

# Projeto 4 de Física Computacional

Pedro Bicudo e Nuno Cardoso , IST,

16 de Dezembro de 2017

## Resumo das instruções

Este projeto deve ser submetido como um único ficheiro comprimido, na página pessoal de aluno no fénix:

- no menu superior devem seleccionar estudante
- no menu lateral devem seleccionar submeter, projetos;
- deve surgir o grupo desta disciplina de FC, e lá podem submeter projetos e visualizar os projetos submetidos.

Este ficheiro comprimido, por exemplo .zip, deve incluir todos os ficheiros desenvolvidos,

- usando as extensões .cpp e .h para os códigos de g++, e **.txt para os inputs**,
- .nb para os códigos de mathematica , ou gráficos em formato .pdf, com labels,
- a memória descritiva em formato .txt (ver parágrafo seguinte).

para diminuir o número de ficheiros, não é necessário enviar os ficheiros de dados gerados na execução (exceto os pedidos). Não devem ser incluídos os ficheiros executáveis.

É importante que enviem uma pequena memória descritiva, em texto simples, chamado `readme_g#p#.txt`, que inclua apenas

- o número e nome dos alunos do grupo
- o número do grupo
- todo e qualquer comentário, observação ou análise pedidos no enunciado,
- a lista de ficheiros submetidos, e para cada ficheiro algumas palavras a descrever o seu objectivo e indicando nos **.cpp** exatamente como compilar, executar, etc
- e finalmente, apenas caso hajam vários ficheiros a linkar em conjunto (por exemplo várias funções e um header), a linha de comandos para os linkar, por exemplo algo do tipo `g++ -o g09p2c1.o g09p2c1.cpp g09p2c2.cpp g09p2c3.cpp`, ou um Makefile.

Para sua própria organização os alunos/grupos devem agrupar os ficheiros por pastas de alíneas. Devem designar os códigos que forem construindo por `g#p3c#.ext` devendo substituir os `#` por números, sendo o primeiro `#` o número do seu grupo, o 3 refere-se ao projeto 3 e o segundo `#` o número do código (ex: `g32p3c2.cpp` será o segundo código que o grupo 32 realiza para este projeto. O próprio ficheiro zip deve ser denominado `g\#p3.zip`.

Serão avaliados,

- se o código compila seguindo o `readme_g#p#.txt` (senão a alínea é anulada),
- se o código executa seguindo o `readme_g#p#.txt` (senão a alínea é anulada),
- a ausência de erros ou de fugas de memória e a robustez, nos executáveis,
- a flexibilidade, simplicidade e eficiência computacionais do código,
- a indentação e os comentários que ajudem ao entendimento e uso do código,
- a correção do algoritmo e a fidelidade às expressões físicas,
- o valor dos resultados (os plots .pdf e os comentários no `readme_g#p#.txt`),
- a clareza dos resultados (de novo na forma de plots, labels e comentários),

Neste projeto, caso já tenha desenvolvido um código semelhante, pode criar **um único código** que produz todos os outputs desejados - referindo isso no `readme_g#p#.txt` e comentando muito bem no código que parte se refere a cada alínea - ou então pode criar **códigos diferentes em cada alínea** se quiser dividir o projeto por tarefas paralelas.

## 1 A eq. de Schrödinger na forma duma eq. matricial

### 1.a

(4pt) Pretendemos resolver a equação de Schrödinger uni-dimensional, e para tal escrevemos a energia cinética e a energia potencial na forma duma matriz.

Considere um potencial e uma energia cinética, por exemplo entre moléculas, da forma,

$$V(x) = \frac{k}{2} \left( \frac{a^4}{b^2} - \frac{a^4}{x^2 + b^2} \right), \quad \hat{K} = -\frac{\hbar^2}{2m} \frac{d^2}{dx^2}.$$

Escreva um código `g#p4c1.cpp` que constrói a matriz correspondente, lendo os valores dos respetivos parâmetros dum ficheiro `inputschrodinger.txt`, bem como lendo os limites do intervalo a discretizar  $]-\Lambda, \Lambda[$  e o número de pontos da discretização `N_max`. Vamos assumir que a função de onda  $\phi(x)$  se anula na fronteira do intervalo.

Crie um ficheiro de dados de output `outmatrix.txt` onde escreva a matriz explicitamente, no caso particular em que, num certo sistema de unidades,  $k = 1, a = 1, b = 1, \hbar = 1, m = 1$ . Considerem também a discretização  $\Lambda = 5, N_{max} = 9$ . Copie a matriz correspondente para o seu ficheiro `g\#p3.zip` pois esta matriz será avaliada.

