'. The main content area on the right has a light blue background and contains a large text block encouraging users to discover projects and people to populate their news feed. It includes a 'ProTip!' section and a 'Subscribe to your news feed' link.

github.com

Search or jump to...

Pull requests Issues Marketplace Explore

Repositories [New](#)

Find a repository...

[jfadul/exercicios-react](#)

[jfadul/curso-git](#)

Discover interesting projects and people to populate your personal news feed.

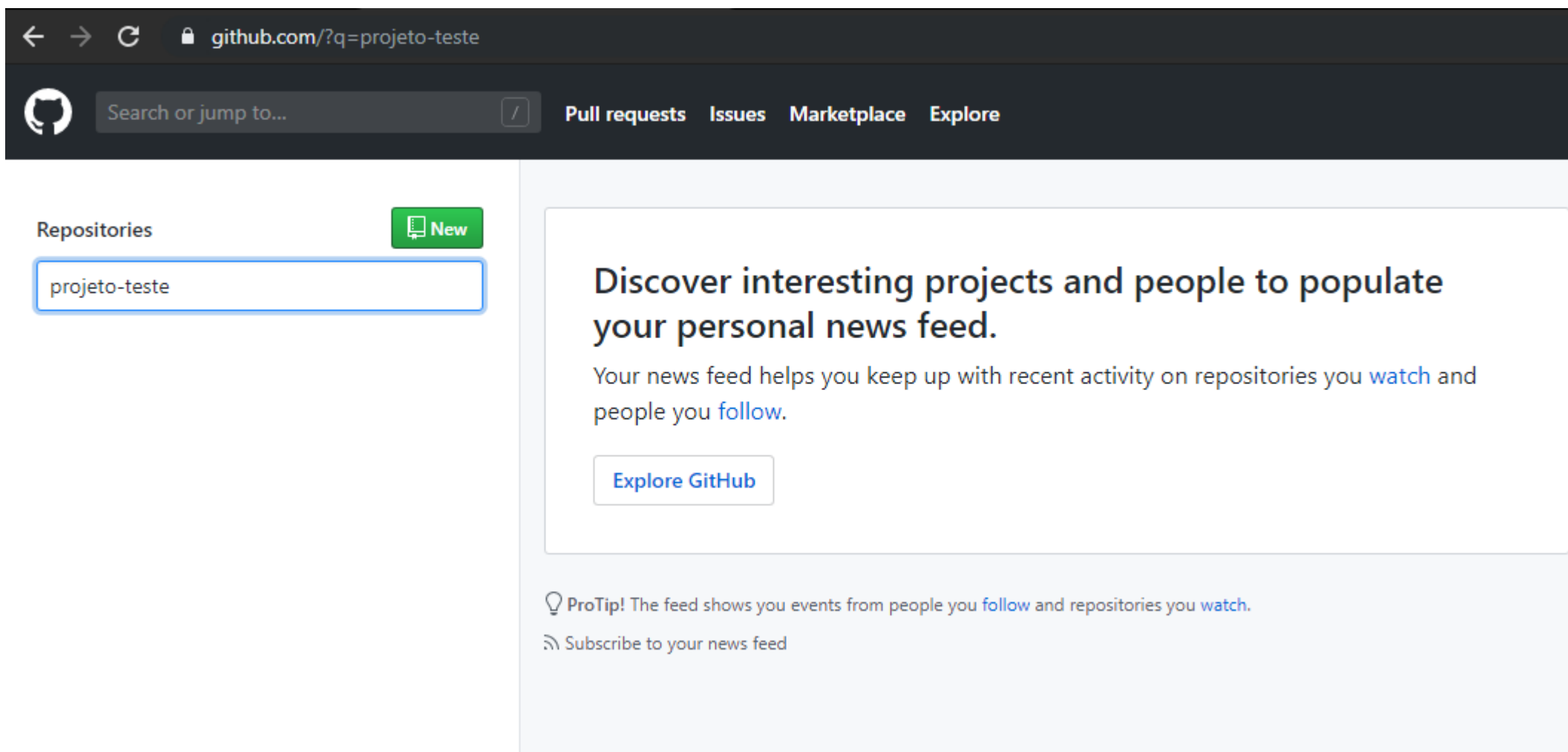
Your news feed helps you keep up with recent activity on repositories you [watch](#) and people you [follow](#).

