

---

# Boas Práticas de Programação e Escrita Científica

Tereza Mendes & Lucas Madeira

Instituto de Física de São Carlos – USP

---

# Perguntas

- O que mais importa em um projeto computacional?

---

# Perguntas

- O que mais importa em um projeto computacional?
- Qual a melhor linguagem de programação?

---

# Perguntas

- O que **mais importa** em um **projeto computacional**?
- Qual a melhor **linguagem de programação**?
- Quais as melhores **ferramentas**?

---

# Perguntas

- O que **mais importa** em um **projeto computacional**?
- Qual a melhor **linguagem de programação**?
- Quais as melhores **ferramentas**?
- Isso tudo influencia minha **trajetória acadêmica**?

---

# Perguntas

- O que **mais importa** em um **projeto computacional**?
- Qual a melhor **linguagem de programação**?
- Quais as melhores **ferramentas**?
- Isso tudo influencia minha **trajetória acadêmica**?

⇒ **SIM!!**



# Projeto Computacional vs. Vida Acadêmica

---



# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?



# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs),

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs), **compilador** (gnu gcc, gfortran),

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs), **compilador** (gnu gcc, gfortran), **gráficos** (gnuplot)

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs), **compilador** (gnu gcc, gfortran), **gráficos** (gnuplot)

⇒ CUDA (GPU), `fortran`, C, python, perl, shell, awk

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs), **compilador** (gnu gcc, gfortran), **gráficos** (gnuplot)

⇒ CUDA (GPU), `fortran`, C, python, perl, shell, awk

**Plataformas**: google drive, slack, Trello, DMPTool

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs), **compilador** (gnu gcc, gfortran), **gráficos** (gnuplot)

⇒ CUDA (GPU), `fortran`, C, python, perl, shell, awk

**Plataformas**: google drive, slack, Trello, DMPTool

**Referências**: Numerical Recipes, site do [Prof. J. Goodman](#), curso de **L<sup>A</sup>T<sub>E</sub>X** do Prof. Lucas

# Organização do Trabalho

---

Do que eu **preciso**? **sofisticação**, **simplicidade**, **controle**?

**Estrutura**: linux, dropbox, ferramentas de organização

**Manuseio** (processamento/análise/visualização): **editor** (**vi**, emacs), **compilador** (gnu gcc, gfortran), **gráficos** (gnuplot)

⇒ CUDA (GPU), `fortran`, C, python, perl, shell, awk

**Plataformas**: google drive, slack, Trello, DMPTool

**Referências**: Numerical Recipes, site do [Prof. J. Goodman](#), curso de **L<sup>A</sup>T<sub>E</sub>X** do Prof. Lucas

⇒ EM BREVE!! **Forum** no Portal da Graduação IFSC



# Aplicação: simulação computacional

---

- O que é a Simulação Computacional?

# Aplicação: simulação computacional

---

- O que é a Simulação Computacional?
  - ⇒ Experimento Virtual (Teórico!)

# Aplicação: simulação computacional

---

- O que é a Simulação Computacional?
  - ⇒ Experimento Virtual (Teórico!)
- Para que serve?

# Aplicação: simulação computacional

---

- O que é a Simulação Computacional?

  - ⇒ Experimento Virtual (Teórico!)

- Para que serve?

  - ⇒ Importância para pesquisa aplicada

# Aplicação: simulação computacional

---

- **O que é** a Simulação Computacional?

  - ⇒ Experimento Virtual (Teórico!)

- **Para que** serve?

  - ⇒ Importância para **pesquisa aplicada**

  - ⇒ **Modelagem** de sistemas

# Aplicação: simulação computacional

---

- O que é a Simulação Computacional?

  - ⇒ Experimento Virtual (Teórico!)

- Para que serve?

  - ⇒ Importância para pesquisa aplicada

  - ⇒ Modelagem de sistemas

  - ⇒ Crucial para o entendimento das interações entre quarks!

