

Modelagem e Avaliação de Desempenho

Simulação de Sistemas

Prof. Carlos Marcelo Pedroso

Pós Graduação em Engenharia Elétrica - PPGE

2025

- Simulação é a técnica de solução de um problema pela análise de um modelo que descreve o comportamento do sistema utilizando um computador digital.
- **Metodologia:**
 - ✓ Construção de um modelo da situação e reprodução computacionalmente.
 - ✓ Inclusão de alterações para o estudo de otimizações desejadas.

Simulação e a Variável Tempo

- Simulação é a técnica de resolver um problema analisando um modelo que descreve o comportamento de um sistema.
- A variável tempo é fundamental na maioria dos modelos de simulação, pois determina a evolução do sistema.
- Existem duas estratégias principais para implementar a variável tempo em uma simulação.

- A simulação avança de forma contínua, utilizando equações diferenciais ou integrais para descrever a evolução do sistema.
- A variável de tempo pode assumir qualquer valor dentro de um intervalo.
- É frequentemente usada para modelar sistemas físicos, como dinâmicas de fluidos, sistemas elétricos ou mecânicos.
- Requer métodos numéricos de integração para resolver as equações do modelo ao longo do tempo.

- O tempo avança em passos fixos ou em eventos.
- **Avanço por Incremento Fixo de Tempo:**
 - ✓ O relógio da simulação avança em intervalos de tempo pré-determinados (por exemplo, 1 segundo, 1 minuto).
 - ✓ A evolução do sistema é avaliada e atualizada a cada passo.
 - ✓ Pode ser ineficiente se houver longos períodos sem eventos importantes.
- **Avanço por Próximo Evento:**
 - ✓ O relógio da simulação avança diretamente para o momento em que o próximo evento ocorre.
 - ✓ É mais eficiente para sistemas onde os eventos são esporádicos ou a ocorrência deles é estocástica (aleatória), como em sistemas de filas.
 - ✓ Requer uma lista de eventos futuros ordenada para saber quando o próximo evento acontecerá.

Tempo Contínuo

- ▶ **Vantagens:** Alta precisão para sistemas com dinâmicas contínuas.
- ▶ **Desvantagens:** Complexidade computacional elevada.
- ▶ **Exemplos:** Simulação de circuitos, movimento de corpos celestes.

Tempo Discreto

- ▶ **Vantagens:** Mais eficiente para sistemas com eventos isolados.
- ▶ **Desvantagens:** Menos adequado para processos com variações contínuas.
- ▶ **Exemplos:**
 - ✓ Simulação de filas (chegadas, atendimentos).
 - ✓ Simulação de redes de computadores.

O Método de Monte Carlo

- O método de Monte Carlo originou-se de uma técnica matemática utilizada por cientistas do projeto Manhattan em Los Alamos na década de 1940, e foi publicada em 1949.
- Na aplicação desta técnica, os dados são gerados empregando-se um gerador de números aleatórios e uma distribuição de probabilidade que descreve a variável aleatória de interesse.
- O nome *Monte Carlo* foi cunhado por Nicholas Metropolis, em homenagem ao famoso cassino de Monte Carlo em Mônaco, por ser um local que evoca jogos de azar e aleatoriedade.
- O projeto buscava desenvolver a bomba atômica, e os cientistas precisavam resolver problemas complexos relacionados à simulação de nêutrons e ao seu transporte através de materiais.
- Baseado em **eventos discretos**.

Monte Carlo: A Ideia Central

- A ideia geral era utilizar a aleatoriedade para modelar e encontrar a solução para problemas determinísticos complexos que não podiam ser resolvidos por métodos analíticos.
- Em vez de resolver as equações diretamente, o método usava a simulação de inúmeros eventos aleatórios para obter uma aproximação estatística da solução.
- O princípio é que a probabilidade de um evento, que pode ser simulado aleatoriamente, se aproxima da sua frequência real quando o número de tentativas se torna muito grande.
- Um exemplo clássico é a estimativa do valor de π através da simulação do lançamento de pontos aleatórios em uma área conhecida.

