

# Dengue Free Feira

## Problema 2 - MI Algoritmos

Maria Clara N Ramos

<sup>1</sup>Departamento de Exatas – Universidade Estadual de Feira de Santana (UEFS)  
Caixa Postal: 44.036-900 – Feira de Santana – BA – Brazil

m clara9593@gmail.com

**Abstract.** *Dengue is an endemic disease that is in an outbreak period, and needs to be controlled. A company requested a surveillance system providing the necessary data to analyze the endemic, and thus have a basis for taking the necessary measures. The report refers to the production of a surveillance system created in Python 3.12, where it is possible to obtain and edit updated data regarding the spread of the disease*

**Resumo.** *A dengue é uma doença endêmica que está em período de surto, e precisa ser controlada. Foi solicitado por uma empresa um sistema de vigilância onde sejam fornecidos os dados necessários à análise da epidemia, e assim, ter uma base para tomar as devidas medidas. Este relatório se refere a produção de um sistema de vigilância criado em python 3.12, onde é possível obter e editar dados atualizados referentes a propagação da doença.*

### 1. Introdução

A dengue é uma doença que necessita de monitoramento frequente, devido à sua rápida disseminação. Uma empresa que atua com vigilância epidemiológica necessita de uma ferramenta que possibilite o controle da epidemia, em surto na cidade de Feira de Santana. Assim, será mais fácil ter uma visualização da situação por localidade, e tomar medidas proporcionais, visando evitar sua propagação. Foi solicitado aos alunos de Engenharia de Computação da Universidade Estadual de Feira de Santana (UEFS), um software que cumpra com os requisitos relacionados às funcionalidades de um sistema de vigilância, capaz de prover e manipular dados precisos a respeito da incidência.

A solução desenvolvida foi um código fonte escrito em *python*, capaz de realizar a leitura de arquivos no formato *.csv* (Comma-Separated Values). Caso não seja encontrado, é retornado um erro, e criado um arquivo com o nome informado e na mesma pasta do arquivo *.py*. Este, possui um cabeçalho pré-definido, contendo, para cada linha, uma data (que pode se repetir); um bairro; a quantidade de habitantes do bairro; casos suspeitos; casos confirmados e casos negados.

Em se tratando da leitura de dados, o software possibilita extração de informações relacionadas à propagação da doença, a partir de filtros aplicados conforme a opção selecionada pelo usuário (estes, com base na data e no bairro informados). Os números dos casos são passíveis de alteração - uma vez que o sistema tem de se manter atualizado - e sua mudança reflete nas informações existentes, referentes à última data.

## 2. Metodologia e Fundamentação Teórica

Para obtenção dos resultados foram, aplicados conhecimentos construídos durante as sessões PBL (Ensino Baseado em Problemas), tanto através das discussões em grupo como por intermédio do tutor Ângelo; pesquisas em fóruns como stackoverflow; vídeo aulas do professor Guanabara, e aulas presenciais ministradas pela professora Michele.

Durante as sessões, foi discutido a respeito das requisições do menu principal e modularização do código, visando deixá-lo mais lúcido e organizado. Ainda almejando tais propriedades, se tornou de consenso geral o uso da formatação que simula uma tabela nas saídas do código. Não obstante, foram definidos como um fatos:

- O “csv” deve ser manipulado indiretamente, com uso da sua própria biblioteca como auxílio, sendo transposto como *lista* de *listas(matriz)* ou *dicionário* a fim de evitar a abertura constante do arquivo;
- O uso da biblioteca `datetime` como auxiliar, e também para acrescentar dias automaticamente

### 2.1. Definição de Requisitos

As funcionalidades do sistema requisitadas foram:

1. Realizar a leitura de um arquivo que contém o boletim de ocorrências de Dengue em cada bairro;
2. Disponibilizar as informações de :
  - 2.1. Dados totais de habitantes, casos positivos, suspeitos e negativos
    - 2.1.1. Geral;;
    - 2.1.2. Por bairro
    - 2.1.3. Por data;
    - 2.1.4. Por intervalo de data;
  - 2.2. Porcentagem de casos confirmados e negados:
    - 2.2.1. Por bairro;
    - 2.2.2. Por intervalo de data;
    - 2.2.3. Geral;
3. Atualização de casos:
  - 3.1. Suspeitos;
  - 3.2. Confirmados, e influência nos suspeitos;;
  - 3.3. Negativos, e influência nos suspeitos
4. Atualização do arquivo .csv

### 2.2. Descrição de alto nível

O programa não possui interface, ofertando apenas interação direta através do terminal onde é solicitado primeiramente o nome do arquivo.csv. Através de um tratamento de erro com `try/except`, considera-se a criação de um arquivo na mesma pasta se o informado não for encontrado; caso contrário, o usuário é direcionado para o Menu Principal.