# Simulação Computacional

---

*A simulação é um processo de **projetar** um **modelo computacional de um sistema real** e **conduzir experimentos** com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação.*

D. Pegden (1990)

Usos

# Simulação Computacional

---

*A simulação é um processo de **projetar** um **modelo computacional de um sistema real** e **conduzir experimentos** com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação.*

D. Pegden (1990)

## Usos

- experimentos que não podemos/não queremos realizar (projeto de aviões, guerra nuclear, evolução)



# Simulação Computacional

---

*A simulação é um processo de **projetar** um **modelo computacional de um sistema real** e **conduzir experimentos** com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação.*

D. Pegden (1990)

## Usos

- experimentos que não podemos/não queremos realizar (**projeto de aviões**, **guerra nuclear**, **evolução**)
- reconstrução para melhor compreensão de eventos ocorridos (e.g. acidentes)

# Simulação Computacional

---

*A simulação é um processo de **projetar** um **modelo computacional de um sistema real** e **conduzir experimentos** com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação.*

D. Pegden (1990)

## Usos

- experimentos que não podemos/não queremos realizar (**projeto de aviões**, **guerra nuclear**, **evolução**)
- reconstrução para melhor compreensão de eventos ocorridos (e.g. acidentes)
- modelagem de sistemas (e.g. bactérias)

# Simulação Computacional

---

*A simulação é um processo de **projetar** um **modelo computacional de um sistema real** e **conduzir experimentos** com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação.*

D. Pegden (1990)

## Usos

- experimentos que não podemos/não queremos realizar (**projeto de aviões**, **guerra nuclear**, **evolução**)
- reconstrução para melhor compreensão de eventos ocorridos (e.g. acidentes)
- modelagem de sistemas (e.g. bactérias)
- estudo de problemas sem solução analítica (e.g. **QCD**)

# Evolução Temporal (Dinâmica)

---

É o que **caracteriza** uma simulação.

Sistema **evolui** no tempo, tem **vida própria!**

Realizamos **medidas**, analisamos **dados**

# Evolução Temporal (Dinâmica)

---

É o que **caracteriza** uma simulação.

Sistema **evolui** no tempo, tem **vida própria!**

Realizamos **medidas**, analisamos **dados**

Regra de evolução (dinâmica) pode ser **clássica** ou mesmo **quântica**. Sistema real, com dinâmica conhecida, em **condições fictícias** (exemplos: dinâmica molecular, Gedankenexperiment)

# Evolução Temporal (Dinâmica)

---

É o que **caracteriza** uma simulação.

Sistema **evolui** no tempo, tem **vida própria!**

Realizamos **medidas**, analisamos **dados**

Regra de evolução (dinâmica) pode ser **clássica** ou mesmo **quântica**. Sistema real, com dinâmica conhecida, em **condições fictícias** (exemplos: dinâmica molecular, Gedankenexperiment)

Casos em que não há formulação física: **Sistema fictício**  
⇒ modelagem (**in silico**)

# Evolução Temporal (Dinâmica)

---

É o que **caracteriza** uma simulação.

Sistema **evolui** no tempo, tem **vida própria!**

Realizamos **medidas**, analisamos **dados**

Regra de evolução (dinâmica) pode ser **clássica** ou mesmo **quântica**. Sistema real, com dinâmica conhecida, em **condições fictícias** (exemplos: dinâmica molecular, Gedankenexperiment)

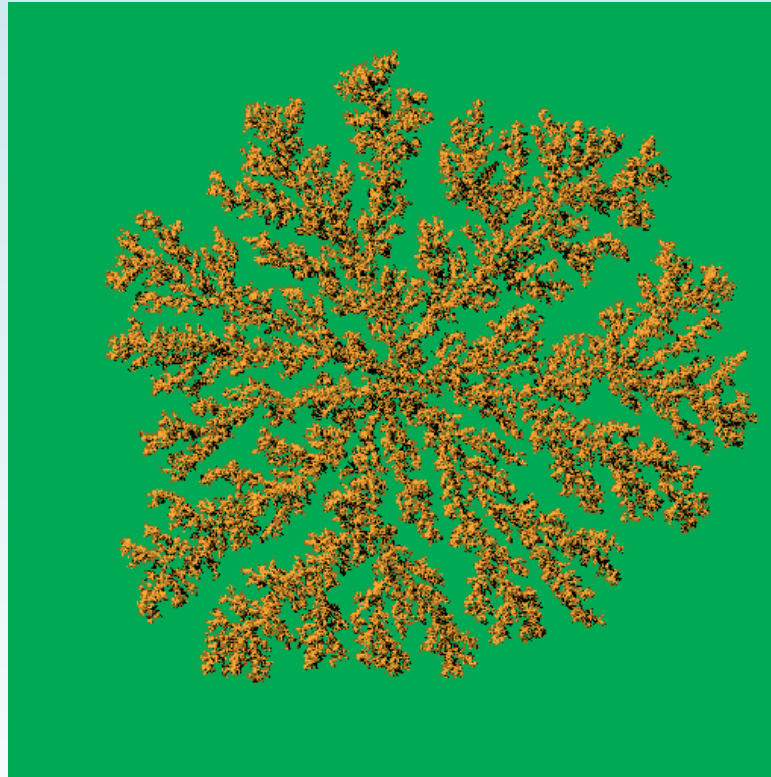
Casos em que não há formulação física: **Sistema fictício**  
⇒ modelagem (**in silico**)