O Método de Monte Carlo: Passos

- 1 Definir o domínio de entradas possíveis.
- 2 Gerar os eventos futuros de acordo com uma distribuição de probabilidade que descreve a entrada.
- 3 Realizar o processamento determinístico dos eventos, movendo a variável tempo para o tempo do próximo evento.
- 4 Agregar os resultados e retornar ao passo 2.

Geração de Variáveis Aleatórias: Método da Inversa

- Toma-se a distribuição acumulada da variável aleatória, dada por $P(X \leq x) = F(x)$.
- Atribui-se um valor randômico (R_i) entre 0 e 1 para $F(x)$.
- Calcula-se o valor de x .
- Desta forma, para cada valor randômico entre 0 e 1 será obtido um valor de X_i .
- **Exemplos:** distribuição exponencial, distribuição uniforme.
 - ✓ Anote os desenvolvimentos durante a aula.

Exemplos

- ❶ Cálculo do valor de π através de uma simulação utilizando a razão das áreas.
 - ✓ A implementação é bastante simples, envolve a geração de duas variáveis aleatórias e o cálculo da hipotenusa em um triângulo retângulo para determinar se o ponto gerado está dentro ou fora do círculo.
 - ✓ Anote o programa desenvolvido durante a aula.
- ❷ Simulação de uma fila M/M/1.
 - ✓ A particularidade é que conseguimos fazer esta implementação gerando apenas um evento futuro.
 - ✓ Desta forma, a implementação é bastante simples.
 - ✓ Anote o programa desenvolvido durante a aula.
- ❸ Simulação de uma fila M/M/c
 - ✓ Neste caso, a implementação de uma fila de eventos é necessária.
 - ✓ Este exemplo é mais completo do ponto de vista didático, já possui todos os elementos de um *motor* de simulação.
 - ✓ Anote o programa desenvolvido durante a aula.

- Um dos problemas a serem resolvidos é como gerar números randômicos, uniformemente distribuídos entre 0 e 1.
- **Gerador Congruente Linear ("LCG")**
 - ✓ Definido pela equação linear $x_{n+1} = (ax_n + b) \bmod m$.
 - ✓ Produz uma sequência entre $\{0, 1, \dots, m - 1\}$.
 - ✓ x_0 é a semente (valor inicial).
- **Exemplos de LCG:**
 - ✓ ANSI C $\rightarrow LCG(2^{31}, 1103515245, 12345, 12345)$.
 - ✓ Minimal Standard $\rightarrow LCG(2^{31}, 16807, 0, 1)$.

Exemplo Prático: LCG

- O Gerador Congruente Linear (LCG) é definido pela equação:

$$x_{n+1} = (ax_n + b) \mod m$$

- Para este exemplo, usaremos os seguintes parâmetros:

- ✓ $a=13$
- ✓ $b=41$
- ✓ $m=64$

- A semente inicial, escolhida de forma aleatória, é $x = 10$.

Valores Gerados

n	x_n
1	10
2	$(13 \cdot 10 + 41) \bmod 64 = 171 \bmod 64 = 43$
3	$(13 \cdot 43 + 41) \bmod 64 = (559 + 41) \bmod 64 = 600 \bmod 64 = 24$
4	$(13 \cdot 24 + 41) \bmod 64 = (312 + 41) \bmod 64 = 353 \bmod 64 = 33$
5	$(13 \cdot 33 + 41) \bmod 64 = (429 + 41) \bmod 64 = 470 \bmod 64 = 22$
6	$(13 \cdot 22 + 41) \bmod 64 = (286 + 41) \bmod 64 = 327 \bmod 64 = 7$
7	$(13 \cdot 7 + 41) \bmod 64 = (91 + 41) \bmod 64 = 132 \bmod 64 = 4$
8	$(13 \cdot 4 + 41) \bmod 64 = (52 + 41) \bmod 64 = 93 \bmod 64 = 29$
...	...