Tal funcionamento se reparte em três casos: leitura; atualização de dados e saída do programa. Por trás de cada seleção do menu , ocorre a chamada de *funções* correspondentes, elencadas dentro do código em 3 seções diferentes.

```

Digite o nome do documento que deseja abrir,","formato (exemplo:DadosDengue.csv): DadosDengue.csv
|-----DENGUE-FREE-FEIRA-----|
|-----LEITURA-----|
|
[1]Informações sobre a dengue
[2]Cadastro
[3]Sair
Opção:

```

**Figura 1. Menu principal**

No que se trata da seção de leitura de arquivo, correspondente a escolha “[1] Informações sobre a dengue”, a mesma é seccionada conforme entrada do usuário, por filtros construídos a partir de *condicionais* sob as *listas* e organizadas em *funções*, guiadas pelas seguintes ramificações:

- Data específica (no formato dia/mês/ano);
- Bairro;
- Intervalo de datas;

Na mesma seção se encontram as funções responsáveis pelo cálculo da porcentagem (Figura 2), exibida em conjunto com a tabela referente a localidade solicitada, portando como parâmetro o bairro informado (referente ao requisito 2.2.1). Ademais, a porcentagem geral de casos, (referente ao requisito 2.2.3), é transmitida junto à tabela exibida em “[4] Boletim”.

```

|-----DENGUE-FREE-FEIRA-----|
|-----LEITURA-----|
|
[1]Dados por data específica
[2]Dados por bairro
[3]Dados por intervalo de data
[4]Boletim completo
Opção:

```

**Figura 2. Menu Secundário**

No que concerne ao terceiro segmento, análogo à “[2] Cadastro” (Figura 1), apresenta outros três acessos nos quais correspondem às funções responsáveis por atualizar o sistema. Cada seleção (Figura 3) possui como entradas o número de casos a acrescentar e bairro correspondente- sendo alterados na última data- e, como saída, uma tabela com dados alterados.

```

|-----DENGUE-FREE-FEIRA-----|
|-----CADASTRAMENTO-----|
|
Alterar número de:
[1]Suspeitos de dengue
[2]Casos positivos
[3]Casos negativos

```

**Figura 3. Menu Terciário**

Nesse íterim, há ainda uma quarta seção contendo a *função* que passa o conteúdo da matriz atualizada para o arquivo .csv, chamada ao fim das três *funções* de alteração de

dados ,e a *main* ,que abrange o menu principal.

Outrossim,encontra-se no início do código uma seção de formatação, composta por *funções* nas quais foram atribuídas as seguintes finalidades: limpar o terminal; importar o arquivo.csv como uma *matriz*; criar o arquivo; exibir a tabela completa com base neste arquivo já convertido,de forma organizada; acrescentar datas automaticamente com *datetime*; converter valores numéricos para formato *int*,a fim de possibilitar cálculos. Tais operações serevem de auxílio, e são *chamadas* em outras seções.

### 2.3. Ordem de Codificação

A ordem de codificação foi norteadada pelo menu.A partir dele foram listadas as funções necessárias a serem desenvolvidas. Inicialmente foi pensado na leitura e formatação do arquivo csv, seguido de funções auxiliares, presentes na seção de formatação indicada por um comentário no código. Posteriormente, foi desenvolvida a seção de leitura e interpretação,a partir do parâmetro informado pelo usuário, caso se encaixe no formato do cabeçalho definido como base para um boletim epidemiológico. Após isso,foram criadas as funções responsáveis pela edição do arquivo

O produto foi construido na linguagem *Python*, na versão 3.12, em com sistema operacional Windows 10 e através da ferramenta Visual Studio Code.

