

# Introdução à Simulação

---

Avaliação de  
Desempenho de  
Sistemas

# O que é “Desempenho”?

---

- Webster's?

- *The manner in which a mechanism performs.*

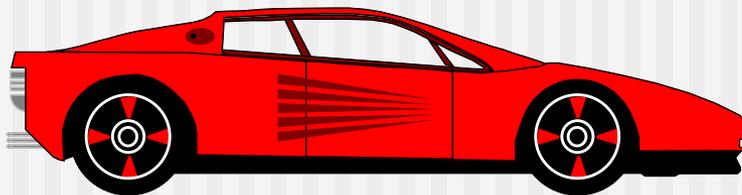
- Aurélio:

- **Mil.** Conjunto de características ou de possibilidades de atuação de **uma aeronave**, tais como velocidade de cruzeiro, velocidade de pouso, capacidade de carga, autonomia de vôo, etc.

# Medidas de desempenho de um automóvel

---

- Velocidade máxima
- Aceleração (tempo para ir de 0 a 100 km/h)
- Espaço de frenagem a uma dada velocidade



# Medidas de desempenho de sistemas computacionais

---

- **Vazão/Taxa (*Throughput*)**
  - Taxa na qual os pedidos são atendidos (servidos) pelo sistema.
- **Utilização:**
  - Fração do tempo em que o recurso permanece ocupado atendendo os pedidos dos usuários.
- **Tempo de resposta:**
  - tempo decorrido entre o pedido e o início/conclusão da realização do serviço.

# Medidas de desempenho

---

- Velocidade
  - tempo de resposta, vazão e utilização
- Confiabilidade
  - Probabilidade de erro
  - Intervalo entre erros
- Disponibilidade
  - Duração da falha
  - Intervalo entre falhas

# Técnicas de Avaliação

---

- Medição
- Modelagem Analítica
- Simulação

# Medição

---

- Para efetuarmos medições (como as Benchmarks) é preciso termos à disposição ao menos um protótipo do sistema.
- Normalmente é difícil comparar alternativas.

# Modelagem Analítica

---

- Teoria das filas
- Filas associadas a recursos
- Caracterização:
  - Processo de chegada
  - Processo de atendimento
  - Número de servidores
  - Tamanho máximo da fila
  - Política de atendimento da fila

# Modelagem Analítica

---

- É uma técnica aproximada
- Aproxima a realidade por um modelo
- Se o modelo for simples e a aproximação boa, é possível avaliar facilmente compromissos entre alternativas

# Simulação

---

- Simulação de eventos discretos
- Cada evento (ex.: chegada de usuário, término de serviço, etc.) é tratado quando do instante de sua ocorrência.
- Simula o comportamento de um sistema real
- Em geral, é possível construir um modelo muito mais próximo da realidade do que com a teoria das filas

# Critérios para seleção da técnica de avaliação

<b>Critério</b>	<b>Modelagem analítica</b>	<b>Simulação</b>	<b>Medição</b>
<b>Estágio</b>	Qualquer	Qualquer	Protótipo
<b>Tempo necessário</b>	Pouco	Médio	Variado
<b>Ferramentas</b>	Analistas	Linguagens de Programação	Instrumentação
<b>Precisão</b>	Pouca	Moderada	Variada
<b>Avaliação de Compromissos</b>	Fácil	Moderada	Difícil
<b>Custo</b>	Baixo	Médio	Alto
<b><i>Saleability</i></b>	Baixa	Média	Alta

# Simulação

---

- Técnica útil para a análise de desempenho de sistemas computacionais.
- Especialmente:
  - se o sistema não estiver disponível;
  - para prever o desempenho de diversas alternativas;
  - facilidade de efetuar comparações para uma maior variedade de cargas e de ambientes.

