

---

# Exercício Programa 1

## Resolução de sistemas de equações lineares usando decomposição LU em C e Fortran

---

São Paulo, 5 de outubro de 2014

**Bruno Sesso**

**8536002**

**Gustavo Estrela de Matos**

**8536051**

---

## Introdução

O objetivo desse trabalho é analisar o tempo que leva a resolução de um sistema de equações lineares utilizando a decomposição LU. O algoritmo é implementado de três formas distintas:

1. Na linguagem C
2. Na linguagem Fortran
3. Na linguagem C, convertido de Fortran (utilizando o programa f2c)

Foi escolhido dessa forma, pois C é uma linguagem orientada a linhas e Fortran a colunas. Assim, conseguimos comparar se de fato há uma diferença relevante entre as orientações das implementações para cada linguagem.

### Arquivos entregues

Na pasta “c” encontra-se o arquivo `c_ep1.c` com o código em c do EP. Na pasta “fortran” encontram-se os arquivos `f_ep1.f` e `f_ep1.c` respectivamente com o código em fortran e com o código em c gerado pelo programa `f2c`.

Para compilar os arquivos `c_ep1.c` e `f_ep1.f` basta o uso dos compiladores gcc e gfortran sem nenhuma flag especial. Para compilar o arquivo `f_ep1.c` utilize o comando abaixo:

```
gcc lu_f77_solver.c -L/usr/lib -lf2c
```

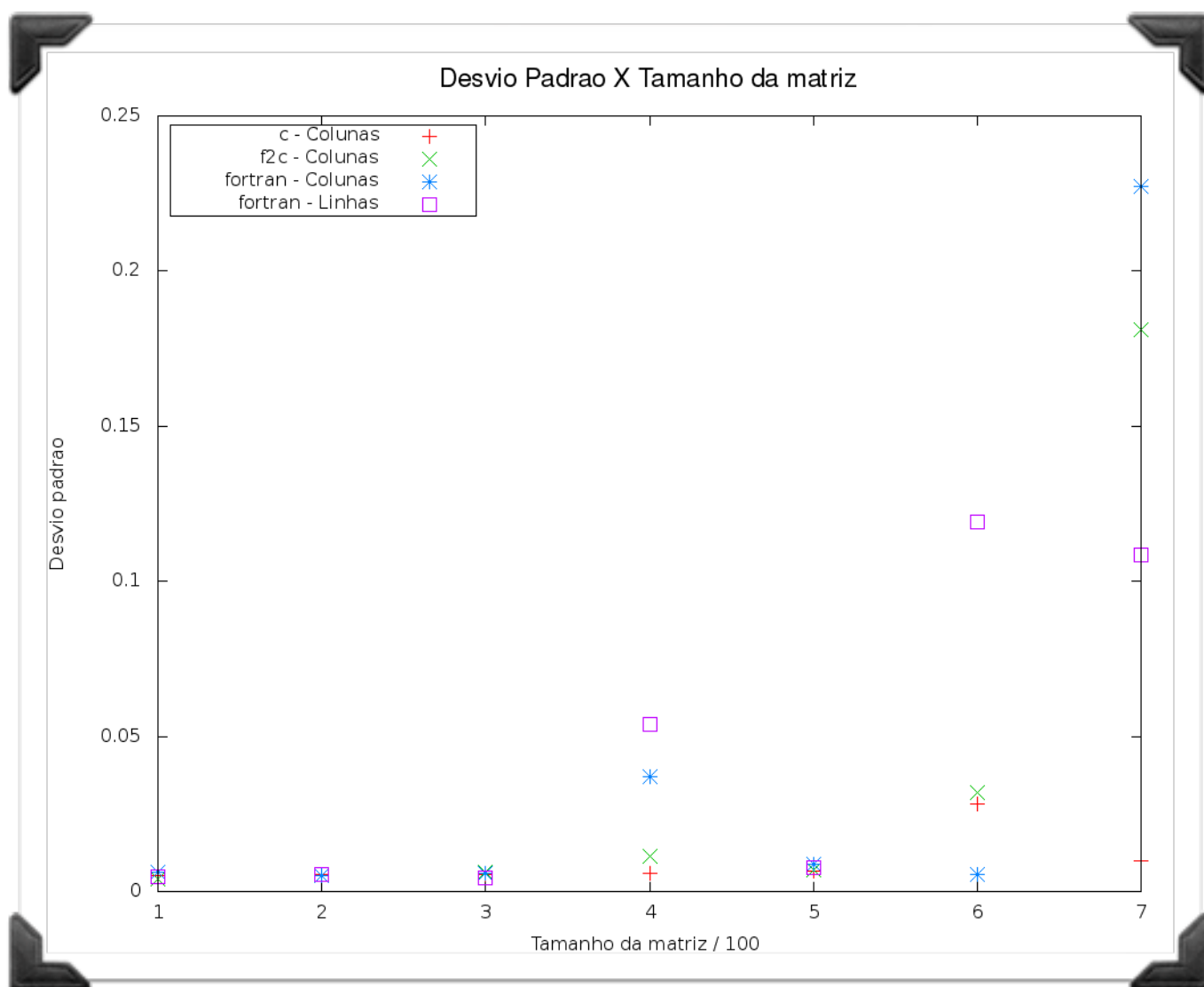
Além dos códigos do exercício programa e deste relatório, foram entregues os arquivos com os resultados dos testes feitos para a elaboração dos gráficos. Os mesmos se encontram na pasta “resultados”.