## 3. Resultados e Discussões

Para utilizar o programa é necessário ter um interpretador do Python instalado no dispositivo,e de um arquivo .csv localizado na mesma pasta que o *DengueFreeFeira.py*

Ao iniciar a execução,após abertura do arquivo .py, o usuário deve informar o nome do arquivo .csv,e então serão exibidas as seguintes opções (nas quais, para acessar a opção desejada,o usuário deve entrar com o número correspondente):

### 1 Informações sobre a dengue

- 1.1 Data específica
- 1.2 Bairro
- 1.3 Intervalo de data
- 1.4 Boletim completo

### 2 Cadastro

- 2.1 Suspeitos de dengue
- 2.2 Casos negativos
- 2.3 Casos positivos

### 3 Sair

A selção 1.1 (corresponde à 1 do menu secundário,e assim sucessivamente),a 3º e última sai de um *loop* e fecha o programa .São listadas a seguir, as entradas e saídas referentes a cada item, respectivamente:

1.1 Data ; Tabela com dados do boletim, de todos os bairros,referentes a data informada

1.2 Bairro ; Tabela com dados do boletim referentes ao bairro informado,apenas da última data,e a porcentagem de casos positivos e negativos referentes aos habitantes

1.3 Data “inicial” e “final” ; Tabela com dados do boletim, de todos os

bairros, referentes as data informadas, seguido da diferença bruta e percentual de casos

1. Sem entrada ;Tabela com todos os bairros e datas, e porcentagem de casos negativos, positivos e suspeitos em relação ao total de notificados

## 2 Cadastro

2.1 Bairro, quantidade de suspeitos a acrescentar; Tabela da última data e do bairro, com dados atualizados

2.2 Bairro, quantidade de negativos a acrescentar; Tabela da última data e do bairro, com dados atualizados

2.3 Bairro, quantidade de positivos a acrescentar; Tabela da última data e do bairro, com dados atualizados

A respeito dos testes, foram direcionados principalmente com relação ao processamento do programa, visando verificar se as saídas correspondiam aos cálculos e trechos do arquivo em que se desejava retirar informações, e nem tanto nos problemas que podem ser causados pelo próprio usuário. Limitando-se às entradas no formato solicitado, o programa cumpre com a maioria dos requisitos, exceto pela impossibilidade de encontrar o arquivo csv caso esteja fora da pasta.

Nesse último caso, foi testada uma função que tenta encontrar o arquivo através do caminho, utilizando como auxílio a biblioteca `os`, mas retornava o erro: “permission denied”. Também foi testada uma função para acrescentar bairros, caso o arquivo tenha apenas cabeçalho, mas sua funcionalidade não foi eficiente, então foi apagada.

## 4. Conclusão

Tendo em vista a solução proposta é possível concluir que o produto cumpre com a maioria dos requisitos, na medida em que:

### 1. Possui um menu de opções

1. Seleção de data ou intervalo para visualização de dados;
2. Porcentagem de casos de pessoas suspeitas e confirmadas por bairro;
3. Porcentagem de casos suspeitos, confirmados e negados de acordo com o total de casos notificado;
4. Total de habitantes de casos suspeitos, negativos e positivos cadastrados;
5. Atualização de casos suspeitos
6. Atualização de casos positivos e negativos refletindo nos suspeitos

Não obstante, tais informações são passíveis de acesso por parte do usuário, de maneira organizada através do terminal. O software foi devidamente modularizado, uma vez que sua repartição em blocos - comentados de maneira adequada - tornou o código fonte mais limpo, evitando repetições. Tanto as funções, quanto as variáveis possuem nomes e tipos adequados, e, quanto ao arquivo, é manipulado de modo que é possível realizar sua leitura e alteração.

Diante de tal solução se fazem necessárias melhorias quanto a manipulação do arquivo, posto que o `.csv` só é importado caso alocado na mesma pasta que o `.py`. Ademais, visando otimizá-lo, seria de melhor funcionalidade a aplicação de verificações através de laços de repetição ou funções como *isdigit*, bem como de uma interface interativa.

## 5.Referências

Foundation, P. S. (2024). Python 3.12.3 documentation. <https://docs.python.org/3/>.

Guanabara,G.(2019) “Execício Pyhton #115a- Criando um menu em Python”[https://youtu.be/JGztEBLGj5E?si=2\\_BYpv\\_hYbMoF0N](https://youtu.be/JGztEBLGj5E?si=2_BYpv_hYbMoF0N), Outubro

<https://stackoverflow.com/questions/12201928/open-gives-filenotfounderror-ioerror-errno-2-no-such-file-or-directory>

Guanabara,G.(2018) “Exercício Python #087 - Mais sobre Matriz em Python”(6421)  
[Exercício Python #087 - Mais sobre Matriz em Python - YouTube](#)