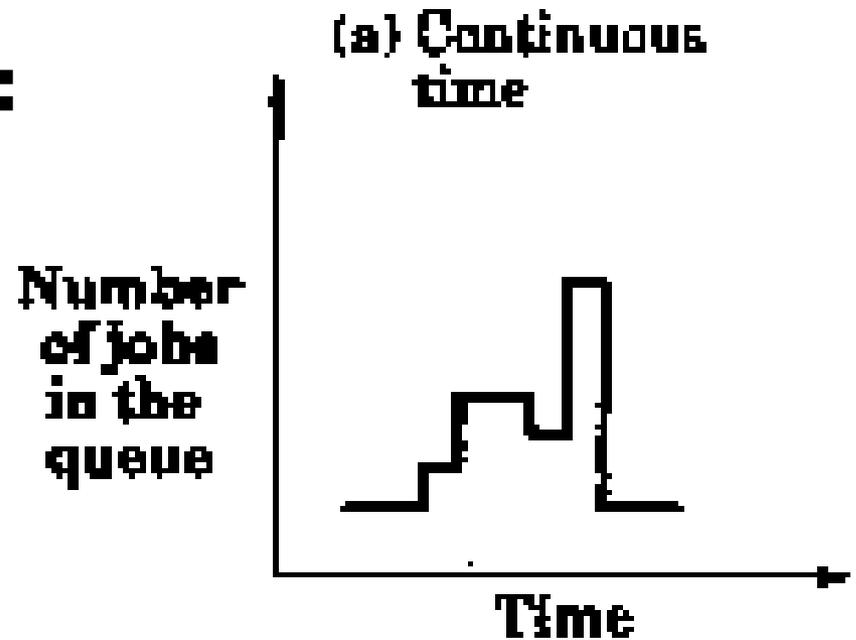
# Terminologia

---

- **Variáveis de estado:** definem o estado do sistema.  
A simulação pode ser continuada a partir do conhecimento das variáveis de estado.  
Exemplo: comprimento da fila de jobs.
- **Evento:** mudança no estado do sistema.  
Exemplos:
  - chegada de um job;
  - início de uma nova execução;
  - partida do job.

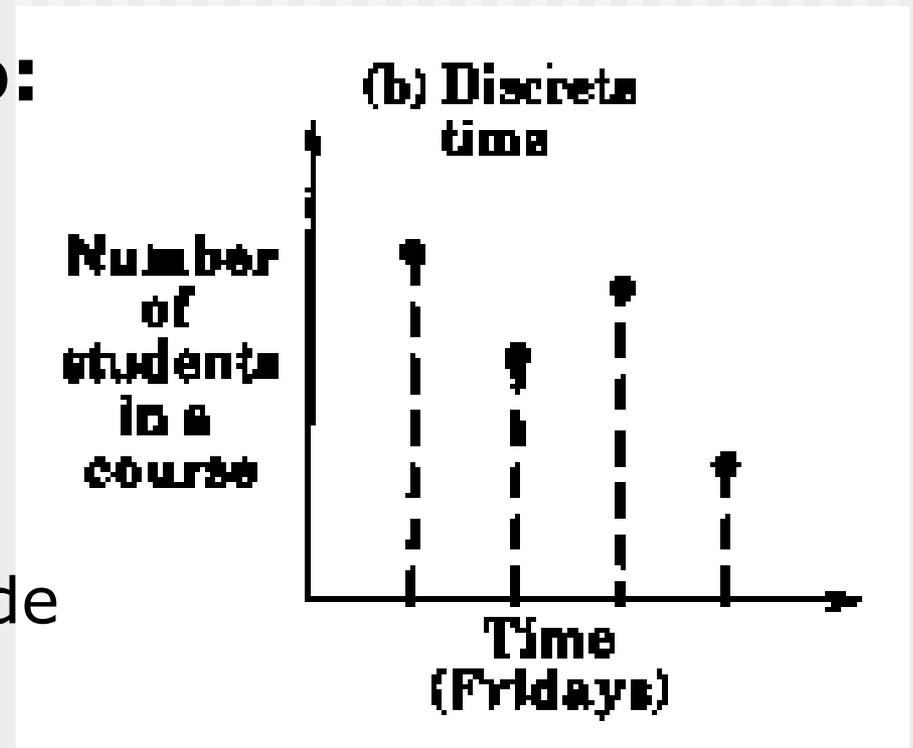
# Terminologia

- **Modelo de Tempo Contínuo:** o estado do sistema está definido em todos os instantes. Exemplo: Modelo de escalonamento de CPU.