Geração de Números Randômicos: Método Tausworthe

- 1 **Escolha do Polinômio Recorrente:** Um polinômio primitivo de grau r é escolhido, definindo como os bits são gerados.
- 2 **Recorrência Linear:** Os bits são gerados recursivamente usando a fórmula $x_n = (x_{n-r} \oplus a_1 x_{n-r+1} \oplus \dots \oplus a_{r-1} x_{n-1})$.
- 3 **Agrupamento de Bits:** Os bits podem ser agrupados para formar números pseudoaleatórios, como números de ponto flutuante entre 0 e 1.
- 4 **Período da Sequência:** O período máximo é $2^r - 1$, e para alcançá-lo, o polinômio deve ser primitivo.

Exemplo Prático: Tausworthe

- Considere um exemplo com $r = 3$ e o polinômio primitivo $P(x) = x^3 + x + 1$.
- O método de Tausworthe utiliza uma recorrência linear definida por um polinômio primitivo. A forma geral do polinômio primitivo de grau r é:

$$P(x) = x^r + a_1x^{r-1} + \dots + a_{r-1}x + 1$$

- A recorrência linear é dada por:

$$x_n = (x_{n-r} \oplus a_1x_{n-r+1} \oplus a_2x_{n-r+2} \oplus \dots \oplus a_{r-1}x_{n-1})$$

- Para o polinômio primitivo fornecido, $P(x) = x^3 + x + 1$, o grau é $R = 3$. A forma geral para $r = 3$ é $P(x) = x^3 + a_1x^2 + a_2x + 1$
- Comparando o polinômio dado ($x^3 + x + 1$) com a forma geral ($x^3 + a_1x^2 + a_2x + 1$) é possível extrair os coeficientes $a_1 = 0$ e $a_2 = 1$.

Valores Gerados - Exemplo

- A fórmula para gerar os bits é:

$$x_n = x_{n-3} \oplus x_{n-1}$$

- A semente inicial são os bits $x_0 = 1$, $x_1 = 0$ e $x_2 = 1$.

n	x_n
0	1
1	0
2	1
3	$x_3 = x_0 \oplus x_2 = 1 \oplus 1 = 0$
4	$x_4 = x_1 \oplus x_3 = 0 \oplus 0 = 0$
5	$x_5 = x_2 \oplus x_4 = 1 \oplus 0 = 1$
6	$x_6 = x_3 \oplus x_5 = 0 \oplus 1 = 1$
7	$x_7 = x_4 \oplus x_6 = 0 \oplus 1 = 1$

- A sequência gerada é: 1, 0, 1, 0, 0, 1, 1,

Polinômios primitivos geralmente usados

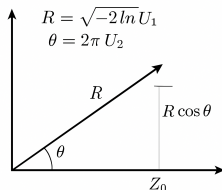
Grau (r)	Polinômio Primitivo
5	$x^5 + x^2 + 1$
7	$x^7 + x + 1$
9	$x^9 + x^4 + 1$
11	$x^{11} + x^2 + 1$
13	$x^{13} + x^4 + x^3 + x + 1$
17	$x^{17} + x^3 + 1$
23	$x^{23} + x^5 + 1$
31	$x^{31} + x^3 + 1$
35	$x^{35} + x^2 + 1$
63	$x^{63} + x^2 + 1$

Demais Métodos para Geração de Variáveis Aleatórias Uniformes

- ▶ Além dos Geradores Congruentes Lineares (LCG) e dos Geradores baseados em Registro de Deslocamento com Feedback (como o método Tausworthe), existem outros métodos conhecidos para gerar variáveis aleatórias uniformemente distribuídas:
 - ✓ **Mersenne Twister**: muito popular e robusto
 - ✓ Possui um período extremamente longo ($2^{19937} - 1$) e alta uniformidade, sendo ideal para aplicações que exigem alta qualidade de aleatoriedade, como simulações complexas.
 - ✓ **Geradores Criptográficos**: Projetados para serem imprevisíveis e seguros.
 - ✓ Exemplos incluem algoritmos baseados em hash (como SHA-1) ou em criptografia de fluxo, mas são mais lentos que os geradores não criptográficos, mas essenciais para aplicações de segurança.
 - ✓ **Geradores de Hardware**: Geram aleatoriedade a partir de fenômenos físicos imprevisíveis, como ruído térmico ou decaimento radioativo.

