

Aplicação de Avaliação de Desempenho e Tempo de Execução

Enviroment: Python 3.6

Implementar métodos de redução de banda de matrizes com base nas heurísticas apresentadas em http://www.sbmac.org.br/arquivos/notas/livro_75.pdf.

- Implementar uma das Heurísticas apresentadas no livro;
- Implementar o Gradiente Conjugado com Técnica de Armazenamento de Linha (**CSR**);
- Comparar a Heurística escolhida e o Gradiente Conjugado com relação ao tempo de processamento.

OBS.: Usar matrizes grandes com aqueles esquemas de compactação. Para maiores detalhes, acesse este catálogo de matrizes <https://sparse.tamu.edu>.

Como utilizar

Primeiramente, baixe o repositório para a sua máquina. Após baixá-lo, o repositório **raiz** (*cefet-rj-algebra-final*) terá a seguinte estrutura:

- Arquivos:
 - README.md
 - application.py
- Diretórios:
 - datasets
 - modules
 - tests

Em seguida, acesse o diretório **datasets** na sua máquina e verifique os arquivos de matrizes do tipo *.mtx. Esses arquivos serão carregados na etapa de execução. Feito isso, basta executar o comando no formato padrão abaixo:

- **python application.py [datasets/'file'.mtx]**

Exemplo

- **python application.py datasets/bcsstk18.mtx**

Com o comando acima, será executado o formato padrão da aplicação, que contempla o seguinte modo de execução:

- **python application.py 100 0.01 False datasets/bcsstk18.mtx**

Os parâmetros representados pelos valores **default 100**, **0.01** e **False** representam, respectivamente, máximo de iterações (**--maxit**), erro de tolerância (**--err**) e a propriedade simétrica da matriz (**--symetric_mode**).

Para conhecer os parâmetros, execute **python heuristic_cuthillmckee.py -h**

OBS.: É necessário ter o Python (3.6) instalado em sua máquina.

Por fim, para executar o **Aplicação** com outros parâmetros, basta executar o arquivo *application.py* da seguinte forma:

- Para matrizes não-simétricas:
 - **python application.py Num-INT Num-FLOAT datasets/'file'.mtx**
- Para matrizes simétricas:
 - **python application.py Num-INT Num-FLOAT True datasets/'file'.mtx**

Resultado

Ao final da execução será apresentado um relório descrevendo a execução da aplicação e apresentando os resultados comparativos. Segue um exemplo do resultado de execução abaixo:

```

-----
[RELATORIO] - Descricao da Execucao
-----

Aplicando o Metodo Iterativo GRADIENTE-CONJUGADO
      Inicio - 2018-06-03 22:35:27.449495
      Termina - 2018-06-03 22:35:27.451965
Tempo de Execucao do Metodo Iterativo GRADIENTE-CONJUGADO:
0.002469778060913086

Aplicando a Heuristica REVERSE-CUTHIL-MCKEE
      Inicio - 2018-06-03 22:35:27.456298
      Termina - 2018-06-03 22:35:27.456473
Tempo de Execucao da Heuristica REVERSE-CUTHIL-MCKEE:
0.00017523765563964844

```

```

-----
[SUMARIO] - Apresentando a Matriz
-----

Dimensao (NxN):      (153, 153)
Elementos NONZERO:    2423
Arquivo da Matriz:    datasets/input153x153.mtx
Matriz Simétrica:     True ==> Execução mais otimizada ;)

```

```
[RESULTADO] - Avaliacao do Tempo de Execucao
```

```
-----  
REVERSE-CUTHIL-MCKEE teve DESEMPENHO MELHOR que GRADIENTE-CONJUGADO  
-----
```

```
[Ajuda] Para conhecer os parâmetros, execute: 'python application.py -h'  
[Exemplo] Para execucao padrao, execute: 'python application.py  
datasets/<FILE>.mtx'
```

Para conhecer os parâmetros, execute **python application.py -h**. Então "mãos à obra" e bons resultados.

Abs =D

Agradecimentos

- **Gustavo Alexandre**
- **Marcelo Silveira**
- **Raphael Fialho**