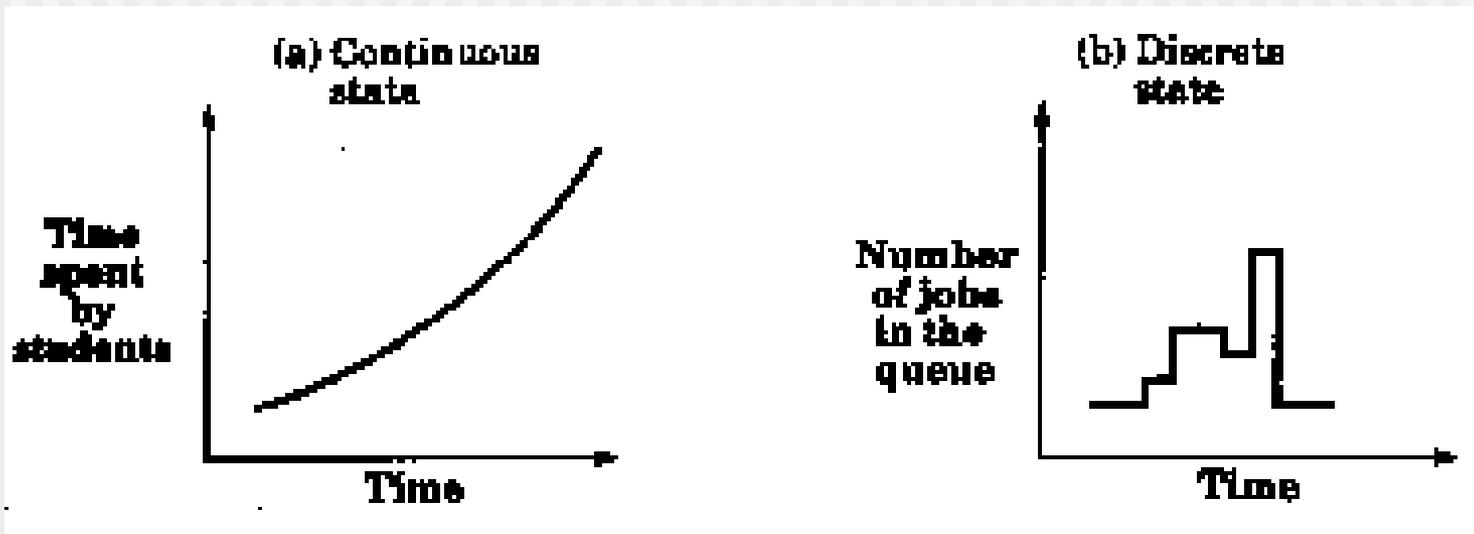
# Terminologia

- **Modelo de Tempo Discreto:** o estado do sistema está definido apenas em instantes particulares. Exemplo: Número de estudantes que assistem este curso.



# Terminologia

- **Modelos de Estado Contínuo e de Estado Discreto:** dependendo de se as variáveis de estado são contínuas ou discretas.
  - Exemplo: Tempo gasto estudando uma determinada matéria vs Número de estudantes.



# Seleção de uma Linguagem para Simulação

---

- Linguagem de simulação
- Linguagem de propósito geral
- Extensão de uma linguagem de propósito geral
- Pacote de simulação

# Linguagens de Simulação

---

- Economizam tempo de desenvolvimento
- Recursos embutidos para:
  - avançar o tempo
  - programar eventos
  - manipulação de entidades
  - geração de valores aleatórios
  - coleta de dados estatísticos
  - geração de relatórios
- Mais tempo para questões específicas do sistema
- Código modular, bastante legível

# Tipos de Linguagens de Simulação

---

- Linguagens de simulação contínuas:
  - CSMP, DYNAMO
  - Equações diferenciais
  - Usadas em engenharia química
- Linguagens de simulação de eventos discretos:
  - SIMULA, GPSS
- Combinadas:
  - SIMSCRIPT e GASP
  - Permitem simulações discretas, contínuas ou combinadas.

# Linguagem de Propósito Geral

---

- Familiaridade do analista
- Grande disponibilidade
- Início imediato
- Tempo gasto com o desenvolvimento de rotinas para tratamento de eventos e geração de valores aleatórios

# Linguagem de Propósito Geral

---

- Outras questões:
  - Eficiência
  - Flexibilidade
  - Portabilidade
- **Recomendação:** Aprenda ao menos uma linguagem de simulação, ou extensão de linguagem.

# Extensão de uma linguagem de propósito geral

---

- Exemplos:
  - GASP (para FORTRAN)
  - SMPL (para C)
  - TARVOS (para C)
  - CSIM (C e C++)
  - Coleção de rotinas para tratar tarefas de simulação
  - Compromisso entre eficiência, flexibilidade e portabilidade

# Pacotes de Simulação

---

- Exemplos: QNET4, RESQ, BONEs, NS, OPNET
  - Diálogo de entrada
  - Biblioteca de estruturas de dados, rotinas e algoritmos
  - Grande economia de tempo
  - Às vezes, pouco flexível ⇒ Simplificação

# Tipos de Simulações

---

- Emulação: utilizando hardware ou firmware
  - Exemplos: emulador de terminal, emulador de processador
  - Envolve basicamente questões de projeto de hardware
- Simulação de Monte Carlo
- Simulação Dirigida por Traces
- **Simulação de Eventos Discretos**

# Simulação de Eventos Discretos

---

- Os eventos alteram o estado do sistema de forma discreta
- Os estados só mudam em cada evento
  - Entre eventos, “nada” acontece
- Número de jobs  $\Rightarrow$  Eventos discretos
- Estados discretos  $\neq$  tempo discreto
  - Não atrapalhar estados discretos com números inteiros
  - As filas são discretas
    - 1 cliente, 2 clientes, 3 clientes, ...
  - Tempo simulado geralmente é fracionário
  - As estatísticas não são necessariamente inteiras

# Componentes da Simulação de Eventos Discretos

---

- Escalonador de Eventos
  - Programa o evento  $X$  para o instante  $T$ .
  - Congela o evento  $X$  durante o intervalo de tempo  $dt$ .
  - Cancela um evento  $X$  previamente programado.
  - Congela o evento  $X$  indefinidamente
  - Programa um evento congelado indefinidamente.

