FEELT31201
Programação Procedimental

Aula 00:
Git, Github, Printf/Scanf, Controle de Fluxo Simples (if-else, while)

Prof. Igor Peretta

Git, Github

Controle de versões / revisões

Git é um **sistema de controle de versões** distribuído, usado principalmente no desenvolvimento de software, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo (Exemplo: alguns livros digitais são disponibilizados no GitHub e escrito aos poucos publicamente).

O Git foi inicialmente projetado e desenvolvido por **Linus Torvalds** para o desenvolvimento do kernel Linux, mas foi adotado por muitos outros projetos.

Cada diretório de trabalho do Git é um **repositório com um histórico completo e habilidade total de acompanhamento das revisões**, não dependente de acesso a uma rede ou a um servidor central. O Git também facilita a reprodutibilidade científica em uma ampla gama de disciplinas, da ecologia à bioinformática, arqueologia à zoologia.

O Git é um **software livre**, distribuído sob os termos da versão 2 da GNU General Public License. Sua manutenção é atualmente supervisionada por Junio Hamano.

Git, configuração

git config --global user.name “[firstname lastname]”

git config --global user.email “[valid-email]”

Git, inicialização

git init

Git, área de preparação e consolidação

git status

git add [file]

git commit -m “[descriptive message]”

git log

Git, ignorando arquivos

Criar arquivo .gitignore na pasta

Git, ramos e fusão

git branch

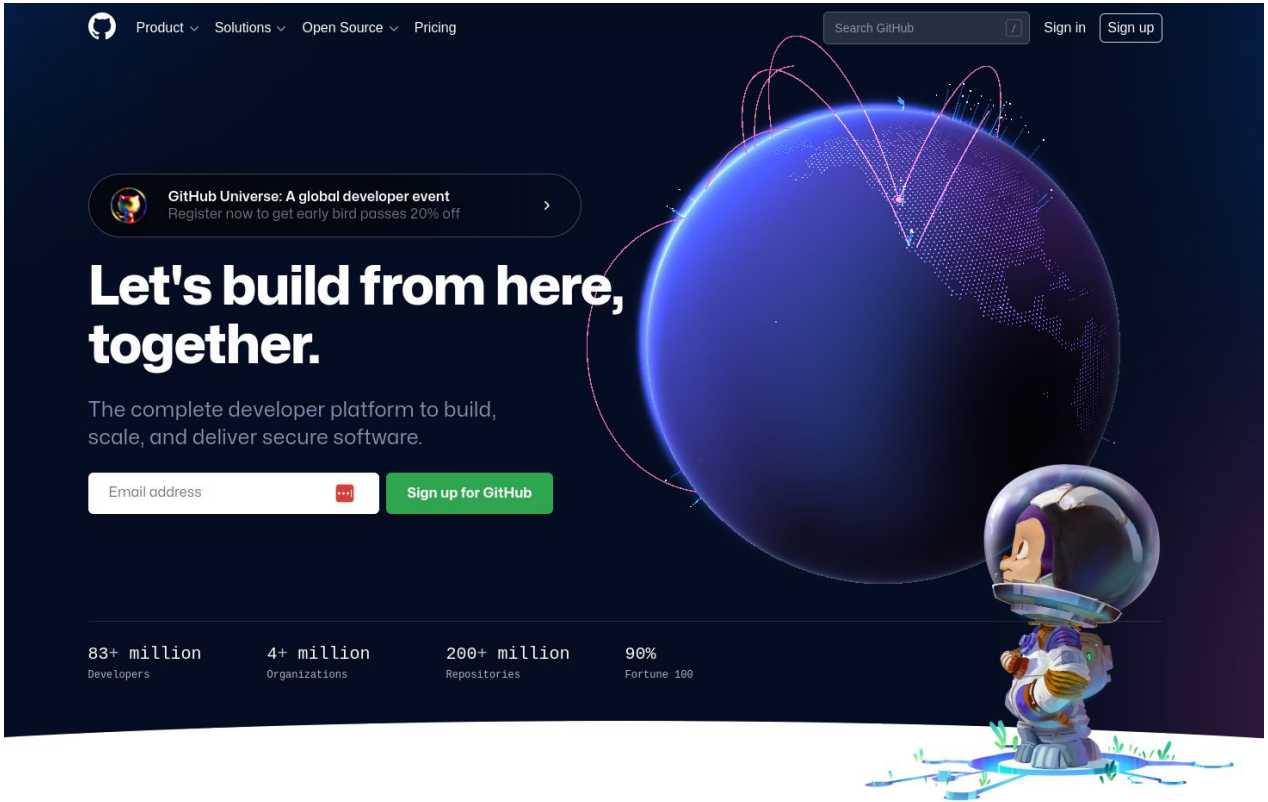
git branch [branch-name]

git checkout

git merge [branch]