[Explore GitHub](#)

ProTip! The feed shows you events from people you [follow](#) and repositories you [watch](#).

[Subscribe to your news feed](#)

Criando um repositório remoto no Github



The screenshot shows the GitHub homepage. The browser's address bar displays 'github.com/?q=projeto-teste'. The GitHub logo is on the left, followed by a search bar containing 'projeto-teste' and a green 'New' button. Navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore' are visible. The main content area features a large text block: 'Discover interesting projects and people to populate your personal news feed.' Below this, it says 'Your news feed helps you keep up with recent activity on repositories you watch and people you follow.' and includes an 'Explore GitHub' button. At the bottom, there is a 'ProTip!' section with a lightbulb icon and a 'Subscribe to your news feed' link.


Repositories New


projeto-teste

Discover interesting projects and people to populate your personal news feed.

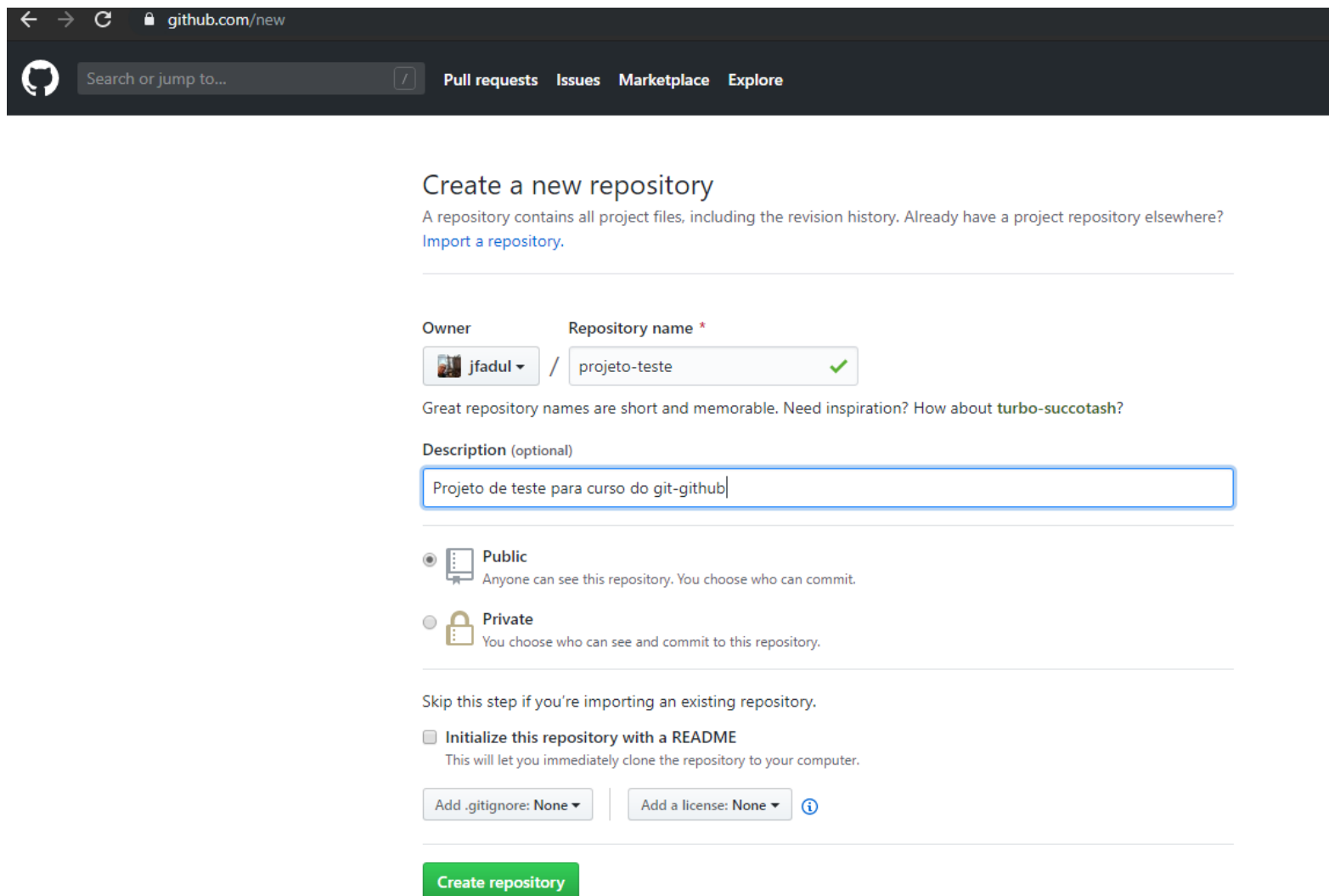
Your news feed helps you keep up with recent activity on repositories you [watch](#) and people you [follow](#).