# Componentes da Simulação de Eventos Discretos

---

- Relógio de Simulação e Mecanismo de Avanço do Tempo
  - abordagem baseada em unidades de tempo
  - abordagem dirigida a eventos
- Variáveis de Estado do Sistema
  - Global = Número de jobs
  - Local = Tempo de CPU necessário para um dado job

# Componentes da Simulação de Eventos Discretos

---

- Rotinas associadas aos eventos: Uma para cada evento.
  - Exemplo: chegada de jobs, escalonamento de jobs e partida de jobs
- Rotinas de Entrada:
  - Obtenção dos parâmetros do modelo
  - Variação dos parâmetros dentro de uma certa faixa.
- Gerador de Relatórios

# Componentes da Simulação de Eventos Discretos

---

- Rotinas de Inicialização:  
Atribui o estado inicial. Inicializa as sementes.
- Rotinas de Trace: Podem ser ligadas ou desligadas
- Gerenciamento Dinâmico de Memória: Coleta de Lixo
- Programa Principal

# Algoritmos de Tratamento de Conjunto de Eventos

---

- Conjunto de Eventos = Lista ordenada do registro dos eventos futuros
  - **Cadeia de Eventos**
- Operações Básicas:
  - Inserção de um novo evento
  - Remoção do próximo evento a ser executado
- Estruturas de Dados:
  - Lista Encadeada Ordenada
  - Lista Linear Indexada
  - Fila de Calendários
  - Estruturas em Árvores
  - Heap

# Máquina de Simulação

---

*Cadeia de Eventos*

# Cadeia de Eventos

---

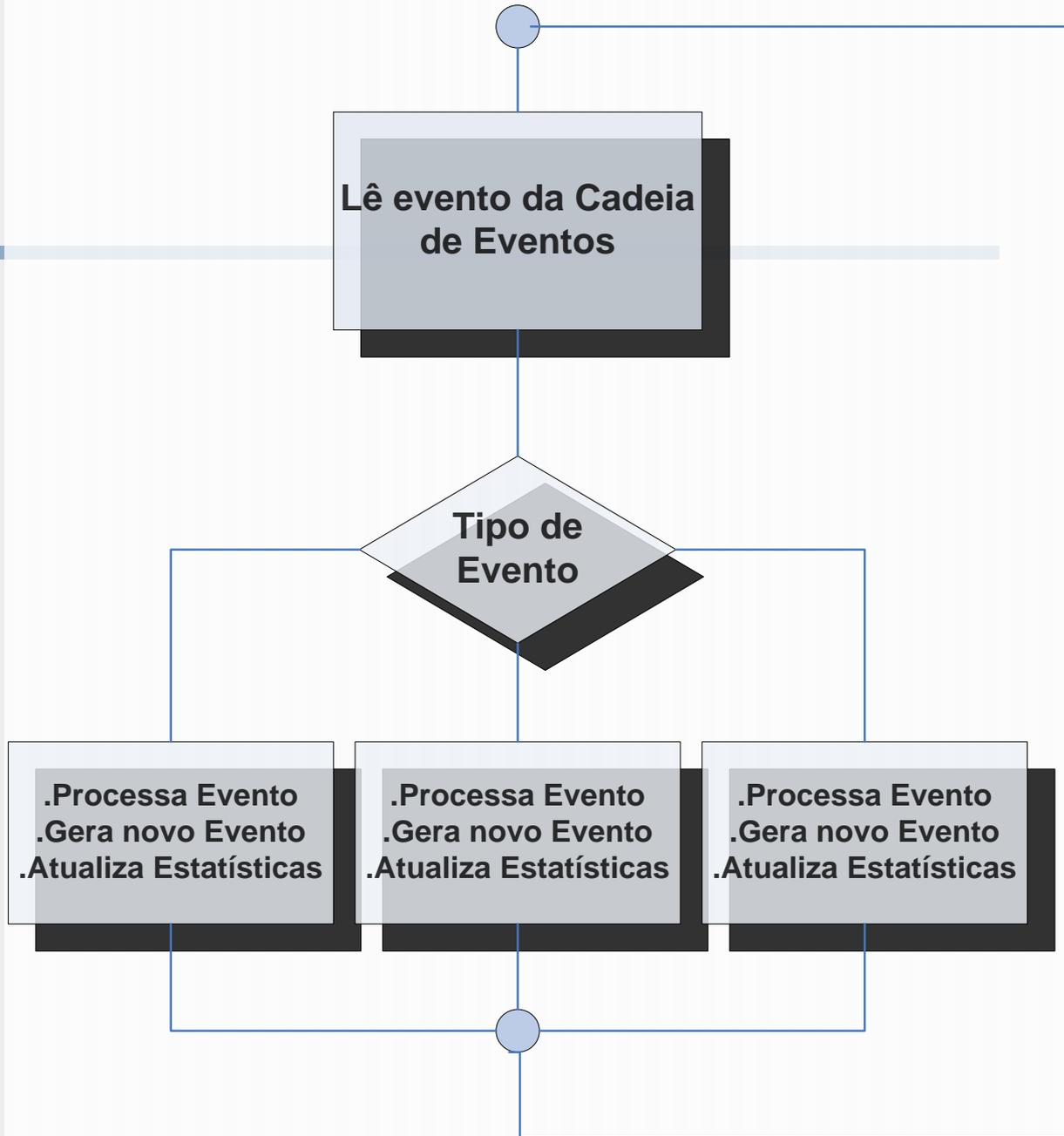
- Estrutura principal da máquina de simulação
- É geralmente uma lista dinâmica duplamente encadeada
  - Cada elemento da lista é um evento, contendo:
    - Tempo Simulado
    - Tipo de evento
    - etc.