**Ou:** sistema real, **dinâmica fictícia** (truque!)

# Autômato Celular

---

**Células** assumem valores finitos a cada instante de tempo.  
Regras **locais** de transição  $\Rightarrow$  comportamento **emergente**,  
solução numérica de equações diferenciais, geração de padrões  
visuais interessantes, e.g. **agregação limitada por difusão**





# Método de Monte Carlo

---

Sistemas estocásticos são simulados no computador usando um **gerador de números aleatórios**



⇒ tratamento teórico, com aspectos experimentais:

- dados, erros
- “medidas” no tempo



# Gerador de Números Aleatórios

---

*Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.*

John Von Neumann (1951)

**gerador** = **prescrição algébrica** que produz sequência de números  $r_i$  com distribuição desejada (em geral **uniforme** em  $[0,1]$ ) dada uma **semente**.

**Nota:** esta sequência é **determinística**, a operação repetida a partir do mesmo ponto inicial gera a mesma sequência  $\Rightarrow$  números **pseudo-aleatórios**.

# Exemplo: rand()

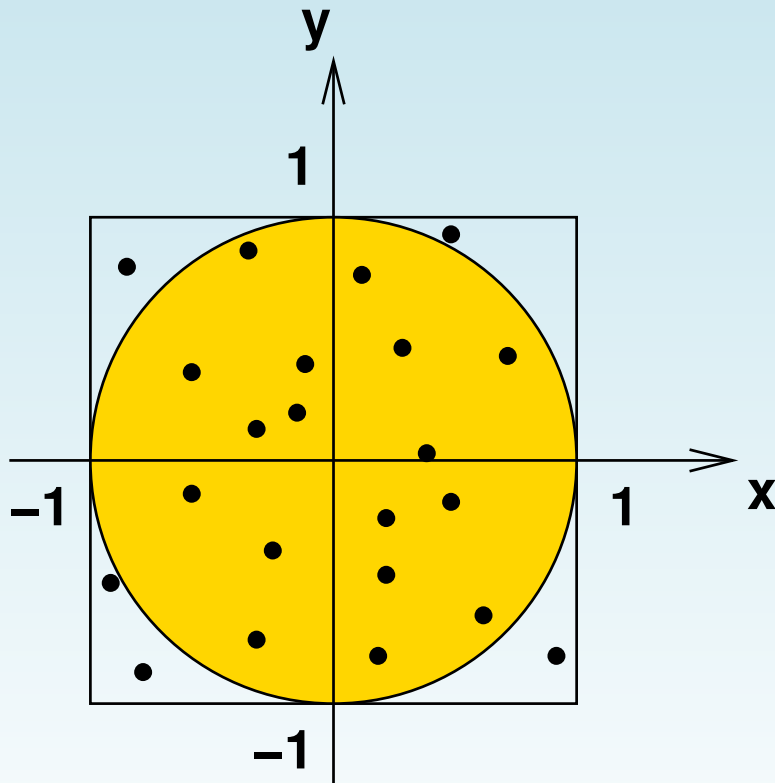
---

```
! inicializacao
iseed = 2011
call srand(iseed)
! numero aleatorio
r = rand()
```