[Explore GitHub](#)

 **ProTip!** The feed shows you events from people you [follow](#) and repositories you [watch](#).

 [Subscribe to your news feed](#)

Criando um repositório remoto no Github



The screenshot shows the GitHub 'Create a new repository' page. At the top is a dark navigation bar with the GitHub logo, a search bar, and links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The main heading is 'Create a new repository', followed by a subtext explaining what a repository is and a link to 'Import a repository'. The form fields include: 'Owner' (a dropdown menu showing 'jfadul'), 'Repository name' (a text box with 'projeto-teste' and a green checkmark), 'Description (optional)' (a text box with 'Projeto de teste para curso do git-github'), 'Public/Private' (radio buttons with 'Public' selected), 'Initialize this repository with a README' (a checkbox), 'Add .gitignore' (a dropdown menu with 'None' selected), and 'Add a license' (a dropdown menu with 'None' selected and an information icon). A green 'Create repository' button is at the bottom.

github.com/new

Search or jump to...

Pull requests Issues Marketplace Explore

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner Repository name *

jfadul / projeto-teste ✓

Great repository names are short and memorable. Need inspiration? How about **turbo-succotash**?

Description (optional)

Projeto de teste para curso do git-github

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.


Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: None | Add a license: None ⓘ

Create repository


Criando um repositório remoto no Github

 jfadul / projeto-teste

Unwatch 1Star 0Fork 0

[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Quick setup — if you've done this kind of thing before

 Set up in Desktop or [HTTPS](#) [SSH](#)

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# projeto-teste" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/jfadul/projeto-teste.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/jfadul/projeto-teste.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Associando um repositório remoto ao seu
repositório local

Associando um repositório remoto ao seu repositório local

1) Associar nosso repositório local ao repositório remoto, dando o apelido de "origin" a ele:

```
git remote add origin <URI do repositório remoto>
```

Quick setup — if you've done this kind of thing before

[📄 Set up in Desktop](#)

or

[HTTPS](#)[SSH](#)<https://github.com/jfadul/projeto-teste.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# projeto-teste" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/jfadul/projeto-teste.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/jfadul/projeto-teste.git
git push -u origin master
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

Associando um repositório remoto ao seu repositório local

1) Associar nosso repositório local ao repositório remoto, dando o apelido de "origin" a ele:

```
git remote add origin <URI do repositório remoto>
```

 Node.js command prompt

```
C:\Teste>git remote add origin https://github.com/jfadul/projeto-teste.git
```

```
C:\Teste>_
```

Associando um repositório remoto ao seu repositório local

1) Associar nosso repositório local ao repositório remoto, dando o apelido de "origin" a ele:

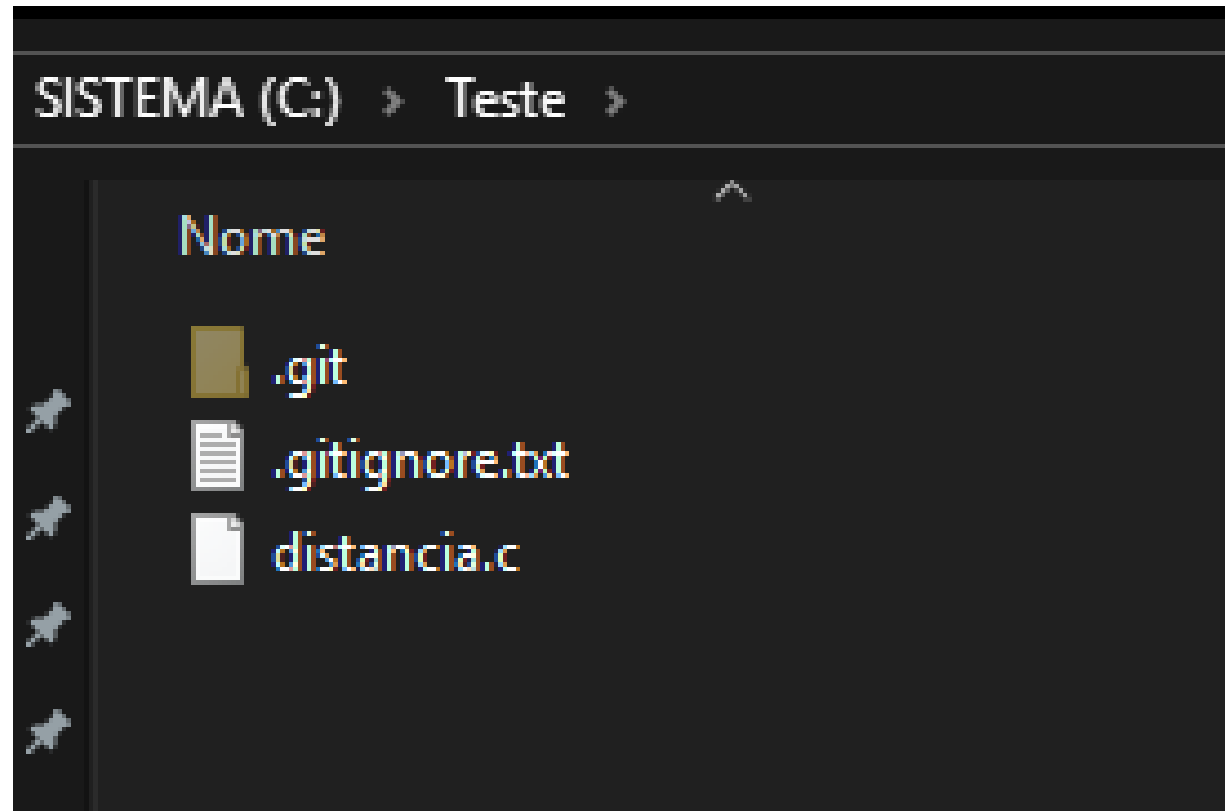
```
git remote add origin <URI do repositório remoto>
```

2) Associar o repositório local a um outro repositório remoto, porém mantendo o mesmo apelido:

```
git remote set-url origin <URI do repositório remoto>
```


Como enviar o repositório local para o Github

Como enviar o repositório local para o Github



Como enviar o repositório local para o Github

```
git push -u origin master
```

 Node.js command prompt

```
C:\Teste>git push -u origin master_
```

Como enviar o repositório local para o Github

Node.js command prompt - git push -u origin master

```
C:\Teste>git push -u origin master
Username for 'https://github.com': _
```

Node.js command prompt - git push -u origin master

```
C:\Teste>git push -u origin master
Username for 'https://github.com': jfadul
Password for 'https://jfadul@github.com': _
```

Node.js command prompt

```
C:\Teste>git push -u origin master
Username for 'https://github.com': jfadul
Password for 'https://jfadul@github.com':
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 1.39 KiB | 283.00 KiB/s, done.
Total 10 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/jfadul/projeto-teste.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

C:\Teste>_
```



Search or jump to...



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

[jfadul](#) / [projeto-teste](#)

Unwatch ▾

1

★ Star

0

Fork

0

↔ Code

! Issues 0

Pull requests 0

Projects 0

Wiki

Security

Insights

Settings

Projeto de teste para curso do git-github

Edit

[Manage topics](#)

3 commits

1 branch

0 releases

1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾



[jfadul](#) calcula a distancia e mostra o resultado

Latest commit 90b3067 12 hours ago

[.gitignore.txt](#)

Função que calcula a distancia

2 days ago

[distancia.c](#)

calcula a distancia e mostra o resultado

12 hours ago

Help people interested in this repository understand your project by adding a README.

