

Pipeline

INF028 – ARQUITETURA DE COMPUTADORES E SOFTWARE BÁSICO

FLAVIAMARISTELA@GMAIL.COM

Sistemas Computacionais modernos

Sistemas
Computacionais
Modernos

The diagram consists of two orange chevron-shaped boxes pointing to the right, connected by a white arrow. The first box contains the text 'Sistemas Computacionais Modernos' and the second box contains 'Processadores buscam maior velocidade'. A solid orange bar is at the bottom of the slide.

Processadores
buscam maior
velocidade

Como obter maior desempenho computacional?

Tecnologias mais avançadas → circuitos mais rápidos;

Melhor organização da CPU → o uso de múltiplos registradores e memória cache

Pipeline de instruções

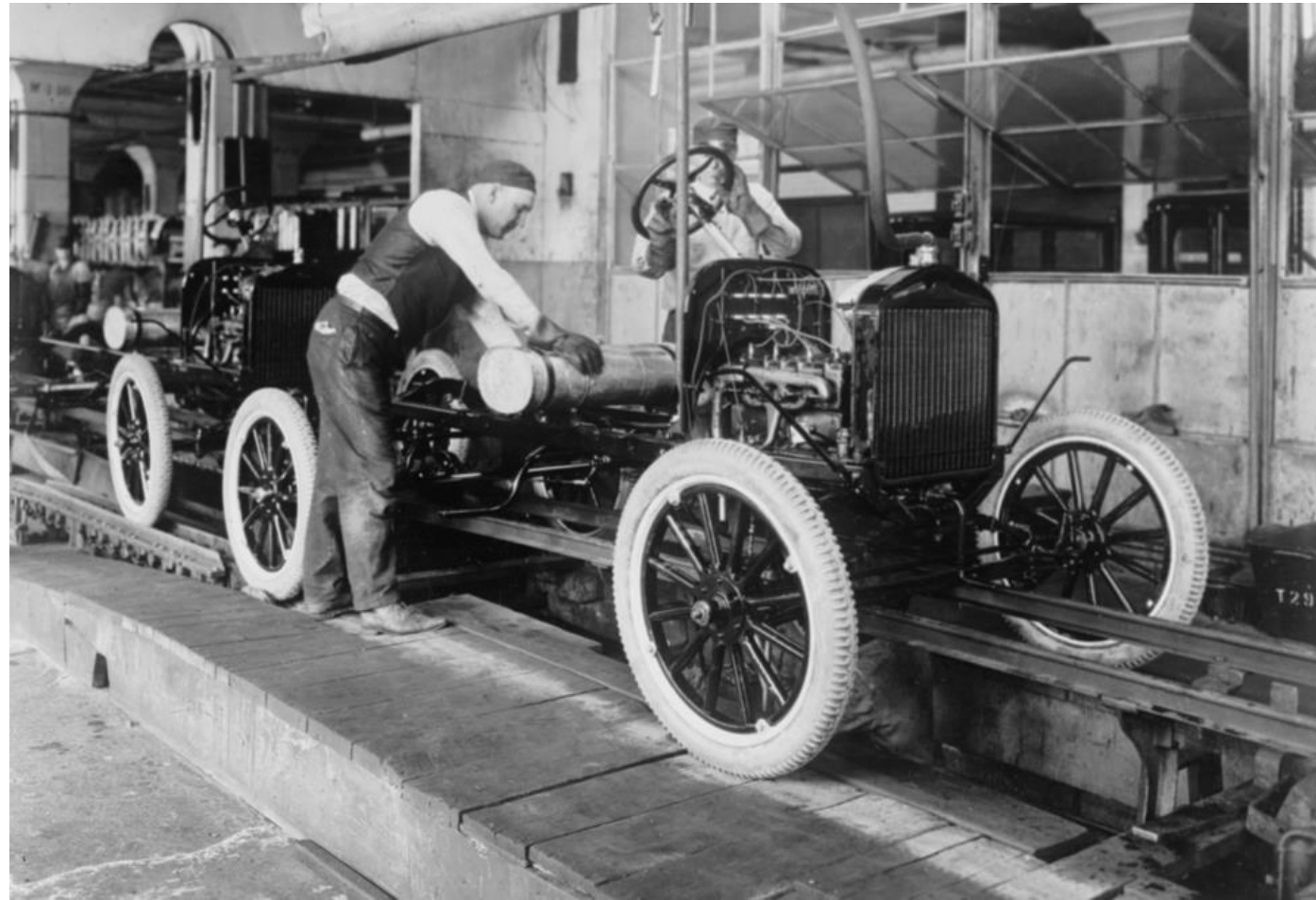
O que é pipeline?