- semente: número inteiro
- a cada passo um novo inteiro é produzido e usado como semente para o passo sucessivo
- inteiros são convertidos em reais em  $[0,1]$
- número aleatório em  $[a,b]$ :  $a + \text{rand}() * (b-a)$

# Área do Círculo de Raio 1

Lançando  $N$  pontos aleatórios uniformemente no quadrado:  $x \in [-1, 1]$ ,  $y \in [-1, 1]$



razão entre as áreas

$$\frac{A_{\circ}}{A_{\square}} = \frac{\pi}{4} = \frac{n}{N}$$

$n < N$  é o número de pontos no círculo

# Integral Unidimensional

---

**Integral** como soma de variáveis aleatórias igualmente distribuídas

$$I = \int_0^1 f(x) dx \rightarrow \frac{\sum_i f(x_i)}{N}$$

com  $x_i$  uniformemente distribuídos em  $[0,1]$ .

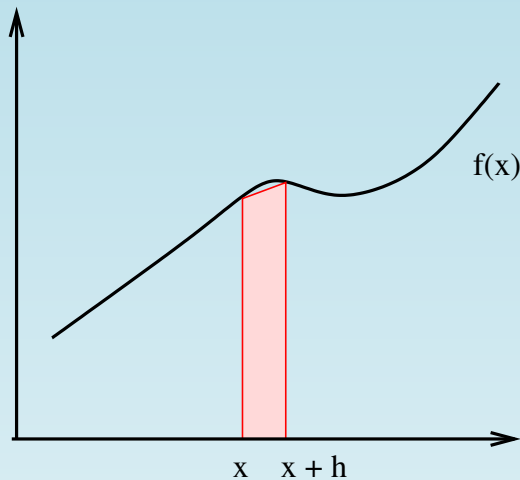
Na verdade, para  $N$  finito  $\bar{I} \equiv \frac{\sum_i f(x_i)}{N}$

é uma **variável aleatória**, que converge para seu valor médio  $I$  com erro proporcional a  $1/\sqrt{N}$  (teorema central do limite).

$$\sigma_{\bar{I}}^2 = \frac{\sigma_f^2}{N} = \frac{\langle f^2 \rangle - \langle f \rangle^2}{N}$$

**Exercício:** derive a expressão acima

# Métodos Determinísticos



**Regra do trapézóide:** estima a área compreendida entre  $x$  e  $x+h$  como a área do trapézóide definido pela aproximação linear da função entre estes dois pontos.

$$\int_x^{x+h} f(x') dx' = \frac{h}{2} [f(x) + f(x+h)] + \mathcal{O}(h^3 f'')$$

erro para integral em  $[a, b]$  é  $\mathcal{O}(h^2) \sim N^{-2}$

**Regra de Simpson:** aproximação de 3 pontos para  $f(x) \Rightarrow$   
erro é  $\sim N^{-4}$

# Comparação

---

- $d = 1$ :

métodos determinísticos tipicamente têm erros  $\mathcal{O}(N^{-2})$  (regra do trapézio) ou  $\sim \mathcal{O}(N^{-4})$  (regra de Simpson); **Monte Carlo** tem  $\mathcal{O}(N^{-1/2})$ : com  $2N$  pontos o erro diminui por um **fator 4** (trapezóide), **16** (Simpson) ou  $\sqrt{2}$  (Monte Carlo)

- $d > 1$ :

para integral  $d$ -dimensional  $N \sim 1/h^d \Rightarrow$  erro  $N^{-2/d}$  (trapezóide) ou  $N^{-4/d}$  (Simpson)  $\Rightarrow$  **Monte Carlo** começa a ser vantagem a partir de  $d = 8...$

# O Problema da Mecânica Estatística

---

**Mecânica Estatística:** probabilidade de uma configuração  $S$  para um sistema em equilíbrio à temperatura  $T$  é dada (no **ensemble canônico**) em termos de sua **hamiltoniana**  $\mathcal{H}(S)$  pela **distribuição de Boltzmann**

$$P(S) = \frac{e^{-\beta\mathcal{H}(S)}}{Z}; \quad Z = \int dS e^{-\beta\mathcal{H}(S)}; \quad \beta = 1/KT$$

Média termodinâmica do observável  $A$  dada por

$$\langle A \rangle = \int dS A(S) P(S)$$

e.g. energia:  $E = \langle \mathcal{H}(S) \rangle$

Integral (multi-dimensional) muito complicada!