Para executar os programas, usa-se a flag ‘-c’ para utilizar o algoritmo orientado a colunas e caso não use essa flag sera executado o algoritmo orientado a linhas. Também é aceito a flag ‘-q’ que omite a impressão do vetor ‘x’ (solução).

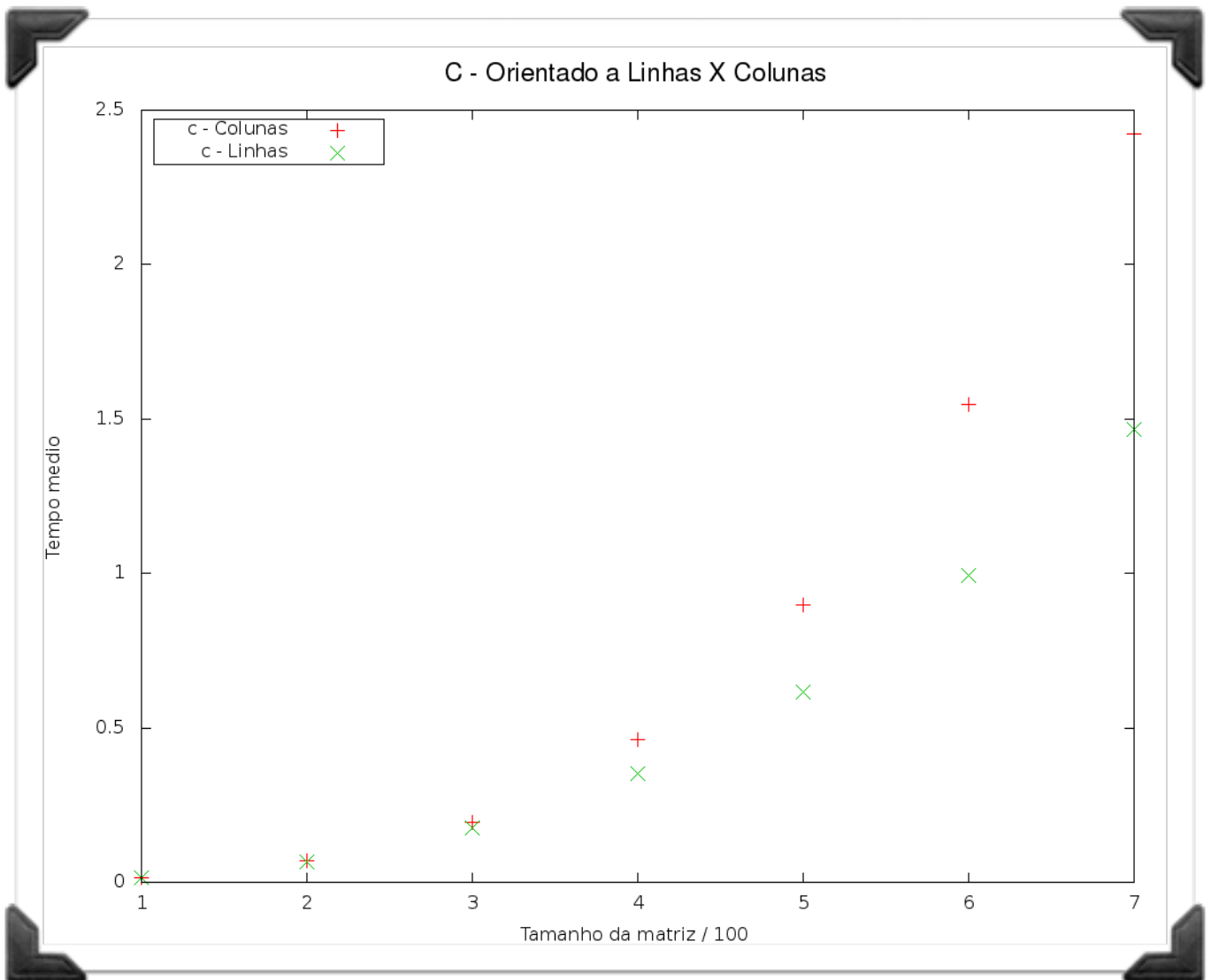
## Testes

Os testes foram feitos utilizando o programa *time*. Para cada matriz gerada por *genmat.c* foi rodado cem vezes cada versão do algoritmo para cada uma das linguagens. Após os testes serem feitos, calculamos a média de tempo e construímos os gráficos usando *gnuplot*.

Para garantir que os dados são relevantes, calculamos o desvio padrão para cada uma das amostras de tempo. Como podemos ver no gráfico abaixo, o desvio padrão foi pequeno, portanto, podemos considerar que as médias calculadas são estatisticamente relevantes.