Também conhecido como “linha de montagem”

O Pipeline divide a CPU em várias partes funcionais distintas (estágios)

Cada estágio corresponde a uma determinada atividade.

Linha de Montagem :: Henry Ford



Linha de Montagem :: Henry Ford



Linha de Montagem :: Nissan





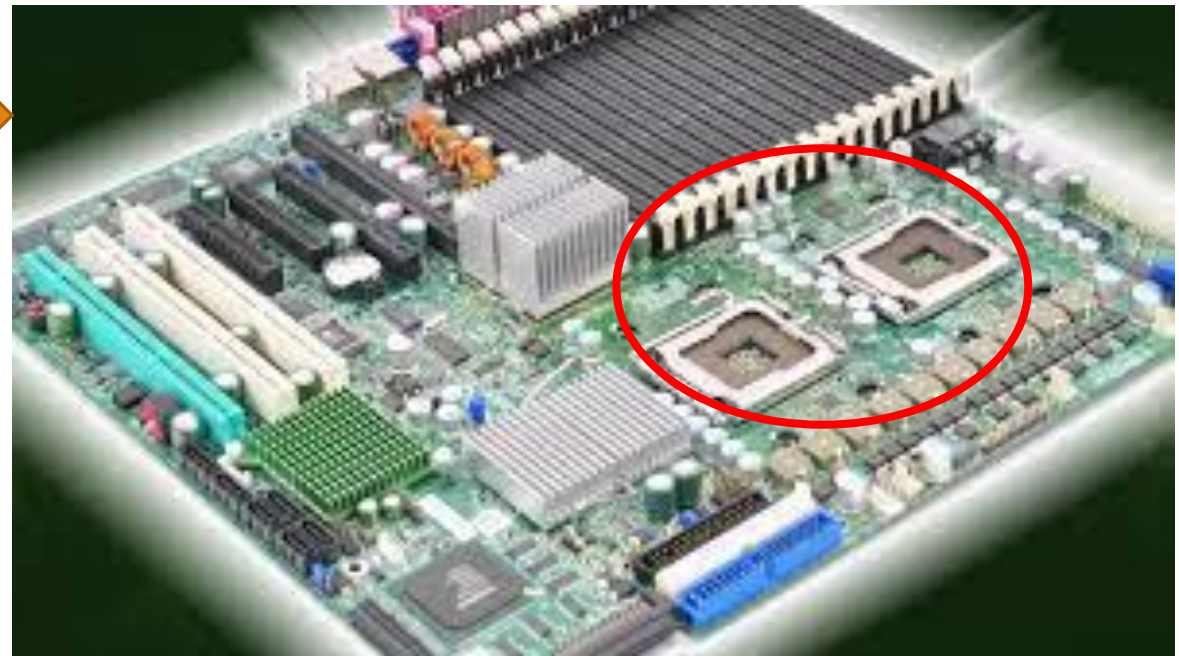
Single core

Time sharing (compartilhamento do tempo)
Multithreading (execute vários fluxos do mesmo processo)



paralela

multicore



O que é pipeline?

Pipeline é uma técnica que permite que os processadores executem processos diferentes ao mesmo tempo, **respeitando a ordem de execução das instruções que chegam ao processador;**

Em outras palavras, **novas entradas** são aceitas , antes que as entradas aceitas previamente tenham terminado;

Esta técnica aumenta o desempenho do processador e reduz o tempo de execução global de tarefas.





Vocês sabem dizer
um exemplo de
pipeline em
Computação???

!!!!



