

RELATÓRIO EP2 (27/04)

Construção do grafo de encontros

| | |
|--------------------------------|----------|
| Alexandre Kenji Okamoto | 11208371 |
| Daniel Feitosa dos Santos | 11270591 |
| Fernanda Cavalcante Nascimento | 11390827 |
| Giovani Verginelli Haka | 11295696 |
| Larissa Yurie Maruyama | 11295928 |
| Luísa Dipierri Landert | 8010698 |
| Matheus Antonio Cardoso Reyes | 11270910 |
| Otávio Nunes Rosa | 11319037 |

REPOSITÓRIO NO GITHUB

<https://github.com/matheus-reyes/AEDII-Grafos/tree/master/EP2>

CENÁRIO ESCOLHIDO

1. Tudo está funcionando normalmente: neste cenário são consideradas todas as origens e todos os destinos e as arestas são criadas entre quaisquer pessoas que frequentem os mesmos lugares (origem e/ou destinos coincidentes).

COMO O PROGRAMA FOI FEITO

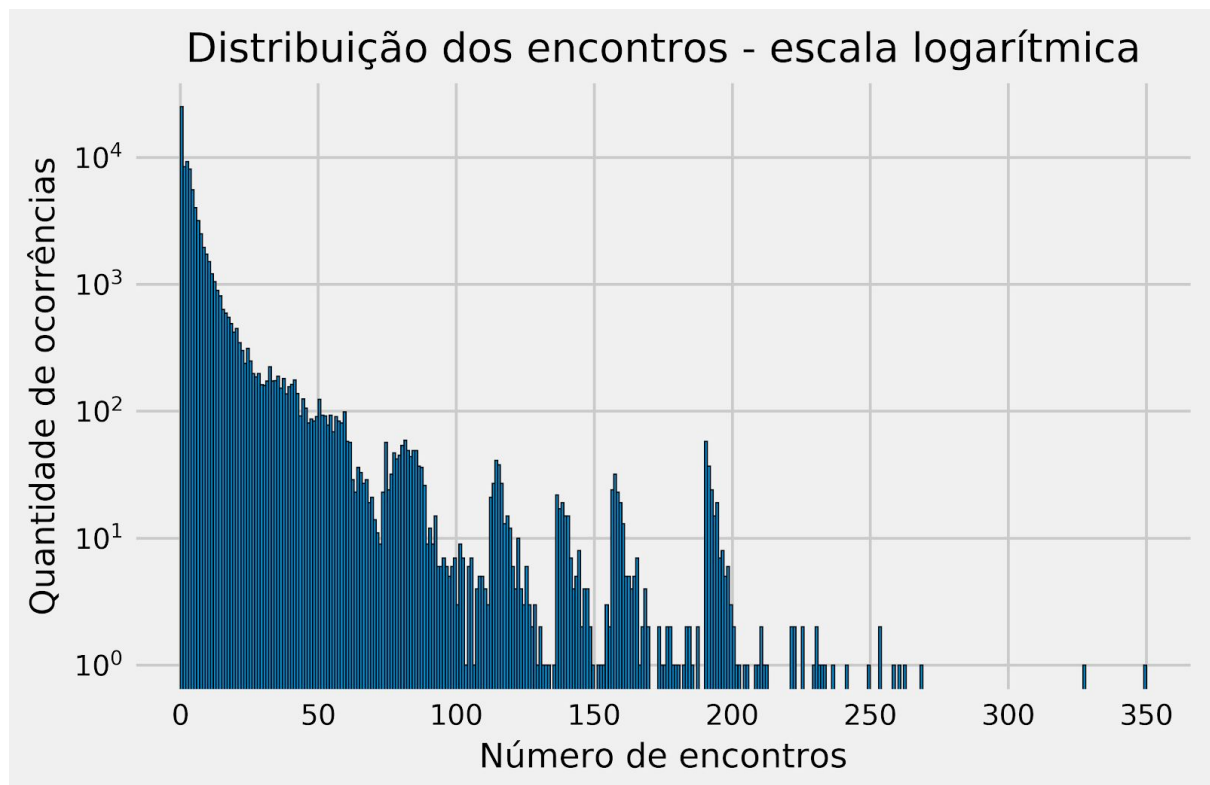
O programa foi realizado em **Python**, sem a utilização de Panda, e com o auxílio do plug-in Live Share para o Visual Studio Code para possibilitar o acesso a todos os integrantes. Como o grupo optou pela estrutura de grafo *array of adjacency lists*, foram criadas duas classes para facilitar a organização:

1. A classe **Grafo** contém um construtor que cria uma variável para os nós e recebe os parâmetros e depois cria um vetor vazio de adjacência, que será preenchido com os dados fornecidos pela outra classe. Há duas funções print que auxiliam nos testes, mas não é chamado. Enfim, exporta os dados para uma nova tabela .csv para a realização do histograma da distribuição de nós.

2. A classe **Processar** importa os arquivos .csv para processar a relação de IDs e .txt para acessar os encontros das pessoas. Em seguida, cria lista para os vértices e arestas e percorre os dados da tabela .csv; percorre as linhas com informações das arestas. Transforma a lista de duas dimensões para uma lista de tuplas. A classe **Grafo** é chamada e há a organização dos nós e as quantidades. Por fim, a classe fecha os arquivos .csv e .txt utilizados.

Também foi utilizado o Colab Notebook, do Google, que facilita a visualização e análise de dados, para melhor montagem do histograma. Então, a partir da tabela de saída, utilizou-se a ferramenta, junto com a biblioteca Panda, para organizar a distribuição de encontros e, para melhorar ainda mais a visualização, foi feita em escala logarítmica.

HISTOGRAMA DOS GRAUS DOS NÓS



TEMPO REAL DE PROCESSAMENTO TOTAL (EM SEGUNDOS)

| | |
|--|--------------------|
| Dell Inspiron 5584 - i7 (8º Geração) 8565U - 8GB RAM - SSD 128GB / HDD 1TB | 0.6991288661956787 |
| Lenovo Ideapad 330 - i7 (8ª Geração) 8550U - 8GB RAM - SSD 550GB | 0.7271697521209717 |
| Acer Aspire 5 - i5 (8ª geração) 8265U - 16GB RAM - SSD 512GB | 0.7343685626983643 |
| Lenovo Legion Y7000 - i7 (8ª Geração) 8750H - 8GB RAM - SSD 550GB | 0.7575170993804932 |
| Lenovo ThinkPad T430 - i5 (3º Geração) 3320M - 8GB RAM - HDD 500GB | 0.8015604019165039 |
| MacBook Pro Retina - i5 Dual-Core - 8GB RAM | 1.0664129257202148 |
| Samsung Essentials E31 NP370E4K - i3 (5ª Geração) 5005U - 4GB RAM | 1.3802094459533691 |
| Acer Aspire V3 - i7 (3ª Geração) 3632QM - 4GB RAM - HD 500GB | 1.7243523597717285 |
| Itautec InfoWay Note W7415 - Pentium(R) Dual-Core CPU T4500 - 3GB RAM - HD 500GB | 2.3431339263916016 |

ESTIMATIVA DO TEMPO DE PROCESSAMENTO ASSINTÓTICO

O processamento e o uso de memória será proporcional ao tamanhos dos vértices e das arestas, portanto:

$$O(v + e)$$