# Estimativa

---

Tipicamente, em mecânica estatística o número de dimensões (i.e. número de graus de liberdade) é  $d \sim 10^3$  (e.g. modelo de Ising em 3d com 10 pontos por direção)

⇒ tempo para somar  $2^{1000}$  termos em computador de 1 Tflops:

# Estimativa

---

Tipicamente, em mecânica estatística o número de dimensões (i.e. número de graus de liberdade) é  $d \sim 10^3$  (e.g. modelo de Ising em 3d com 10 pontos por direção)

⇒ tempo para somar  $2^{1000}$  termos em computador de 1 Tflops:

$$t = 10^{288} s = 10^{270} \times \text{idade do universo}$$

# Estimativa

---

Tipicamente, em mecânica estatística o número de dimensões (i.e. número de graus de liberdade) é  $d \sim 10^3$  (e.g. modelo de Ising em 3d com 10 pontos por direção)

⇒ tempo para somar  $2^{1000}$  termos em computador de 1 Tflops:

$$t = 10^{288} s = 10^{270} \times \text{idade do universo}$$

⇒ Monte Carlo não é a melhor escolha,  
é a única escolha!

# Amostragem da Distribuição de Boltzmann

---

Mesmo com bons métodos e amostragem por importância, para uma distribuição conjunta de muitos graus de liberdade como a distribuição de Boltzmann

$$\langle A \rangle = \int A(x) w(x) dx, \quad w(x) = \frac{e^{-\beta \mathcal{H}(x)}}{Z}$$

não há esperança de amostragem direta!

**Solução:** Monte Carlo dinâmico. **Inventamos** uma evolução temporal de modo que as configurações geradas sejam distribuídas de acordo com  $w(x)$ . Isto pode ser feito para uma **dinâmica markoviana** escolhida de forma conveniente.

# Método de Metropolis

---

based on proposing and accepting/rejecting a step  $x \rightarrow y$

- **accept** if  $w(y)/w(x) \geq 1$
- otherwise **accept with probability**  $w(y)/w(x)$

the probability of acceptance is  $A_{xy} = \min \{1, w(y)/w(x)\}$ . Then consider the transition matrix  $p_{xy} = T_{xy} A_{xy}$  (with general  $T_{xy} = T_{yx}$ )

**Exercise:** show that the above choice satisfies **detailed balance**

For the Boltzmann distribution this means

$$w(x) = \frac{e^{-\beta E(x)}}{Z} \Rightarrow \frac{w(y)}{w(x)} = e^{-\beta \Delta E}; \quad \Delta E \equiv E(y) - E(x)$$

$\Rightarrow$  **accept** if  $\Delta E \leq 0$ ; otherwise **accept with probability**  $e^{-\beta \Delta E}$

**Note:** if proposed step is rejected, keep old value and move to a new site; when possible, choose  $T_{xy}$  such that acceptance is 50%

# Monte Carlo Method: Summary

---

Integral becomes sum of random variables

$$\int f(x) d\mu, \quad d\mu = \frac{e^{-\beta\mathcal{H}(x)}}{Z} dx \quad \Rightarrow \quad \frac{1}{N} \sum_{i=1}^N f(x_i)$$

where  $x_i$  have statistical distribution  $\mu$

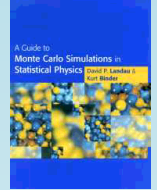
- **Static** Monte Carlo: independent sampling (error  $\sim 1/\sqrt{N}$ )
- **Dynamic** Monte Carlo: Simulation of a Markov chain with **equilibrium distribution**  $\mu$  (error  $\sim \sqrt{\tau/N}$ ). Autocorrelation time  $\tau$  related to **critical slowing-down**. **Note:** similar to **experimental methods**, **but** temporal dynamics was artificially introduced

**Errors:** either consider only effectively independent samples (via temporal correlation analysis) and error is given by standard deviation, **jack-knife**, **bootstrap** or consider all samples and error is estimated taking correlations into account: **binning** method, self-consistent windowing method