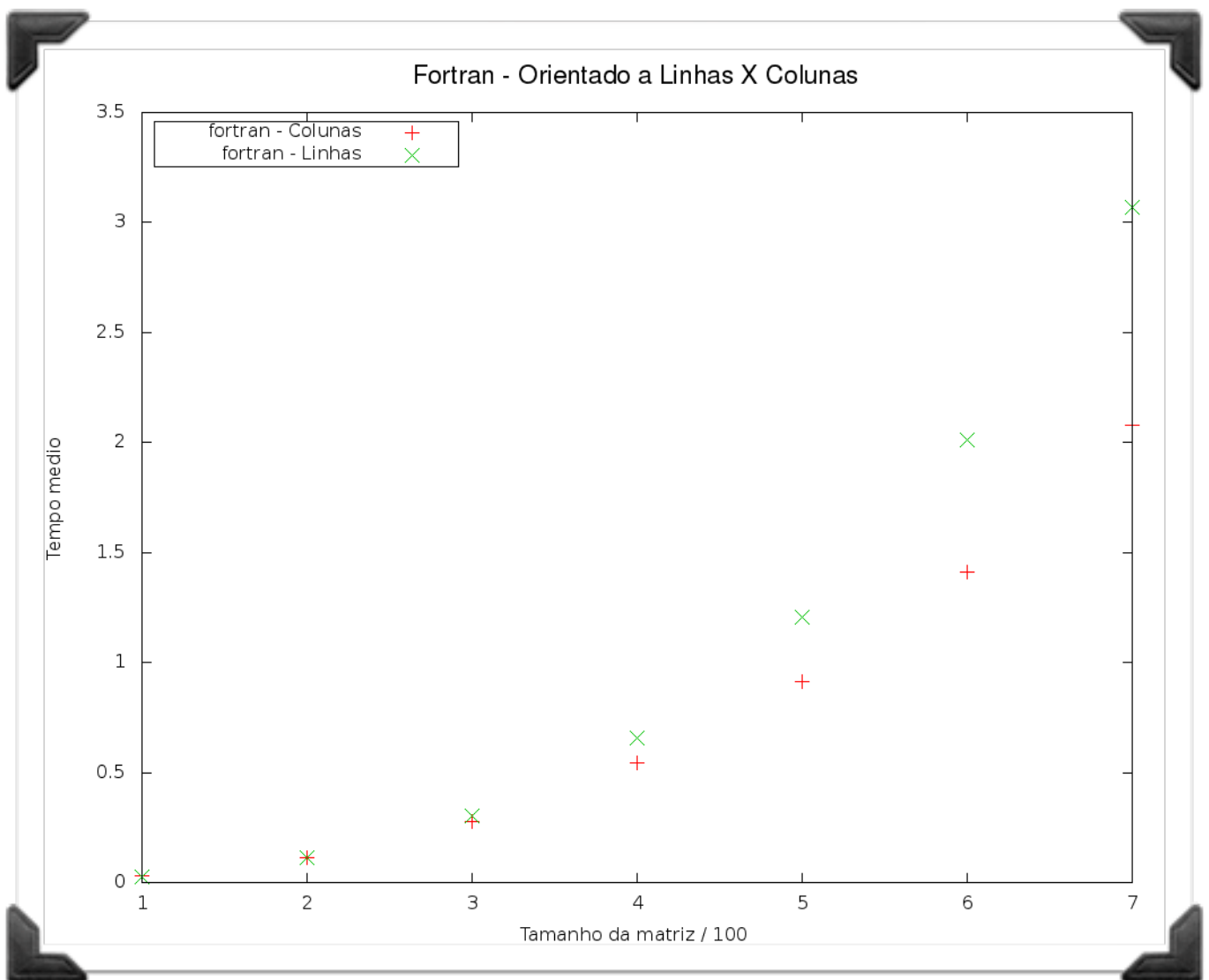


## C



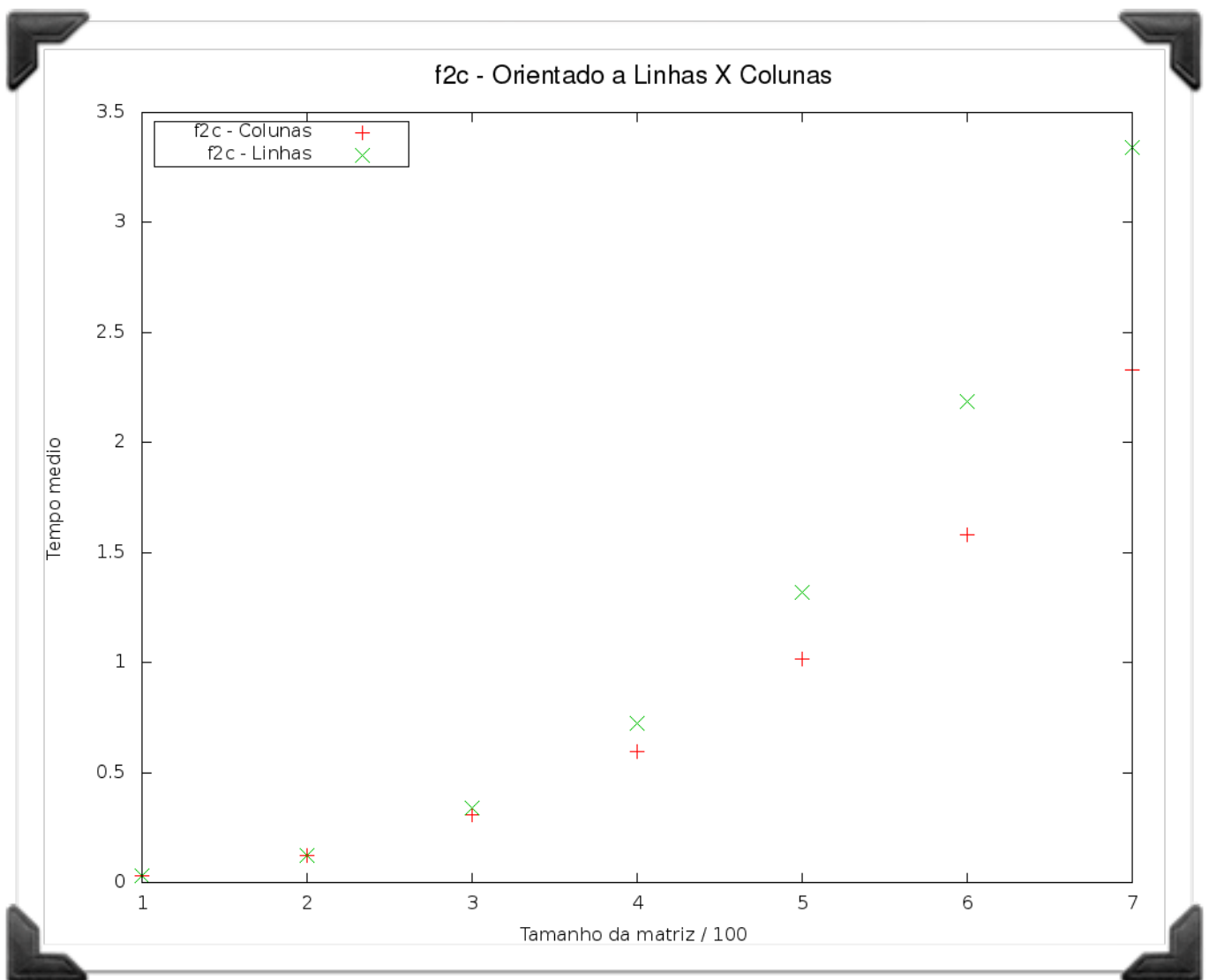
Como pode ser visto no gráfico acima, em C, o algoritmo é mais rápido orientado a linhas do que a colunas. Isso acontece porque C é uma linguagem orientada a linhas, isto é, um elemento  $a_{i,j+1}$  é alocado logo após  $a_{i,j}$  na memória. Portanto é menos custoso percorrer as linhas de uma matriz do que suas colunas.

# Fortran



Em Fortran, o algoritmo teve melhores resultados quando rodado por colunas. Isso aconteceu porque Fortran é uma linguagem orientada a colunas, isto é, um elemento  $a_{i+1, j}$  é alocado logo após  $a_{i, j}$  na memória. Portanto é menos custoso percorrer as colunas de uma matriz do que suas linhas.