Add a README



Branch: master ▾

projeto-teste / distancia.c

Find file

Copy path



jfadul calcula a distancia e mostra o resultado

90b3067 12 hours ago

1 contributor

28 lines (21 sloc) | 618 Bytes

Raw

Blame

History



```
1  #include <stdio.h>
2  #include <math.h>
3
4  // função que calcula a distancia
5  float calculaDistancia(int x1, int y1, int x2, int y2) {
6      return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
7  }
8  // main
9  int main() {
10     // definicao das variaveis
11     int x1,y1,x2,y2;
12     float distancia;
13
14     // define as variaveis
15     printf("Informe as coordenadas para o primeiro ponto: ");
16     scanf("%i %i", &x1, &y1);
17     printf("Informe as coordenadas para o segundo ponto: ");
18     scanf("%i %i", &x2, &y2);
19
20     // calcula a distancia
21     distancia = calculaDistancia(x1,y1,x2,y2);
22
23     // mostra o resultado
24     printf("Distancia: %f", distancia);
25
26 }
27
```

jfadul / projeto-teste

Unwatch 1

Star 0

Fork 0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Security

Insights

Settings

Branch: master

Commits on Nov 5, 2019

calcula a distancia e mostra o resultado



jfadul committed 12 hours ago



90b3067



Commits on Nov 4, 2019

Definicao das variaveis



jfadul committed 2 days ago



2f7c7f0



Função que calcula a distancia





jfadul committed 2 days ago



fdf1c36





calcula a distancia e mostra o resultado

[Browse files](#) master jfadul committed 12 hours ago1 parent [2f7c7f0](#)commit [90b3067cb68d5c1a8de2e0a3e6fa8c4574e20f3c](#) Showing **1 changed file** with 11 additions and 9 deletions.

Unified

Split

▼ 20  distancia.c 

...

@@ -11,15 +11,17 @@	int main() {
11	11
12	12
13	13
14	-
15	-
16	-
17	-
18	-
19	-
20	-
21	-
22	-
	14 + // define as variaveis
	15 + printf("Informe as coordenadas para o primeiro ponto: ");
	16 + scanf("%i %i", &x1, &y1);
	17 + printf("Informe as coordenadas para o segundo ponto: ");
	18 + scanf("%i %i", &x2, &y2);
	19 +
	20 + // calcula a distancia
	21 + distancia = calculaDistancia(x1,y1,x2,y2);
	22 +
	23 + // mostra o resultado
	24 + printf("Distancia: %f", distancia);
23	25
24	26 }
25	27


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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {
    // definicao das variaveis
    int x1,y1,x2,y2;
    float distancia;

    // define as variaveis
    printf("Informe as coordenadas para o primeiro ponto: ");
    scanf("%i %i", &x1, &y1);
    printf("Informe as coordenadas para o segundo ponto: ");
    scanf("%i %i", &x2, &y2);

    // calcula a distancia
    distancia = calculaDistancia(x1,y1,x2,y2);

    // mostra o resultado
    printf("Distancia: %f", distancia);
}
```

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

// função que calcula a distancia
float calculaDistancia(int x1, int y1, int x2, int y2) {
    return sqrt( pow(x2-x1,2) + pow(y2-y1, 2) );
}

// main
int main() {
    // definicao das variaveis
    int x1,y1,x2,y2;
    float distancia;

    // define as variaveis
    printf("Informe as coordenadas para o primeiro ponto: ");
    scanf("%i %i", &x1, &y1);
    printf("Informe as coordenadas para o segundo ponto: ");
    scanf("%i %i", &x2, &y2);

    // calcula a distancia e mostra o resultado
    distancia = calculaDistancia(x1,y1,x2,y2);
    printf("Distancia: %f", distancia);
}
```

Node.js command prompt

```
C:\Teste>git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   distancia.c

no changes added to commit (use "git add" and/or "git commit -a")

C:\Teste>git add .

C:\Teste>git commit -m "Calculo e impressao juntas"
[master e2e3c0a] Calculo e impressao juntas
1 file changed, 1 insertion(+), 3 deletions(-)

C:\Teste>git log --oneline
e2e3c0a (HEAD -> master) Calculo e impressao juntas
90b3067 (origin/master) calcula a distancia e mostra o resultado
2f7c7f0 Definicao das variaveis
fdf1c36 Funcao que calcula a distancia

C:\Teste>
```



Search or jump to...