# Cadeia de Eventos

---

- Simulador opera da seguinte maneira:
  - Retira o primeiro evento da lista (e apaga);
  - Processa o evento;
  - Gera outro evento e coloca na lista;
  - Atualiza estatísticas;
  - Continua o *loop*.

# Início Loop



# Tipos de Eventos

---

- Chegada de Cliente vindo de Fonte
  - Gera evento *Request Servidor*
  - Calcula tempo de interchegada aleatório (baseado na distribuição apropriada)
  - Gera novo evento *Chegada de Cliente* para o tempo calculado
    - Se um novo evento *Chegada de Cliente* não for gerado, as chegadas acabam!
    - Cadeia de Eventos esvaziar-se-á

# Tipos de Eventos

---

- *Request* Servidor
  - Se Servidor está livre:
    - Atualiza variável de estado para servidor ocupado
    - Calcula tempo de serviço aleatório (baseado na distribuição apropriada)
    - Gera evento *Libera (Release) Servidor* para o tempo calculado
  - Se Servidor está ocupado:
    - Enfileira o pedido de serviço (incrementa contador da fila)

# Tipos de Eventos

---

- Libera (*Release*) Servidor
  - (fim do serviço)
  - Atualiza variável de estado para servidor livre
  - Se houver fila conectada à frente do servidor, gerar evento *Request Servidor* com intervalo de tempo zero para o servidor à frente
  - Se Fila não estiver vazia:
    - Gera evento *Request Servidor* para o tempo simulado atual (intervalo de tempo zero) e coloca no topo da Cadeia de Eventos
    - Decrementa contador da fila

# Máquina de Simulação

---

Geração de  
Estatísticas com o  
SimRdAb

# Obtenção de Estatísticas

---

- Geração de tempo simulado (exponencial)
  - Número randômico
    - Tempo de interchegada
    - Tempo de serviço
- Cálculo de estatísticas da simulação
  - Taxa de chegada de clientes
  - Tamanho médio da fila
  - Tempo médio em fila
  - Tamanho máximo de fila
  - Utilização
  - Taxa de serviço

# Geração Tempo

---

- Objetivo: gerar um tempo de interchegada ou serviço, baseado em distribuição exponencial, e a partir de uma probabilidade aleatória
- Estratégia: gerar número  $\mu$  randômico, entre 0 e 1, que será a probabilidade
- Calcular o tempo segundo a distribuição exponencial

# Geração de Tempo

---

$$x = -\frac{1}{\lambda} \cdot \ln(1 - \mu)$$

No programa:

```
switch (tipo)
{
case 1 : /* Distribuicao exponencial */
    t_randon = -((t_proc_medio)*(log(1.-num_alea)));
    break;

case 2 : /* Distribuicao uniforme */
    t_randon = t_proc_inf+(num_alea*(t_proc_sup-t_proc_inf));
    break;
```

# Taxa de Chegada (em fila)

---

- Fórmula:
  - Total Clientes / Tempo Total Simulado
- No programa:

```
taxa_cheg = (tab_file[i].n_tot/t_simul);
```