Github

Github (https://github.com/)



Git/Github, ligação e atualização

git remote add [alias] [url] (tradição alias = origin)

git remote

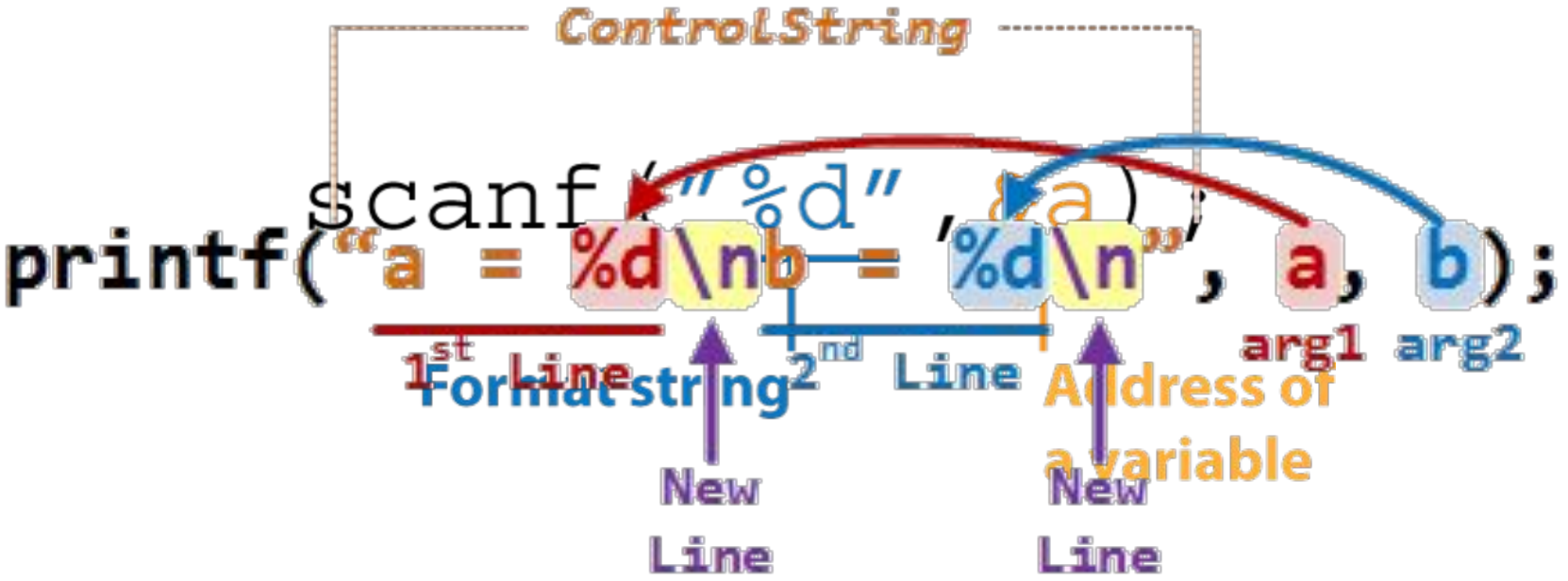
git push [alias] [branch]

git pull

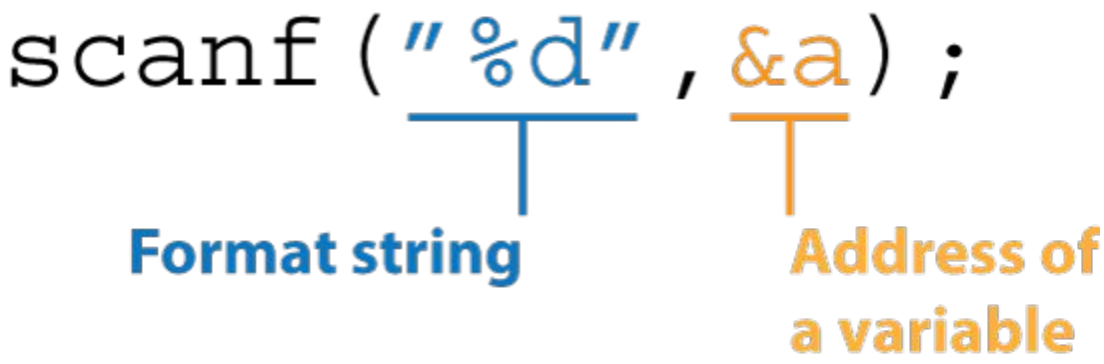
git clone [url]