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

[jfadul](#) / [projeto-teste](#)

Unwatch ▾

1

★ Star

0

Fork

0

↔ Code

! Issues 0

Pull requests 0

Projects 0

Wiki

Security

Insights

Settings

Projeto de teste para curso do git-github

Edit

[Manage topics](#)

3 commits

1 branch

0 releases

1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾



[jfadul](#) calcula a distancia e mostra o resultado

Latest commit 90b3067 12 hours ago

[.gitignore.txt](#)

Função que calcula a distancia

2 days ago

[distancia.c](#)

calcula a distancia e mostra o resultado

12 hours ago

Help people interested in this repository understand your project by adding a README.

Add a README



Como enviar o repositório local para o Github

```
git push -u origin master
```

Nota: se você usou a opção -u, nas próximas vezes que for subir o branch master, basta fazer:

```
git push
```

Node.js command prompt

```
C:\Teste>git push
Username for 'https://github.com': jfadul
Password for 'https://jfadul@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 349 bytes | 349.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/jfadul/projeto-teste.git
    90b3067..e2e3c0a  master -> master

C:\Teste>_
```

Projeto de teste para curso do git-github

Edit

[Manage topics](#)

🕒 4 commits

🌿 1 branch

📦 0 releases

👤 1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾



jfadul Calculo e impressao juntas

Latest commit e2e3c0a 7 minutes ago

[.gitignore.txt](#)

Função que calcula a distancia

2 days ago

[distancia.c](#)

Calculo e impressao juntas

7 minutes ago

Help people interested in this repository understand your project by adding a README.

Add a README



Calculo e impressao juntas

[Browse files](#)

🔗 master

👤 jfadul committed 8 minutes ago

1 parent 90b3067

commit e2e3c0ace0a651f581783ca24027f201fe1929a2

📄 Showing 1 changed file with 1 addition and 3 deletions.

Unified

Split

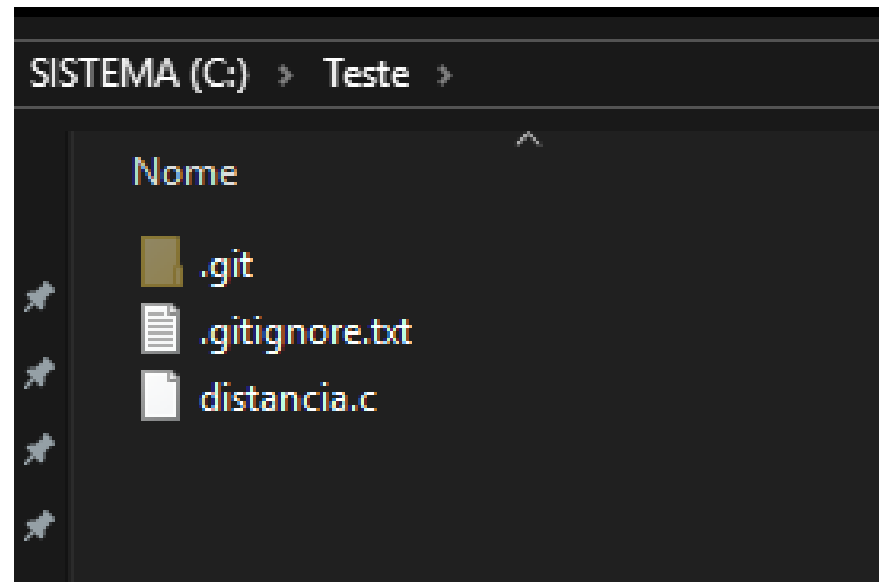
▼ 4 ■ ■ ■ ■ distancia.c 📄

...