\***n\_tot** é um acumulador, incrementado toda vez que um *job* entra na fila

\***t\_simul** é o Tempo Simulado Total ou Atual

# Tamanho Médio da Fila

- Estratégia: Integrar, no tempo, os tamanhos de fila tomados periodicamente; dividir pelo tempo simulado

$$Q_s = \frac{\int_0^T Q(t) dt}{T}$$

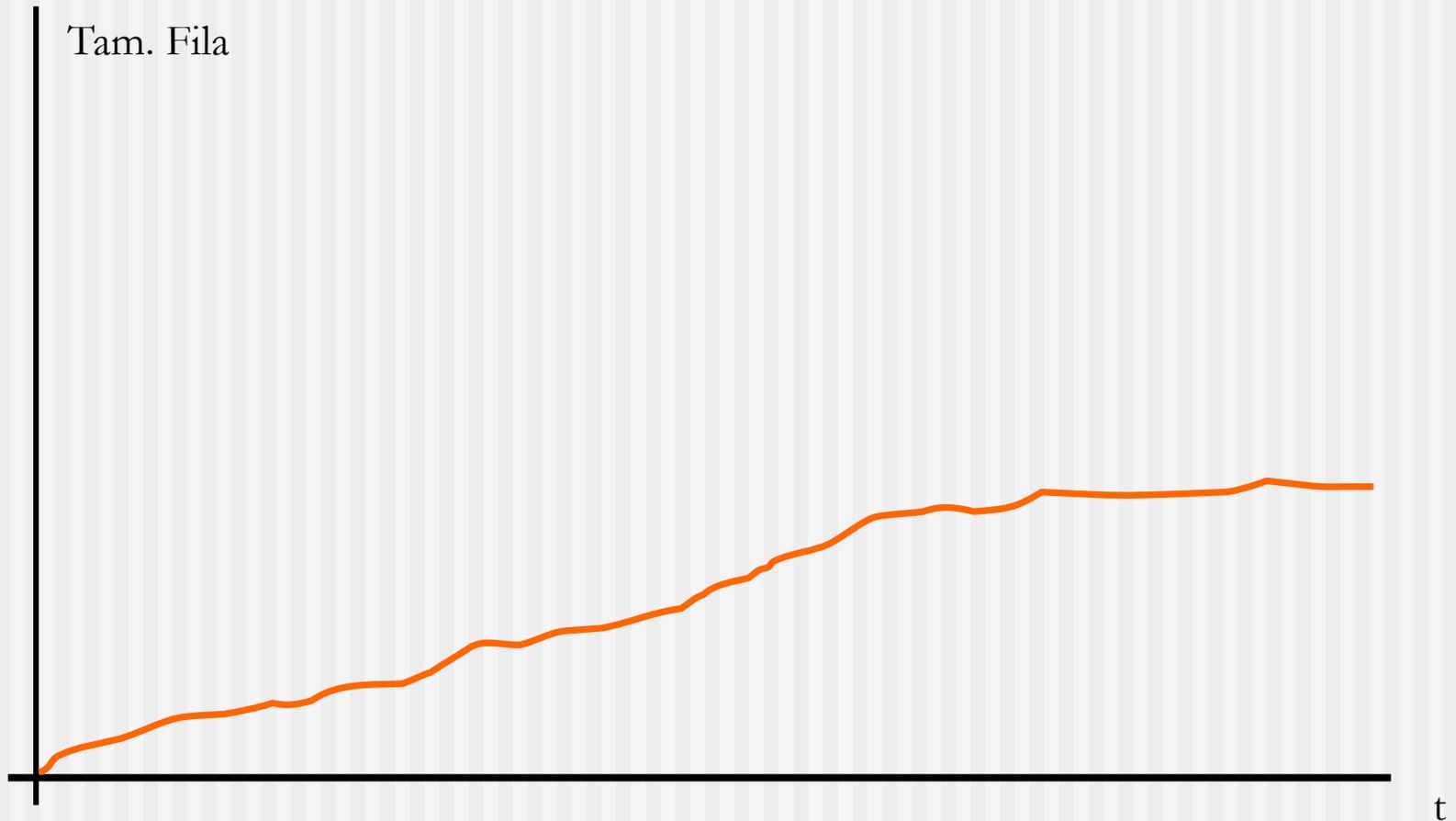
Parâmetro Contínuo

$$Q_s = \frac{\sum_{i=1}^n Q_i \cdot (t_i - t_{i-1})}{T}$$

Parâmetro Discreto

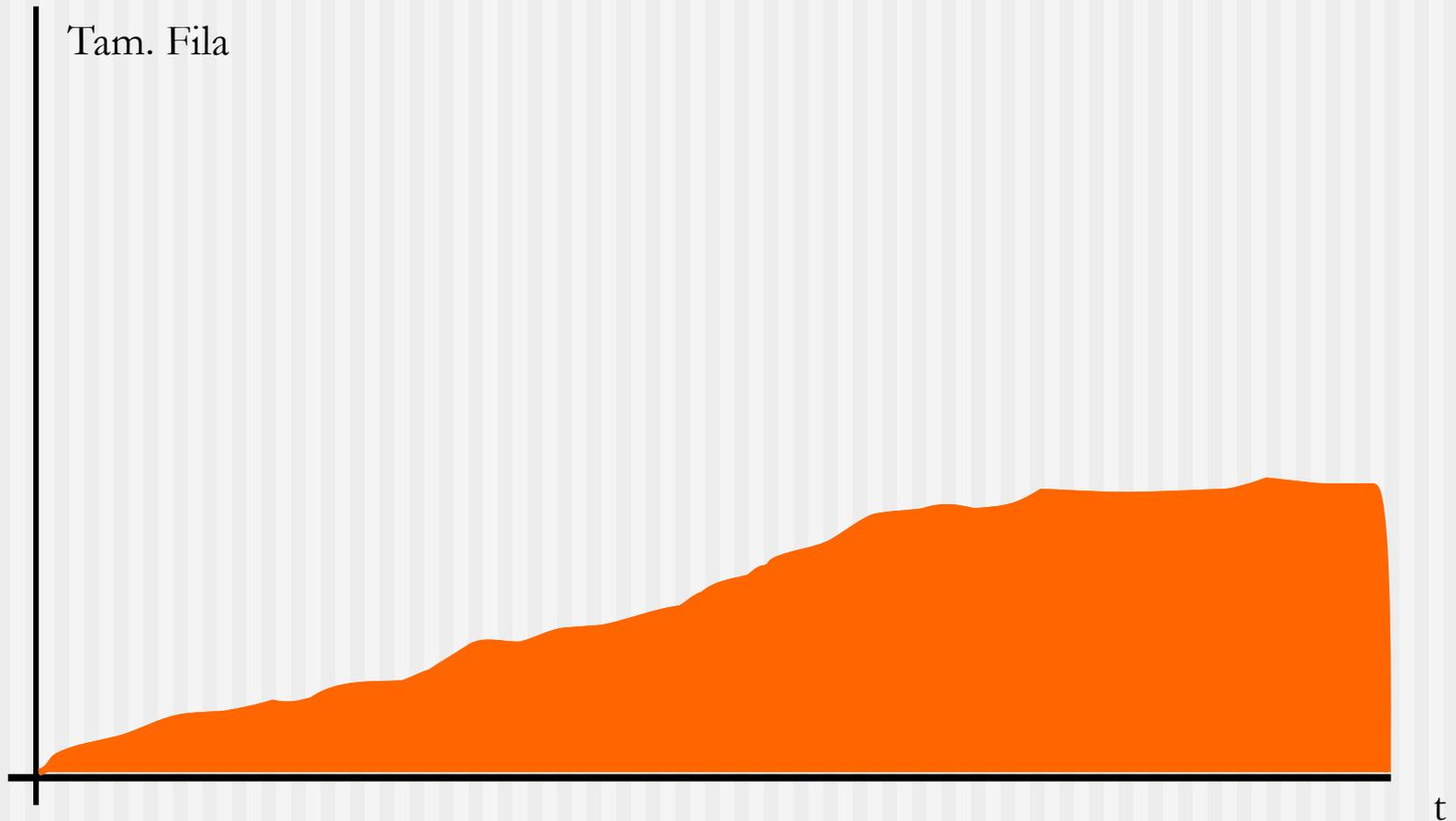
# Tamanho Médio de Fila

---



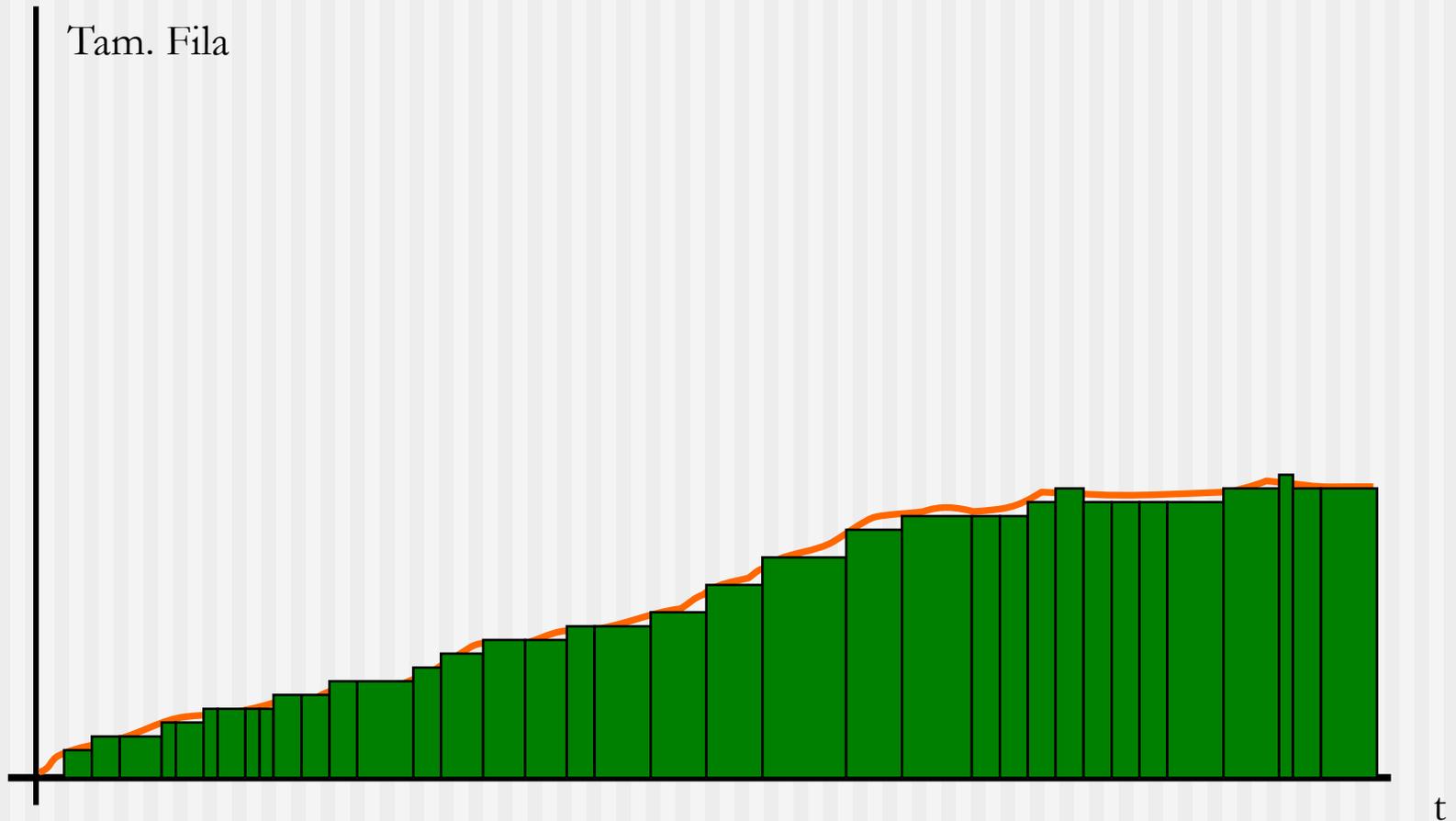
# Tamanho Médio de Fila

---



# Tamanho Médio de Fila

---



# Tamanho Médio de Fila

---

## ■ No programa:

Cálculo iterativo:

```
(fila->tot_tam_acum) += ((t_simul - fila->t_sim_ant) * (float)fila->n_tam_ant);
```

Impressão do Relatório:

```
// TAM MEDIO FILA  
taman_med_fil = (tab_file[i].tot_tam_acum/t_simul);
```