Escreva a equação matricial que a função de onda  $\phi(x)$  deve verificar, e classifique de que tipo de equação matricial se trata.

Verifique explicitamente que a matriz é real e simétrica, com  $M_{ij} = M_{ji}$ . Estas matrizes pertencem à classe das matrizes ditas hermiticas, que são diagonalizáveis e que têm valores próprios reais.

## 2 Inverter grande matrix

Um objetivo deste projeto (*que pode desenvolver em paralelo com o da secção 3*) é inverter uma grande matriz  $M$  quadrada de dimensão  $n \otimes n$  e determinar o tempo gasto e a memória utilizada pelo seu PC para realizar esta operação.

### 2.a

(2pt) Desenvolva uma classe com base na disponível em `Matrix_v1.cpp` e `Matrix_v1.h` na página da disciplina nas **Avaliação e correção dos Projetos e mini-teste**, ou com base na classe que já desenvolveu nas aulas práticas e de laboratório, que inclua os vários overloads e métodos que facilitem as operações em geral mais úteis com matrizes, devendo pelo menos implementar o produto de matrizes e o output de matrizes.

### 2.b

(3pt) Desenvolva um novo método para inverter a matriz quadrada, que utilize o algoritmo de Gauss, para inverter a matriz. Pode utilizar como base o código que desenvolveu nas aulas de laboratório e práticas. Reveja o seu código para ter a máxima eficiência possível.

## 2.c

(1 pt) O objetivo agora é determinar o tempo que o seu código demora a inverter a matriz gerada em 1.a), com os mesmos parâmetros, mas agora para diferentes valores de  $N_{\max}$ . Use o programa `clock.cpp` disponibilizado nos **Códigos exemplo** na página da disciplina (ou outro código que já tenha utilizado) como exemplo a adaptar para determinar o tempo de inversão. Deve fechar todas as janelas possíveis no seu computador exceto as necessárias para o código correr. Aplique a  $N_{\max}=9$ ,  $N_{\max}=99$ ,  $N_{\max}=999$ . Comente quanto tempo demora cada uma das inversões em função de  $N_{\max}$ .

## 3 Vetores e valores próprios da Eq. de Schrödinger

Um outro objetivo deste projeto (*que pode desenvolver em paralelo com o da secção 2*) é determinar os menores vetores próprios duma grande matriz  $M$  quadrada de dimensão  $n \otimes n$ .

### 3.a

(3pt) Desenvolva um novo método para obter o vetor próprio de valor próprio mais alto de uma matriz quadrada e simétrica, que utilize o algoritmo de multiplicação pela matriz. Pode utilizar como base o código que desenvolveu nas aulas de laboratório e práticas. Mas aproveite para rever o seu código e verificar que tem a máxima eficiência possível.

### 3.b

(3pt) Desenvolva um novo método para obter os vetores próprios de valor próprio mais alto de uma matriz quadrada e simétrica, que utilize o algoritmo de multiplicação pela matriz, em conjugação com a ortogonalização de Gram-Schmidt ou com a diferença de matrizes. Pode utilizar como base o código que desenvolveu nas aulas de laboratório e práticas. Mas agora deve rever o seu código para verificar que tem a máxima eficiência possível. Deve ser capaz de, no mínimo, obter os 4 primeiros vetores próprios (1pt por cada vetor próprio para além do primeiro).

### 3.c

(3pt) Finalmente aplique os códigos até aqui desenvolvidos para,

- gerar a matriz  $M$  com os mesmos  $k$ ,  $a$ ,  $b$ ,  $\hbar$ ,  $m$  de 1.a) e com  $N_{\max} = 999$ ,  $\Lambda = 100.$ ,
- inverter a matriz,
- obter os primeiros, mais baixos, valores próprios  $\lambda_0, \lambda_1, \lambda_2 \dots$  da matriz  $M$  dos quais deve tomar nota na memória descritiva, e os seus respetivos vetores próprios  $\phi_0, \phi_1, \phi_2 \dots$ , que deve representar em plots no seu ficheiro `mathematica` ou num ficheiro `.pdf`.

### 3.d

(1pt) Diz-se que molécula fica ligada se a sua função  $\phi(x)$  ficar na região do poço de potencial (se quiser pode representar o potencial num plot). Se a função ocupar todo o intervalo  $]-\Lambda, \Lambda[$ , para um grande valor de  $\Lambda$  como o nosso, diz-se que a molécula fica livre (mesmo sendo perturbada pelo potencial).

Quais dos vetores próprios correspondem então a moléculas ligadas?