Método de Box-Muller (do Seno/Cosseno) para Distribuição Normal

- Objetivo: gerar números pseudoaleatórios com distribuição normal (média 0 e variância 1) a partir de números uniformemente distribuídos.
- A distribuição normal não tem equação conhecida para distribuição acumulada.
- Dado dois números uniformes independentes $U_1, U_2 \sim U(0, 1)$, transformá-los em dois números normalmente distribuídos $Z_0, Z_1 \sim N(0, 1)$.



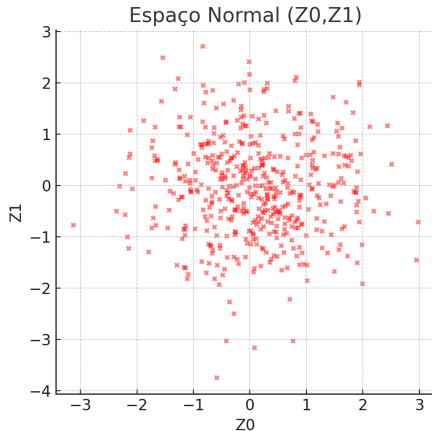
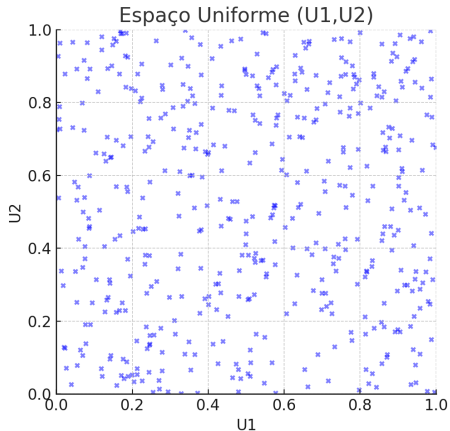
✓ $R = \sqrt{-2 \ln U_1}$

✓ $\theta = 2\pi U_2$

✓ $Z_0 = R \cos \theta$ e $Z_1 = R \sin \theta$

- Z_0 e Z_1 são independentes e seguem $N(0, 1)$.
- Para média μ e desvio σ : $X = \mu + \sigma Z$.

Método de Box-Muller (do Seno/Cosseno) para Distribuição Normal



Aproximação Polinomial para Distribuição Normal

- Este método utiliza uma aproximação polinomial para gerar uma variável aleatória com distribuição normal $N(0, 1)$.
- A aproximação para a variável x_i é dada pela seguinte equação, onde R_i é uma variável uniformemente distribuída entre 0 e 1:

$$x_i = \frac{R_i^{0.135} - (1 - R_i)^{0.135}}{0.1975}$$

- Para obter uma variável com distribuição normal com média μ e desvio padrão σ , basta aplicar a transformação:

$$y_i = \mu + \sigma x_i$$

- Este método é uma alternativa simples e eficiente a outros algoritmos, como o de Box-Muller.

Outros Métodos para a Geração de VA com Distribuição Normal

- Existem diversos métodos para gerar variáveis aleatórias com distribuição normal, além do método de Box-Muller e da Aproximação Polinomial:
 - ✓ **Método Polar de Marsaglia:** É uma alternativa ao método de Box-Muller, muitas vezes considerada mais eficiente computacionalmente, pois evita o uso de funções trigonométricas. Utiliza coordenadas polares e o método de aceitação-rejeição para gerar pares de variáveis normais.
 - ✓ **Algoritmo Ziggurat:** Um dos algoritmos mais rápidos para a geração de variáveis normais. Baseado na divisão da área sob a curva de densidade de probabilidade em retângulos e uma pequena área restante.
 - ✓ **Método de Aceitação-Rejeição:** Técnica geral que pode ser aplicada à distribuição normal. Envolve amostrar de uma distribuição mais simples (distribuição-envolvente) e aceitar ou rejeitar a amostra com base em uma condição.