# Tempo Médio em Fila

- Objetivo: Integrar, no tempo, os tamanhos de fila tomados periodicamente; dividir pelo total de clientes que passaram pela fila
- Estratégia: usar tamanho médio de fila (que já contém a integral) e dividir pela taxa de chegada

$$Q_t = \frac{\sum_{i=1}^n Q_i \cdot (t_i - t_{i-1})}{T} \cdot \frac{T}{\text{TotCli}}$$

# Tempo Médio em Fila

---

- Lei de Little

Tamanho Médio = Tx. Chegada x Tempo Médio

Tempo Médio = Tamanho Médio / Tx. Chegada

# Tempo Médio em Fila

---

- No programa:

```
tempo_med_fil = (taman_med_fil/taxa_cheg);
```

Tamanho Médio = Tx. Chegada x Tempo Médio

Tempo Médio = Tamanho Médio / Tx. Chegada

# Tamanho Máximo de Fila

---

- Estratégia: coletar tamanho imediato da fila periodicamente; atualizar uma variável, sempre que este tamanho for maior que o armazenado anterior
- No programa:

```
// incrementa o numero de clientes na fila  
fila->n_cour++;  
// checa se o numero atual eh maior que o maximo  
if(fila->n_cour > fila->n_max) fila->n_max=fila->n_cour;
```

# Utilização

---

- Objetivo: calcular utilização
  - $\text{Tempo total ocupado} / \text{Tempo total simulação}$
- Estratégia:
  - acumular cada tempo de serviço de cada cliente em uma variável
  - dividir, no final, pelo tempo total simulado

# Utilização

---

## ■ No programa:

Cálculo iterativo:

```
//acrescenta o tempo do cliente atual no tempo ocupado total  
serv->t_tot_ocupado += serv->t_ocupado;
```

Impressão do Relatório:

```
util=( tab_serv[i].t_tot_ocupado/t_simul );
```

# Taxa de Serviço

- Objetivo: calcular taxa de serviço do servidor
  - Total de Clientes servidos/Tempo total ocupado
- No programa:

```
// TEMPO DE SERVICO  
tx_serv=(num_cli_serv/(t_simul*util));
```

$$\text{TotCli} \cdot \frac{\text{TotTime}}{\text{TotBusy}} \cdot \frac{1}{\text{TotTime}}$$