

Produzindo lindos relatórios com Pweave

Melissa Weber Mendonça

`pythonbrasil[12]`



Logo by winner [Matt Zucker](#)

The International Obfuscated C Code Contest

[[The judges](#) | [IOCCC home page](#) | [How to enter](#) | [FAQ](#) | [Mirrors](#) |
[IOCCC news](#) | [People who have won](#) | [Winning entries](#)]

The [source code](#) for the winners of the 24th IOCCC has been released.

Please see the following news items.

Goals of the Contest

- **Obfuscate:** tr.v. -cated, -cating, -cates.
 1. a. To render obscure.
b. To darken.
 2. To confuse: his emotions obfuscated his judgment.
[LLat. obfuscare, to darken : ob(intensive) + Lat. fuscare, to darken < fuscus, dark.] -obfuscation n. obfuscatory adj
- **The IOCCC:**
 - To write the most Obscure/Obfuscated C program within the rules.
 - To show the importance of programming style, in an ironic way.
 - To stress C compilers with unusual code.
 - To illustrate some of the subtleties of the C language.
 - To provide a safe forum for poor C code. :-)

IOCCC news

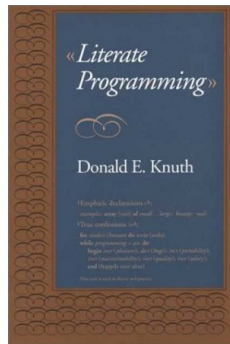
• 2016-04-07

Literate Programming

- Linguagem natural + macros/código
- *Compilável*
- Lógica da máquina v. Lógica humana



Don Knuth (1981)

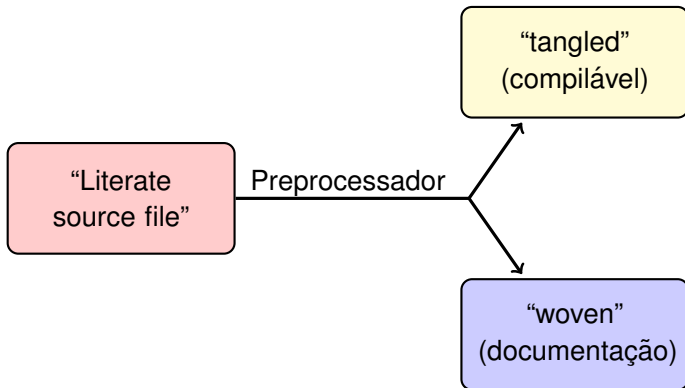


Linguagem natural v. Linguagem de máquina

- Código como literatura
- Descrever a lógica explicitamente
- Documentação orgânica
- Pesquisa e reproducibilidade



Literate Programming: WEB



► wc.lit

Implementações

- WEB — Pascal + T_EX (Knuth)
- CWEB — C + T_EX (Knuth e Silvio Levy)
- Sweave — R + T_EX
- knitr — R + T_EX
- noweb — Linguagem? + Formato? (HTML)
- iJulia
- iPython/Notebooks
- Pweave — Python + Formato?



O que não é Literate Programming

- Código bem documentado
- Relatórios incluindo código e texto
- Documentação automática
- Notebooks escritos de maneira tradicional

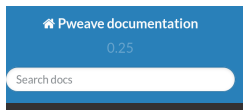


Donald Knuth. "Literate Programming (1984)" in Literate Programming. CSLI, 1992, pg. 99.

Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do.

The practitioner of literate programming can be regarded as an essayist, whose main concern is with exposition and excellence of style. Such an author, with thesaurus in hand, chooses the names of variables carefully and explains what each variable means.





3 Pweave - Scientific Reports Using Python

Features:

Install and quickstart:

Documentation

Thanks

Pweave Basics

Code Chunk Options

Changing defaults

Output Formats

Using pweave module

Publishing scripts

Customizing output

Subclassing formatters

Pweave example gallery

[Docs](#) » Pweave - Scientific Reports Using Python

[View page source](#)

Pweave - Scientific Reports Using Python

Pweave is a scientific report generator and a literate programming tool for Python. Pweave can capture the results and plots from data analysis and works well with NumPy, SciPy and matplotlib. It is able to run python code from source document and include the results and capture [matplotlib](#) plots in the output.

It can produce reST, Sphinx, Latex, HTML and markdown (pandoc and leanpub) output from several input formats.

- Noweb uses [noweb](#) syntax for separating code from documentation.
- Markdown. Run code from markdown code blocks.
- Script. Python script with special markup in comments. See: [publishing scripts](#).

Pweave is good for creating reports, tutorials, presentations etc. with embedded python code It can also be used to make websites together with e.g. Sphinx or rest2web.

There is now also Pweave syntax highlighting for Atom editor using [language-weave](#).

Pweave

- input: documentação e código separados por markup [▶ exemplo](#)

Para gerar um HTML simples:

```
$ pypublish [-t bootstrap/cerulean] ctd.py
```

Para gerar um PDF:

```
$ pypublish -f pdf ctd.py
```

Code chunks em documentos: noweb

Dentro de algum documento .(ext)w:

Texto e documentação na sintaxe do documento.

Para incluir código:

```
<<>>=
```

```
codigo
```

```
@
```

Algumas opções:

- `<<fig = True, width = '12 cm', echo = False>>=`
- Código inline não é incluído no documento: `<% e %>` ou `<%= e %>` (imprime o resultado)



Formatos de saída

- HTML com pygments
- markdown
- pandoc/pandoc2html/pandoc2latex
- tex/texminted/tepygments
- texpweave
- rst
- sphinx
- leanpub

Em ação

Exemplo \LaTeX : [exemplo](#)

```
$ pweave -f tex exemplo.texw  
$ pdflatex exemplo.tex
```

Em ação

Exemplo markdown: [▶ exemplo](#)

```
$ pweave -f pandoc fird.mdw  
$ pandoc -s --mathjax fird.md -o firdpandoc.html
```

Tangling

Podemos extrair o código de um arquivo pweave:

```
$ ptangle fird.mdw
```

Outras ferramentas

- Pandoc
- PythonTeX
- Notebooks!



`github.com/melissawm`
`www.mtm.ufsc.br/~melissa`

Obrigada!