- A análise de resultados é a etapa mais crítica de uma simulação. Conclusões válidas só podem ser extraídas com o devido rigor estatístico.
- **Atenção à Aleatoriedade:**
 - ✓ Não generalize resultados de uma única rodada de simulação.
 - ✓ A natureza estocástica da simulação exige a execução de múltiplas réplicas, cada uma com uma semente de números aleatórios diferente.
- **Necessidade de Análise Estatística:**
 - ✓ Use métodos estatísticos para quantificar a confiabilidade das estimativas.
 - ✓ Calcule o semi-intervalo de confiança (h) para determinar a precisão das medidas de desempenho (por exemplo, tempo médio na fila, tempo médio no sistema).
 - ✓ O intervalo de confiança oferece uma probabilidade de que a verdadeira média do sistema esteja contida no intervalo $[\bar{X} - h, \bar{X} + h]$.

➤ Melhorando a Precisão:

✓ Para reduzir o intervalo de confiança e aumentar a precisão:

- 1 Aumentar o número de rodadas de simulação (n).
- 2 Reduzir a variância dos resultados, se possível.

- Quando uma simulação começa, o sistema está em um estado inicial, o que não reflete o seu comportamento em regime estacionário (steady-state).
- O período de warm-up é o tempo de simulação inicial necessário para que o sistema atinja o regime estacionário, ou seja, um estado representativo de sua operação normal.
- Durante este período, os dados coletados são influenciados pelas condições iniciais e não devem ser incluídos na análise de resultados.

Por que o Warm-up é Importante?

- **Evita Vício nos Resultados:** Se a simulação começar vazia, as primeiras observações (por exemplo, tempo de espera na fila) serão atipicamente baixas, viciando a média dos resultados finais.
- **Garante Representatividade:** A análise só deve começar após o sistema se estabilizar, garantindo que os dados coletados reflitam o comportamento real do sistema sob condições de operação normal.

Como Determinar o Período de Warm-up

- Não existe uma regra única. O período de warm-up deve ser determinado empiricamente.
- Método Gráfico:
 - ✓ Realize uma única simulação longa.
 - ✓ Plote a média acumulada da variável de interesse (por exemplo, tempo médio na fila) ao longo do tempo.
 - ✓ O ponto em que a curva se estabiliza indica o fim do período de warm-up.
- Rejeição dos Dados Iniciais:
 - ✓ Execute a simulação, descarte os dados coletados durante o período de warm-up e inicie a coleta para análise apenas após este ponto.

Verificação e Validação da Simulação

- A confiabilidade de uma simulação depende de dois processos críticos: Verificação e Validação.
- A verificação garante que o modelo de simulação foi implementado corretamente, ou seja, que o modelo computacional reflete com precisão o modelo conceitual.
- A validação garante que o modelo conceitual representa com precisão o sistema real que está sendo simulado.

Métodos de Verificação da Simulação

- A verificação foca em responder à pergunta: *a implementação da simulação funciona como o planejado?*.
- Simulação de Teste:
 - ✓ Realizar simulações com dados de entrada conhecidos e comparar os resultados com os valores esperados.
- Depuração e Rastreamento:
 - ✓ Analisar o código passo a passo para garantir que as operações e o fluxo de controle estão corretos.
- Revisão por Pares:
 - ✓ Outros programadores revisam o código para identificar erros de lógica ou de implementação.

Métodos de Validação da Implementação da Simulação

- Validação de Dados de Entrada: Garantir que os dados de entrada utilizados na simulação são representativos do sistema real.
- Validação de Comportamento: Comparar os resultados da simulação com o sistema real em condições idênticas ou semelhantes.
- Validação da Implementação da Simulação:
 - ✓ Comparar as métricas de desempenho da simulação (ex, tempo médio na fila) com os dados de desempenho observados no sistema real.
 - ✓ Se possível, comparar os resultados da simulação com soluções de modelos analíticos - se os resultados concordam para um cenário simples, isso aumenta a confiança de que a implementação está correto.
 - ✓ Validação de Condições Extremas:
 - ✓ Ex: Cenário de Baixa Carga: Simule o sistema com um tráfego muito baixo para ver se o tempo médio de espera tende a zero.
 - ✓ Ex: Cenário Não Estacionário: Altere drasticamente a taxa de chegada ou de serviço para verificar se o tempo de espera explode se a taxa de chegada for maior que a de serviço.