O exemplo clássico da lavanderia

Suponha uma lavanderia que execute o processo completo de lavagem em 2h:

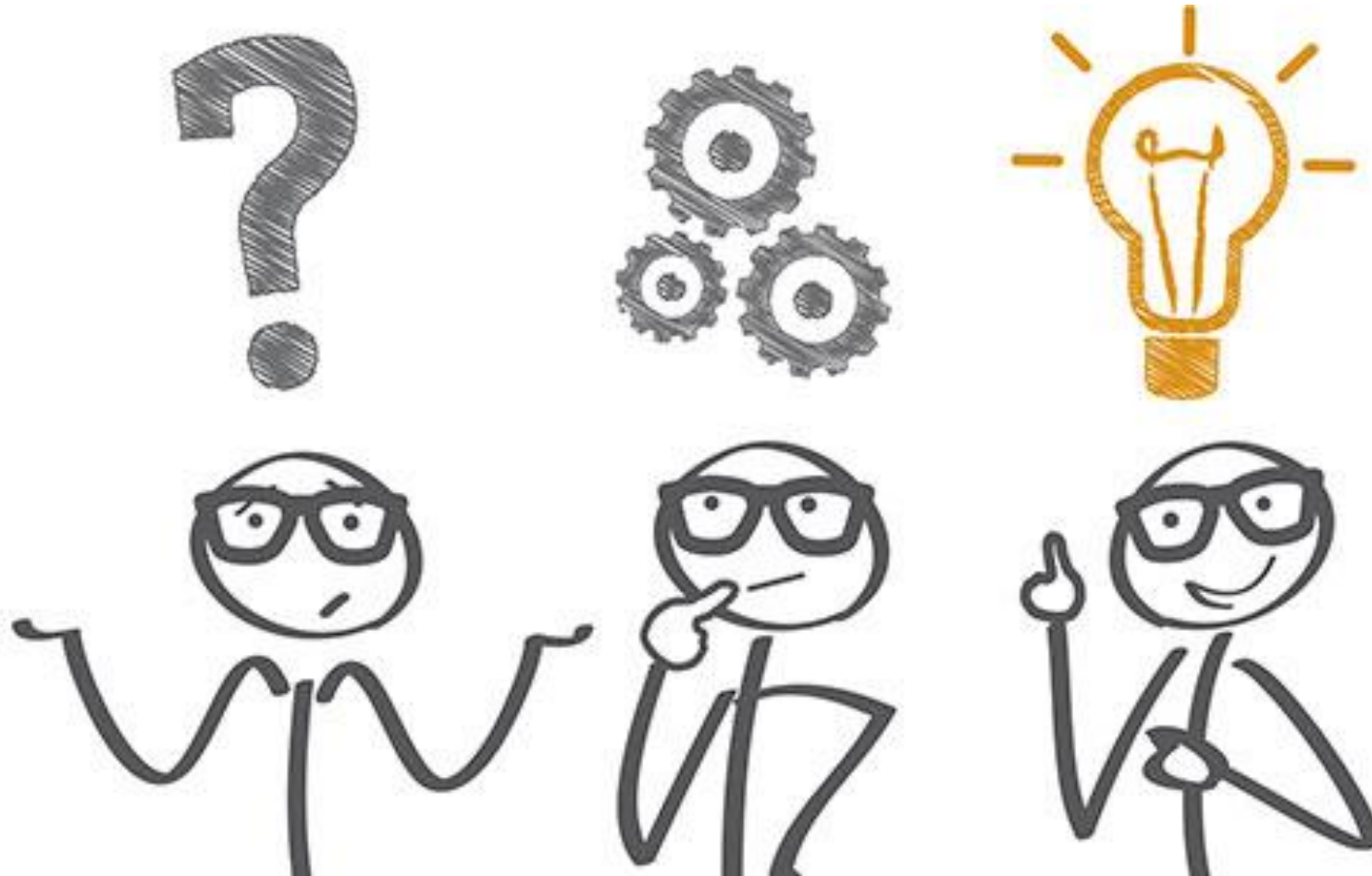
- Colocar a roupa na máquina de lavar 
- **Depois** de lavada, colocar na máquina de secar roupa 
- **Depois** de seca, passar a ferro 
- **Depois** de passada, arrumá-la no armário 

O exemplo clássico da lavanderia



Vamos pensar...

Qual é o problema?



Vamos pensar...

Qual é o problema?

O problema é que após **Esther** terminar a lavagem de roupas, a máquina de lavar ficará ociosa até que a roupa passe pela secadora, seja passada e guardada.

Somente após todo esse processo é que a máquina de lavar será liberada para **Tales**.

É possível melhorar o desempenho desta lavanderia?