Programando em C

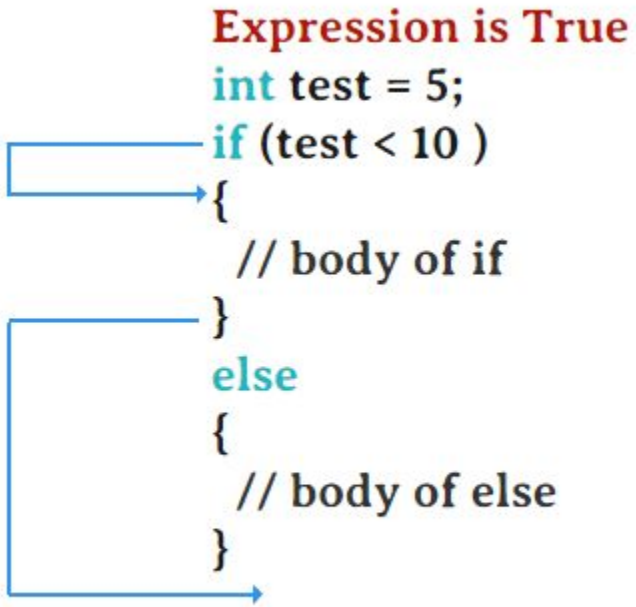
Mostrando informações ao usuário

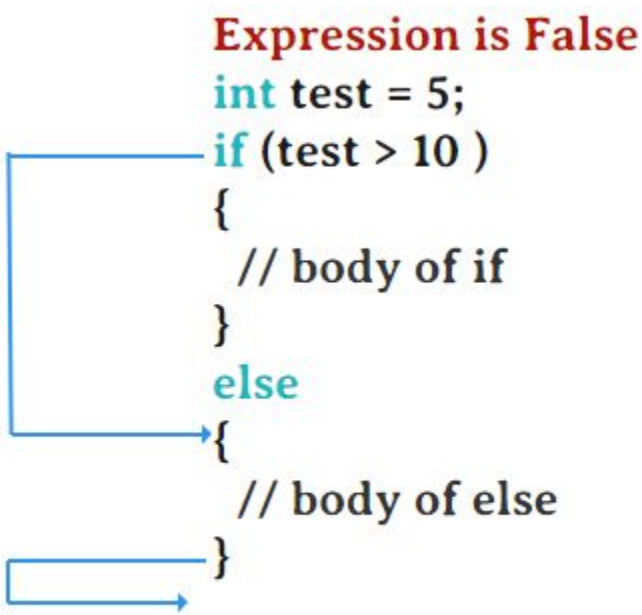


Recebendo informações do usuário

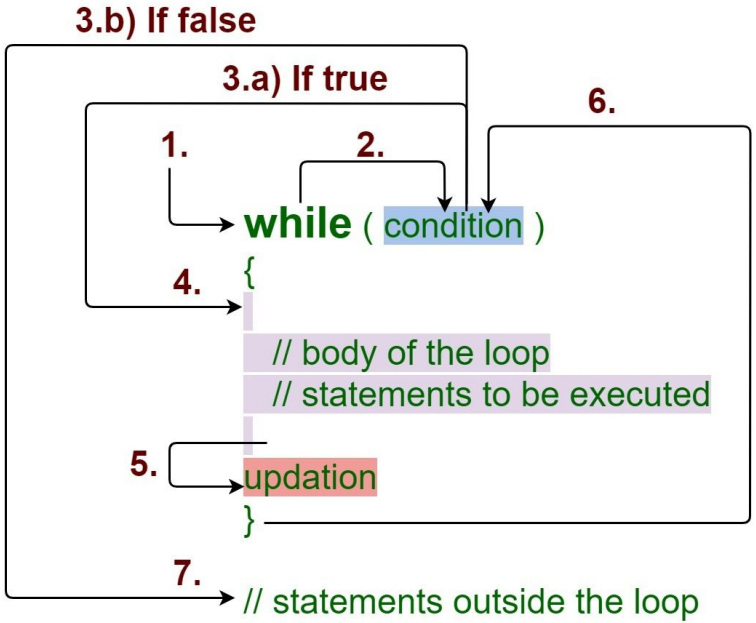


Condição


Expression is True
int test = 5;
if (test < 10)
{
// body of if
}
else
{
// body of else
}