► Validação com Dados Históricos:

- ✓ Use dados históricos reais do sistema para alimentar o modelo de simulação.
- ✓ Compare as métricas de desempenho da simulação (ex: tempo de espera médio, utilização de recursos) com os dados históricos observados.
- ✓ Validação por Especialistas:
 - ✓ Apresente o modelo e os resultados da simulação a especialistas no domínio do sistema real.
 - ✓ O conhecimento e a experiência desses especialistas podem identificar comportamentos ou resultados que parecem implausíveis ou incorretos.

► Confiança Estatística:

- ✓ Um **intervalo de confiança** compreende um intervalo numérico que possui uma probabilidade igual a $(1 - \alpha)$ de incluir o verdadeiro valor da medida de desempenho sob análise.
- ✓ $(1 - \alpha)$ representa o intervalo de confiança.
- ✓ α representa o erro admitido.
- ✓ O **semi-intervalo** h é calculado por: $h = t_{(n-1, 1-\alpha/2)} \frac{\sigma}{\sqrt{n}}$.
 - ✓ n : número de rodadas.
 - ✓ σ : desvio padrão.
 - ✓ t : valores críticos para distribuição t-student.

- Suponha que foi simulado o tempo médio na fila em um sistema.
- Assumindo que a variável aleatória X representa o tempo médio na fila.
- A simulação foi realizada 5 vezes, tomando-se o cuidado de iniciar a simulação com valores de sementes diferentes.

- Os resultados obtidos foram:

Rodada	X
1	63,2
2	69,7
3	67,3
4	64,8
5	72

- O semi-intervalo h é calculado por:

$$h = t_{n-1, 1-\alpha/2} \frac{\sigma}{\sqrt{n}}$$

- n é o número de rodadas.
- σ é o desvio padrão.
- t indica os valores críticos para a distribuição t-student.

Valores críticos t-student

v	0,995	0,99	0,975	0,95	0,90
1	63,66	31,82	12,71	6,31	3,08
2	9,92	6,96	4,30	2,92	1,89
3	5,84	4,54	3,18	2,35	1,64
4	4,60	3,75	2,78	2,13	1,53
5	4,03	3,36	2,57	2,02	1,48
6	3,71	3,14	2,45	1,94	1,44
7	3,50	3,00	2,36	1,90	1,42
8	3,36	2,90	2,31	1,86	1,40
9	3,25	2,82	2,26	1,83	1,38
10	3,17	2,76	2,23	1,81	1,37

- No caso anterior, a média calculada é 67.22 e o desvio padrão é igual a 3.84.
- Para 95% de confiança, $\alpha = 0.05$, $t_{4,0.975} = 2.78$.
- O valor de h calculado é de 4,77.
- Com 95% de confiança a verdadeira média estará entre 62.44 e 71.99.

Exemplo

- Considerando uma simulação que estima o valor de π .
- Considerando uma simulação para uma fila M/M/1.
- Para ambos os casos:
 - ✓ Valide a implementação utilizando modelos conhecidos.
 - ✓ Valide a implementação utilizando situações extremas.
 - ✓ Calcule o semi intervalo h para um nível de confiança de 99%.
 - ✓ O que fazer para melhorar a resposta? (Melhorar a resposta implica em reduzir ao mínimo o valor de h).

- Suponha novamente o sistema com uma fila. No entanto, desta vez, suponha que a chegada é modelada por uma distribuição exponencial com média 4 e o atendimento é modelado também por uma distribuição de Pareto com parâmetros $\alpha = 2,5$ e $\beta = 2$.
- A distribuição de Pareto é uma distribuição de cauda pesada.
- Determine o tempo médio de fila e tempo médio no sistema.
 - ✓ Valide a implementação utilizando modelos conhecidos (é possível).
 - ✓ Valide a implementação utilizando situações extremas.
 - ✓ Determine o tempo médio de fila e tempo médio no sistema.
 - ✓ Realize a simulação de forma a obter uma boa resposta para o nível de confiança de 99%.
 - ✓ Interprete os resultados.