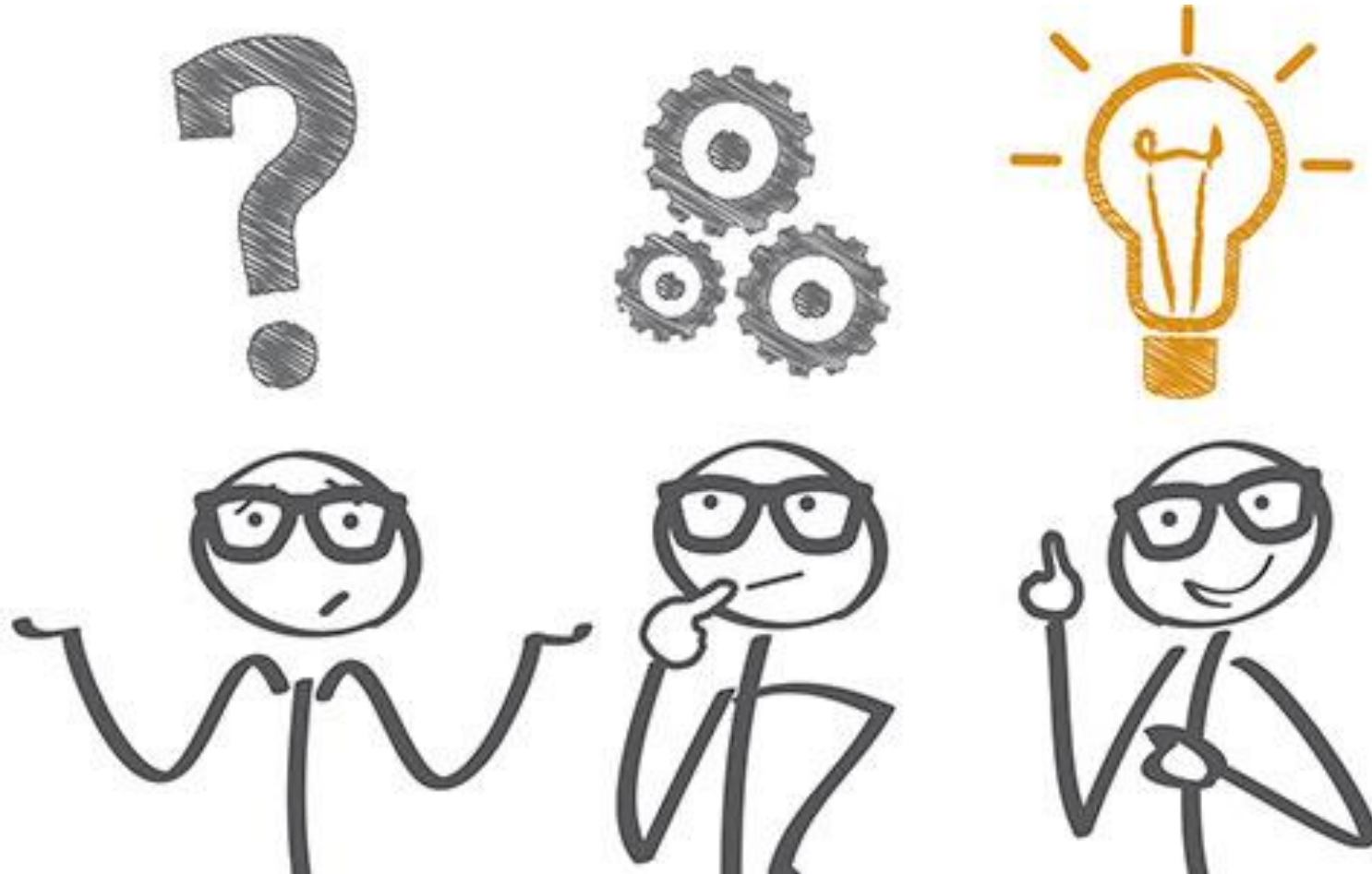




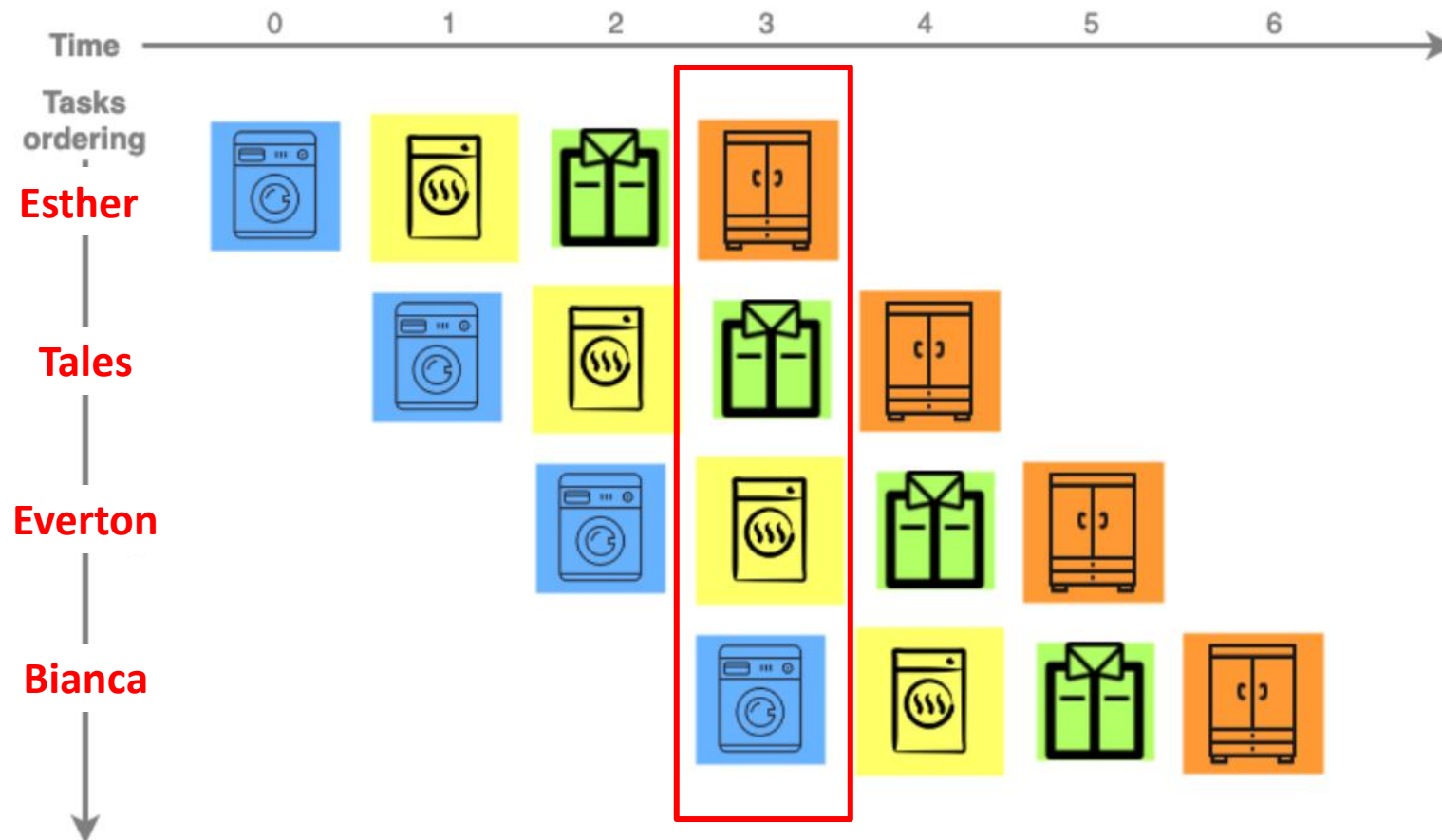
Qual é o segredo então?

- O segredo do pipeline é que assumir que cada instrução possui diferentes estágios.
- Num dado momento, cada instrução está em um único estágio;
- Os demais estágios, ao invés de ficarem ociosos, são utilizados para outras instruções;
- Isso permite executar várias instruções na CPU, cada uma em um estágio diferente

Como fica então o problema da lavanderia?



:: O exemplo clássico da lavanderia :: com pipeline



Importante!

Note que:

* Nos dois exemplos, a tarefa de “lavar a roupa” dura 2 horas.

Qual é então a vantagem de se usar o pipeline?



-
- 1) Não melhora a latência de cada tarefa individualmente;
 - 2) Melhora o *throughput* (*vazão*);
 - 3) Vários processos executam simultaneamente usando recursos diferentes;
 - 4) *Speed-up* potencial depende número de estágios do pipeline.

Um exemplo sobre execução de um programa

- Busca de Instrução (BI)
- Decodificação da Instrução (DI)
- Busca de Operandos (BO)
- Execução da Instrução (EI)
- Escrita de Operando (EO)