Expression is False
int test = 5;
if (test > 10)
{
// body of if
}
else
{
// body of else
}

Laço


3.b) If false
3.a) If true
6.
1.
2.
while (condition)
4.
{
// body of the loop
// statements to be executed
5.
updation
}
7.
// statements outside the loop

Templates para C

```c
/**
* Arquivo:
* Autor:
* Matricula:
* Criado em:
*/

#include <stdio.h>

int main(void) {

    return 0;
}
```

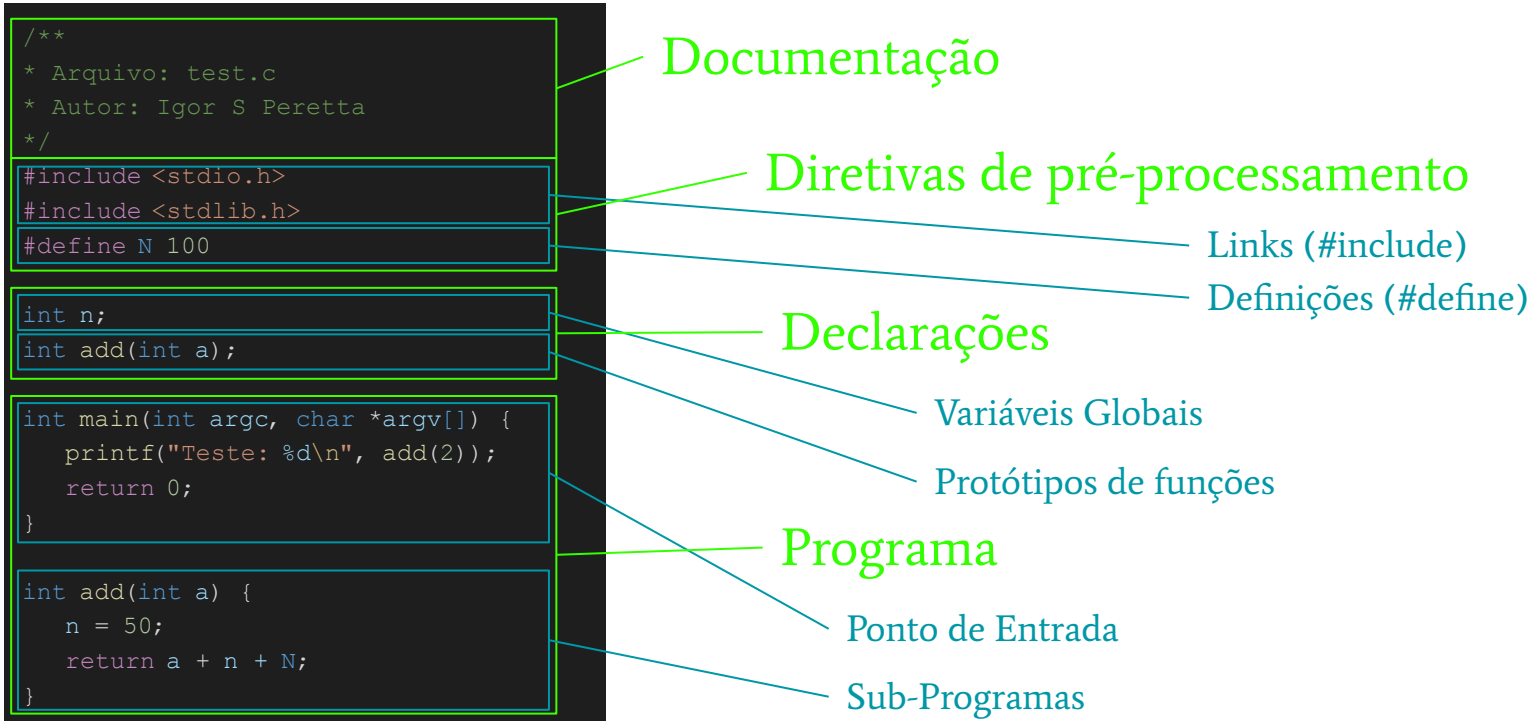
```c
/**
* Arquivo:
* Autor:
* Matricula:
* Criado em:
*/

#include <stdio.h>

int main(int argc, char *argv[]) {

    return 0;
}
```

```c
/**
* Arquivo:
* Autor:
* Matricula:
* Criado em:
*/

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

int main(int argc, char *argv[]) {

    return EXIT_SUCCESS;
}
```

Se falhar, use `EXIT_FAILURE`

Compilando e executando

Arquivo texto com código-fonte (extensão **.c**)



Compilando e executando (terminal, powershell)