Distribuição de Pareto

- Parâmetros: $\alpha > 0$, parâmetro de forma, $\beta > 0$, parâmetro de escala.

Limites	$b \leq x < +\infty$
Densidade de Probabilidade	$f(x) = \frac{\alpha\beta^\alpha}{x^{\alpha+1}}$
Distribuição Acumulada	$F(x) = 1 - \left(\frac{\beta}{x}\right)^\alpha$
Esperança ($E[X]$)	$\frac{\alpha\beta}{\alpha-1}, \alpha > 1$
Variância ($Var[X]$)	$\frac{\alpha\beta^2}{(\alpha-1)^2(\alpha-2)}, \alpha > 2$

- Simular a dinâmica de uma epidemia usando o modelo SIR (Suscetíveis–Infectados–Recuperados).
- A simulação é um modelo de eventos discretos.
- O objetivo é observar e analisar a evolução da população em cada estado ao longo do tempo.
- β representa a **taxa de infecção**, é a probabilidade de um indivíduo suscetível se tornar infectado ao entrar em contato com um indivíduo infectado.
- γ representa a **taxa de recuperação**, é a probabilidade de um indivíduo infectado se recuperar e se tornar imune (passando para a categoria de Recuperados). O valor de γ é o inverso do tempo médio de infecção

- **População Total (N):** 1000 pessoas (fixa).
- **Estado Inicial:**
 - ✓ Suscetíveis: $S_0 = 999$
 - ✓ Infectados: $I_0 = 1$
 - ✓ Recuperados: $R_0 = 0$
- **Taxa de Infecção (β):** 0.2
- **Taxa de Recuperação (γ):** 0.1

Regras de Transição

- ▶ A simulação avança em passos de tempo (por exemplo, dias).
- ▶ A cada passo, ocorrem as seguintes transições (modeladas por variáveis de Poisson):
- ▶ **Suscetíveis $\rightarrow$ Infectados ($S \rightarrow I$):**
 - ✓ O número de novas infecções em um passo é

$$\text{NovasInfecções} \sim \text{Poisson}(\lambda_{\text{inf}}),$$

com

$$\lambda_{\text{inf}} = \beta \frac{S I}{N},$$

onde S e I são os valores atuais nesse passo de tempo.

- ▶ **Infectados $\rightarrow$ Recuperados ($I \rightarrow R$):**
 - ✓ O número de novas recuperações em um passo é

$$\text{NovasRecuperações} \sim \text{Poisson}(\lambda_{\text{rec}}),$$

com

$$\lambda_{\text{rec}} = \gamma I.$$

Valores Típicos de β e γ

- Os valores de β (taxa de infecção) e γ (taxa de recuperação) dependem da doença e do contexto.
- São cruciais para determinar a taxa de reprodução básica (R_0), que define a propagação da epidemia.

Doença	Taxa de Infecção (β)	Taxa de Recuperação (γ)
Gripe Comum	0.2 – 0.5	0.14 – 0.2
Sarampo	1.5 – 3.0	0.25
COVID-19	0.5 – 0.7	0.1

➤ 1. Implementação:

- ✓ Desenvolva um programa para simular a evolução de S , I e R por 100 dias, para os casos típicos de Gripe Comum, Sarampo e COVID-19.

➤ 2. Análise de Resultados:

- ✓ Execute a simulação 10 vezes com sementes diferentes.
- ✓ Plote os gráficos de S , I e R para cada rodada.
- ✓ Calcule a média do pico de infectados e o dia em que ele ocorre.
- ✓ Calcule o intervalo de confiança (semi-intervalo) para a média do pico de infectados (95% de confiança).

➤ **Observação:** No contexto da pandemia de COVID-19, o comportamento da infecção e da recuperação foi mais complexo do que o que a simples distribuição de Poisson consegue modelar. Embora o modelo SIR (e suas variantes) tenha sido amplamente utilizado para fins de previsão e análise, o comportamento real apresentou nuances que exigem modelos mais sofisticados.