# Referências

---

- A Guide to Monte Carlo Simulations in Statistical Physics, [Landau & Binder](#) (Cambridge, 2000)



- Monte Carlo Methods in Statistical Physics, [Newman & Barkema](#) (Oxford, 1999)



- Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms, [Sokal](#) (1996)

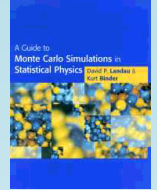
<https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.49.4444>

- Notas de aula: [Transição de fase e fenômenos críticos](#) (2002)

# Referências

---

- A Guide to Monte Carlo Simulations in Statistical Physics, [Landau & Binder](#) (Cambridge, 2000)



- Monte Carlo Methods in Statistical Physics, [Newman & Barkema](#) (Oxford, 1999)



- Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms, [Sokal](#) (1996)

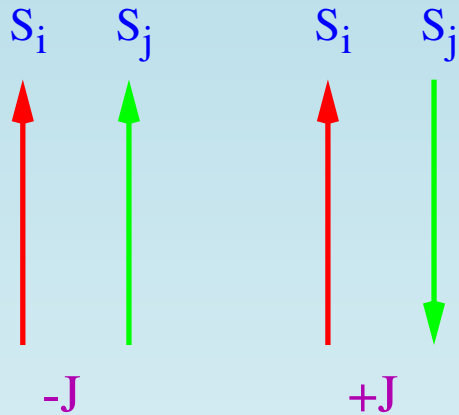
<https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.49.4444>

- Notas de aula: [Transição de fase e fenômenos críticos](#) (2002)

**HOJE:** [Tutorial \*\*SECRETA\*\* de Monte Carlo](#)



# Exemplo: o Modelo de Ising

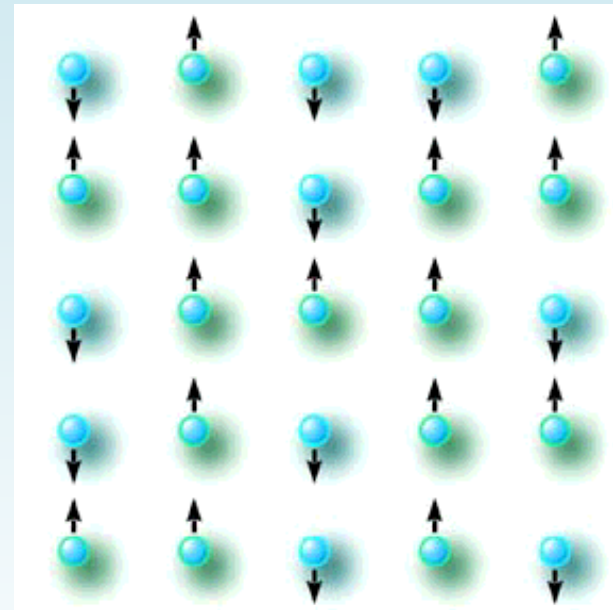


spins de dois estados, que preferem ficar alinhados

$$\mathcal{H}(S) = -J \sum_{\langle i,j \rangle} S_i S_j - H \sum_i S_i$$

## Observáveis de interesse

- Energy:  $E = \langle \mathcal{H}(S) \rangle$
- Specific Heat:  $C_V = \partial E / \partial T$
- Magnetization:  $M = \langle \sum_i S_i \rangle$
- Suscetibility:  $\chi = \partial M / \partial H$



# Aplicações do Método de Monte Carlo

---

- **Mecânica Estatística**: descrição de **sistemas de muitos corpos** ( $\approx 10^{23}$  corpos...) utilizando grandezas médias  $\Rightarrow$  comportamento **macroscópico** (termodinâmica) a partir da descrição **microscópica** de sistemas como fluidos/gases, modelos de **materiais magnéticos**, sistemas biológicos; tratamento de **fenômenos críticos**, sistemas complexos.
- **Matéria Condensada**: descrição aproximada de sistemas quânticos, polímeros, fluidos complexos, propriedades condutoras/magnéticas.

- **Cromodinâmica Quântica (QCD):** teoria quântica de campos que descreve a **força nuclear** como interação forte entre **quarks** e **glúons**; Formulação de **Rede**  $\Leftrightarrow$  Mecânica Estatística.

