

Estatísticas de vazão diária no Rio Iguaçu, estação
65310000

Nelson L. Dias

2 de fevereiro de 2018

Trabalho Computacional 1 para a Disciplina TEA-010 “Matemática Aplicada I”

Curso de Graduação de Engenharia Ambiental

Prof. Nelson Luís Dias

Atenção: este trabalho não deve ser entregue: ele faz parte da matéria, e seu conteúdo será cobrado nas provas parciais e na prova final.

1 Arquivo de dados brutos

A listagem abaixo mostra as primeiras linhas e as 67 primeiras colunas do arquivo de dados 65310000.TXT com as vazões do Rio Iguaçu em União da Vitória. O arquivo foi baixado diretamente do sítio da Agência Nacional de Águas.

Você pode baixar o arquivo completo em <https://nldias.github.io/txt/65310000.TXT>.

```
//-----
//
// Sistema de Informações Hidrológicas
// Versão Web 3.0
// © 2001 Agência Nacional de Águas (ANA)
//
// NivelConsistencia: 1 = Bruto, 2 = Consistido
// MediaDiaria: 0 = Não, 1 = Sim
// MetodoObtencaoVazoes: 1 = Curva de descarga, 2 = Transferência de
// Status: 0 = Branco, 1 = Real, 2 = Estimado, 3 = Duvidoso, 4 = Rég
//
// Restrições da consulta:
// Código da Estação: 65310000
//
//-----

//EstacaoCodigo;NivelConsistencia;Data;Hora;MediaDiaria;MetodoObtenc
65310000;2;01/05/1930;;1;1;;;;;0;0;0;0;;;;;125,2;
65310000;2;01/06/1930;;1;1;103,72;70,3;86,04867;3;30;1;1;1;;0;101,15
65310000;2;01/07/1930;;1;1;142,68;05,96;12613;13;1;1;1;1;;0;68,05;68
65310000;2;01/08/1930;;1;1;596;65,8;175,1626;31;7;1;1;1;;0;74,8;72,5
65310000;2;01/09/1930;;1;1;699,4;125,2;281,2767;3;26;1;1;1;;0;651;68
65310000;2;01/10/1930;;1;1;;;;;0;0;0;0;153,6;159,4;183,4;233;339,8
65310000;2;01/11/1930;;1;1;;;;;0;0;0;0;;;;;336,6;311,5
65310000;2;01/12/1930;;1;1;547,4;198,9;332,029;17;8;1;1;1;;0;245,4;2
65310000;2;01/01/1931;;1;1;745;387;519,9645;17;1;1;1;1;;0;387;390,5;
65310000;2;01/02/1931;;1;1;394;233;299,8821;14;23;1;1;1;;0;387;359;3
65310000;2;01/03/1931;;1;1;547,4;171;294,8322;10;31;1;1;1;;0;267,2;2
```

```
65310000;2;01/04/1931;;1;1;257,8;119,6;138,3;30;10;1;1;1;;0;165,2;15
65310000;2;01/05/1931;;1;1;1268;493;808,7968;4;1;1;1;1;;0;493;1052,5
65310000;2;01/06/1931;;1;1;1453;493;950,9767;13;6;1;1;1;;0;603,4;582
65310000;2;01/07/1931;;1;1;922,8;339,8;638,7516;5;29;1;1;1;1;;0;848,6;
```

Sobre o arquivo, é possível notar que:

- Linhas que **não** começam com “65310000” não contêm dados válidos, e devem ser descartadas.
- Os campos do arquivo são separados por ponto-vírgula (“;”).
- Cada linha corresponde a um mês de dados.

Cada linha contém 79 campos:

Campo	Nome	Descrição
0	EstacaoCodigo	Código da Estação
1	NivelConsistencia	1 = Bruto, 2 = Consistido
2	Data	dd/mm/aaaa
3	Hora	Campo vazio
4	MediaDiaria	Sempre igual a 1 (“sim”)
5	MetodoObtencaoVazoes	1 = Curva de descarga, 2 = Transferência de vazões, 3 = Soma de vazões, 4 = ADCP
6	Maxima	Máxima do Mês
7	Minima	Mínima do Mês
8	Media	Média do Mês
9	DiaMaxima	Dia do Mês em que ocorreu a vazão máxima
10	DiaMinima	Dia do Mês em que ocorreu a vazão mínima
11	MaximaStatus	0 = Vazão máxima não foi calculada, 1 = Vazão máxima foi calculada
12	MinimaStatus	0 = Vazão mínima não foi calculada, 1 = Vazão mínima foi calculada
13	MediaStatus	0 = Vazão média não foi calculada, 1 = Vazão média foi calculada
14	MediaAnual	Vazão média anual (sempre vazio)
15	MediaAnualStatus	0 = Vazão média anual indisponível, 1 = Vazão média anual disponível (sempre igual a 0)
16–46	Vazao<dd>	Vazão do dia <dd>; fazio quando indisponível
47–77	Vazao<dd>Status	0 = Vazão do dia <dd> indisponível, 1 = Vazão do dia <dd> disponível
78	–	Sempre vazio

2 Seu trabalho

Escreva um programa em Python que

- Leia o arquivo 65310000 linha a linha, e jogue fora linhas que não começam com 65310000.

- **Utilizando seu programa em Python**, descubra quantos anos existem no arquivo. **Sugestão:** uma maneira simples é ler todo o arquivo, armazenar os anos em uma estrutura do tipo `list` ou `set`, e depois achar o mínimo e o máximo. Em seguida, feche o arquivo e o abra de novo para continuar com o processamento de dados. Vamos chamar o número de anos disponíveis de `<nanos>`.
- Crie um array tridimensional `Q[<nanos>,12,31]`. Inicialize o array com `nan` (*not a number*), utilizando `math.nan` e a função `full` de `numpy`.
- Volte a percorrer o arquivo linha a linha.
- Para cada linha válida: elimine os caracteres de fim de linha utilizando o método `rstrip`. Informe-se sobre como ele funciona. Para cada linha válida separe os campos utilizando o método `split`. Informe-se sobre como ele funciona.
- Cada linha corresponde a um mês de vazões. Preencha `Q` até o número de dias existente em cada mês, com o auxílio da função `monthrange` do módulo `calendar`. Informe-se sobre eles.
- Quando a vazão de um dia está indisponível, o comprimento da *string* do campo correspondente é zero (campo vazio). Descubra todas as datas em que o dado de vazão não está disponível. A resposta para você comparar está no arquivo <https://nldias.github.io/txt/faltas.txt>.
- O ano 1930 é muito incompleto. Descarte o ano 1930 do array `Q`.
- Para cada dia entre 1 e 365 tire a média de todo o histórico *a partir de 1931* (`<nanos>-1`), usando apenas anos em que o dado do dia está disponível. Esqueça os dias 29 de fevereiro! A resposta para você comparar está no arquivo <https://nldias.github.io/txt/qmeddia.txt>