@@	-17,10	+17,8	@@
17	17		printf("Informe as coordenadas para o segundo ponto: ");
18	18		scanf("%i %i", &x2, &y2);
19	19		
20	-		// calcula a distancia
20	+		// calcula a distancia e mostra o resultado
21	21		distancia = calculaDistancia(x1,y1,x2,y2);
22	-		
23	-		// mostra o resultado
24	22		printf("Distancia: %f", distancia);
25	23		
26	24		}

📄

Copiando um repositório remoto para seu computador



Deletar a pasta Teste – eliminar da Lixeira.

Copiando um repositório remoto para seu computador

Copiar o repositório, inclusive o histórico de commits:

```
git clone <URI do repositório remoto>
```

jfadul / projeto-teste

Unwatch

1

★ Star

0

Fork

0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Security

Insights

Settings

Projeto de teste para curso do git-github

Edit

Manage topics

4 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download



jfadul Calculo e impressao juntas

.gitignore.txt

Função que calcula a distancia

distancia.c

Calculo e impressao juntas

Help people interested in this repository understand your project by adding a README.

Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/jfadul/projeto-teste.



Open in Desktop

Download ZIP

Copiando um repositório remoto para seu computador

Copiar o repositório, inclusive o histórico de commits:

```
git clone <URI do repositório remoto>
```

```
C:\Users\PC>cd\
```

```
C:\>git clone https://github.com/jfadul/projeto-teste.git_
```

Node.js command prompt

Your environment has been set up for using Node.js 10.16.3 (x64) and npm.

C:\Users\PC>cd\

```
C:\>git clone https://github.com/jffadul/projeto-teste.git
Cloning into 'projeto-teste'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 13 (delta 3), reused 13 (delta 3), pack-reused 0
Unpacking objects: 100% (13/13), done.
```

C:\ Node.js command prompt

```
C:\projeto-teste>dir
```

O volume na unidade C é SISTEMA

O Número de Série do Volume é C2EB-B919

Pasta de C:\projeto-teste

```
06/11/2019  10:48    <DIR>          .
06/11/2019  10:48    <DIR>          ..
06/11/2019  10:48                484 .gitignore.txt
06/11/2019  10:48                637 distancia.c
                2 arquivo(s)          1.121 bytes
                2 pasta(s)  929.787.133.952 bytes disponíveis
```

```
C:\projeto-teste>git log --oneline
```

e2e3c0a (HEAD -> master, origin/master, origin/HEAD) Calculo e impressao juntas

90b3067 calcula a distancia e mostra o resultado

2f7c7f0 Definicao das variaveis

fdf1c36 Funcao que calcula a distancia

```
C:\projeto-teste>
```

Copiando um repositório remoto para seu computador

Copiar o repositório, inclusive o histórico de commits:

```
git clone <URI do repositório remoto>
```

ATENÇÃO: simplesmente copiar os arquivos NÃO traz o histórico de commits!

jfadul / projeto-teste

Unwatch ▾

1

★ Star

0

🍴 Fork

0

<> Code

! Issues 0

🔗 Pull requests 0

📁 Projects 0

📖 Wiki

🛡 Security

📊 Insights

⚙ Settings

Projeto de teste para curso do git-github

Edit

[Manage topics](#)

🕒 4 commits

🌿 1 branch

📦 0 releases

👤 1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾



jfadul Calculo e impressao juntas

📄 .gitignore.txt

Função que calcula a distancia

📄 distancia.c

Calculo e impressao juntas

Help people interested in this repository understand your project by adding a README.

Clone with HTTPS ⓘ

[Use SSH](#)

Use Git or checkout with SVN using the web URL.

[https://github.com/jfadul/projeto-teste.](https://github.com/jfadul/projeto-teste)



Open in Desktop

Download ZIP

Como atualizar seu repositório local

Se outra pessoa da sua equipe alterou a versão do projeto no Github, você precisa sincronizar o trabalho em equipe com outras pessoas.

O Git só deixa você continuar um trabalho e depois subi-lo para o repositório remoto, se você mantiver a sequência coerente de commits.

Como atualizar seu repositório local

Comando para atualizar seu repositório local:

```
git pull origin master
```