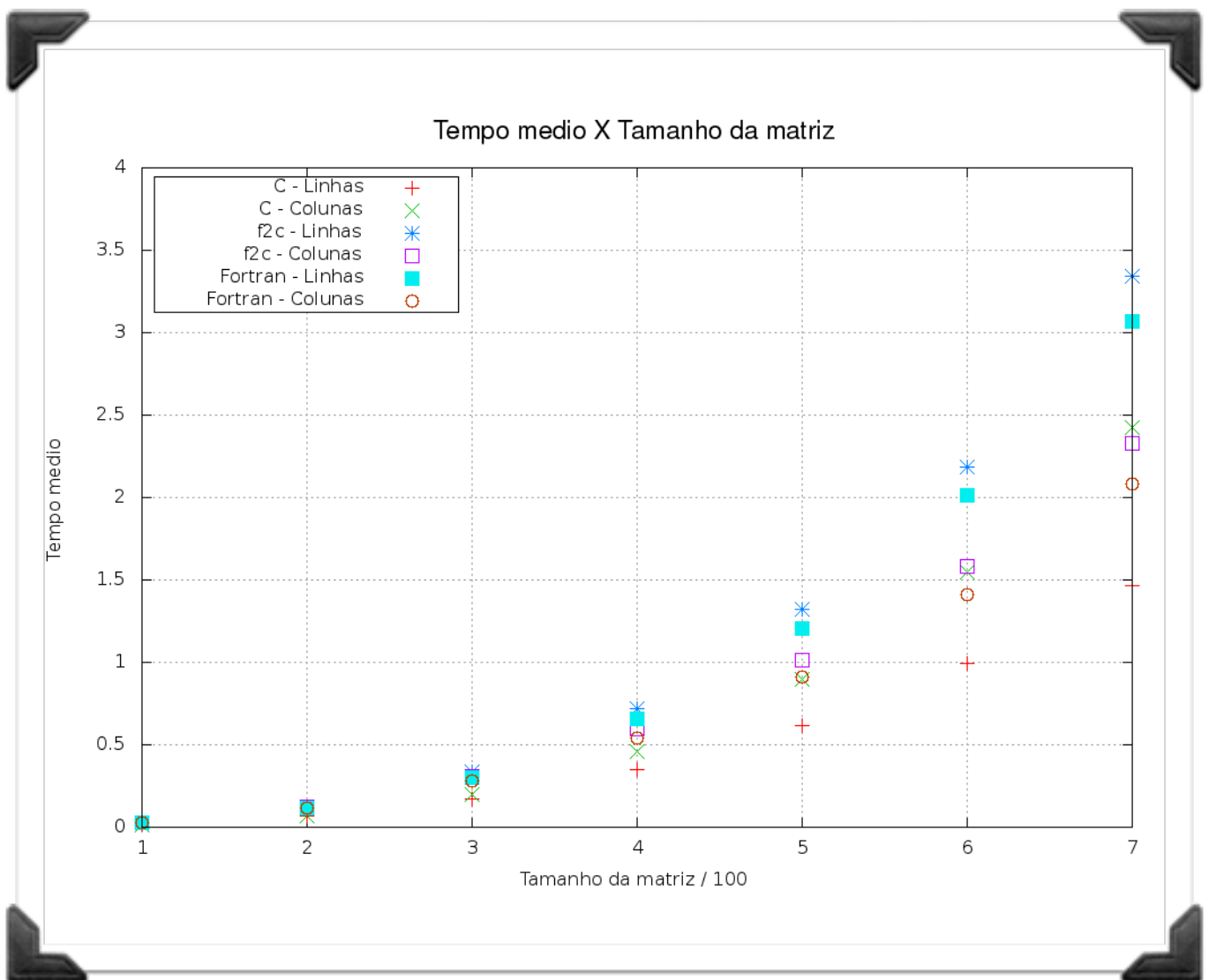
## f2c



Usando f2c para converter o mesmo código de fortran para C obtemos os resultados acima, nos quais é possível ver que o algoritmo foi mais rápido orientado a colunas, assim como foi em Fortran.

A explicação para esse resultado é que o f2c transforma uma matriz em fortran em um vetor no qual cada coluna da matriz original é colocada uma ao lado da outra. Dessa maneira, mantém-se o que ocorre quando acessamos índices da matriz em fortran, ou seja, um elemento  $a_{i+1,j}$  é alocado logo após  $a_{i,j}$  na memória. Portanto é menos custoso percorrer as colunas de uma matriz do que suas linhas.

## Conclusão



A partir do gráfico acima, é evidente que o mais rápido foi C orientado a linhas. surpreendentemente, C orientado a colunas compete bem com Fortran orientado a colunas, o que nos leva a concluir que nesse teste C superou Fortran em velocidade. f2c foi muito semelhante ao código em Fortran, sendo ótimos resultados para um código convertido. No entanto, não podemos utilizar f2c caso queiramos converter um código orientado a linhas em Fortran esperando que ele se comporte melhor em C.

No gráfico acima, os eixos estão escalados de tal forma que a distância entre 1 e 2 no eixo X é a mesma que no eixo Y. Portanto é visível o comportamento exponencial,  $O(n^3)$  do tempo médio em relação ao tamanho de uma matriz.