

TP 1 de Algoritmos e Estruturas de Dados

Introdução

1. [Introdução](#)
2. [Tipos abstrtos](#)
3. [Modulos e funções](#)
4. [Como compilar o projeto](#)

Tipos Abstratos

Módulos

UTILS

Função	Tipo	Argumentos	Descrição
<code>menu()</code>	<code>void</code>		Imprime na saída padrão as opções de interação do sistema
<code>cls()</code>	<code>void</code>		limpa a saída padrão
<code>uid(int)</code>	<code>char *</code>	<code>int length</code>	Gera um identificador único do processo de tamanho <code>length</code> recebido como parâmetro

Comando para compilar o projeto

```
make
```

Exemplo de compilação

```
make
make clean
make all
```

Comandos principais

- adiciona alteração para ser enviado

```
git add <nome_do_arquivo>
```

- adiciona todas alterações

```
git add .
```

- commitar as alterações

```
git commit -m "mensagem"
```

- enviar as alterações para o repositório remoto

```
git push origin master
```

- baixar as alterações do repositório remoto

```
git pull <nome_do_repositorio> <branch>
```

- verificar o status do repositório

```
git status
```

- verificar o log do repositório

```
git log
```

Compilar a documentação

- Gera arquivo PDF com o mesmo nome do arquivo markdown (não mudar nome do arquivo .md, crie uma cópia com outro nome e compile a cópia, ou mude o nome do arquivo resultado)

```
md-to-pdf readme.md
```

- Para gerar o arquivo PDF, é necessário ter instalado o programa `md-to-pdf`, caso não tenha a ferramenta de build de documentação instalada

```
pip install md-to-pdf  
--- ou ---  
npm install md-to-pdf
```