UNIR

Sistemas Operacionais

Aula 05 – Concorrência
SCC5854

Prof. Dr. Jonathan Ramos
jonathan@unir.br

Departamento Acadêmico de Ciências de Computação – DACC

Núcleo de Tecnologia – NT

18/10/2022

Sumário

Sumário

Introdução

UNIR

O que é um SO?

Podem ser vistos como um conjunto de rotinas executadas de forma:

- **Concorrente** (Simultâneo), ao mesmo tempo:
  - Processador executar instruções ao mesmo tempo que outras operações, como, por exemplo, operações de E/S
  - Permite que diversas tarefas sejam executadas concorrentemente pelo sistema
  - Princípio básico para o projeto e a implementação dos sistemas multiprogramáveis
- **Ordenada**: um não pode atrapalhar o outro:
  - Lembra do exemplo do semáforo na rua...

Introdução: *overview*

UNIR

Mecanismos, técnicas e dispositivos que possibilitam a implementação da concorrência

- **Interrupções**;
- **Exceções**;
- ***Buffering***;
- ***Spooling***;
- **Reentrância**.

Conceitos são fundamentais para a arquitetura de um sistema operacional multiprogramável

Sumário

Importância da concorrência

Uma comparação entre mono e multiprogramáveis é um bom exemplo da importância do conceito de concorrência.

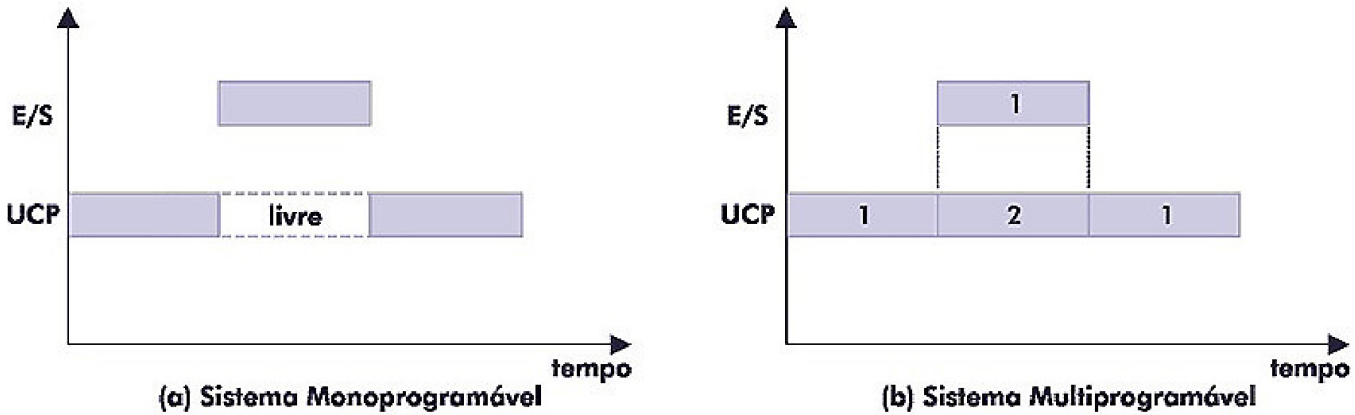


Figura: Sistema monoprogramável × sistema multiprogramável.

Sistemas Monoprogramáveis

- Somente um programa pode estar em execução por vez;
- Desperdício na utilização do processador;
- O tempo de espera de E/S é relativamente longo.

Importância da concorrência

UNIR

Leitura de um registro	0,0015 s
Execução de 100 instruções	0,0001 s
Total	0,0016 s
% utilização da CPU	(0,0001 / 0,0015) = 0,066 = 6,6%

Tabela: Exemplo de utilização do sistema.

Processador ocioso

- Processador gasta aproximadamente 93% do tempo esperando o dispositivo de E/S.
- **Em sistemas monoprogramáveis o processador é subutilizado.**

Memória principal subutilizada

Um programa que não ocupe totalmente a memória ocasiona a existência de áreas livres sem utilização

Importância da concorrência

UNIR

Sistemas multi-programáveis

Vários programas podem estar residentes em memória, concorrendo pela utilização do processador

- Quando um programa solicita uma operação de E/S **outros programas poderão utilizar o processador**;
- UCP permanece **menos tempo ociosa**;
- A memória principal é **utilizada de forma mais eficiente**;
- Existem **vários programas residentes** se **revezando na utilização do processador**.

Desafios!

- Como implementar concorrência de forma eficiente?
- Quando um programa para de processar para outro processar, deve retomar exatamente de onde parou.
- Ao usuário deve parecer que nada aconteceu.

Importância da concorrência: um exemplo prático

UNIR

Características	**Prog1**	**Prog2**	**Prog3**
Utilização da UCP	Alta	Baixa	Baixa
Operação de E/S	Poucas	Muitas	Muitas
Tempo de processamento	5 min	15 min	10 min
Memória utilizada	50 Kb	100 Kb	80 Kb
Utilização de disco	Não	Não	Sim
Utilização de terminal	Não	Sim	Não
Utilização de impressora	Não	Não	Sim

Tabela: Características de execução dos programas.

- **Prog1** não realiza operações de E/S, ao contrário de **Prog2** e **Prog3**;
- **Prog1** é processado em cinco minutos (monoprogramável);
- **30 minutos na execução dos três programas**;

Importância da concorrência: um exemplo prático

UNIR

Já em um sistema multiprogramável:

Características	Mono	Multi
Utilização da UCP	17%	33%
Utilização da memória	30%	67%
Utilização de disco	33%	67%
Utilização de impressora	33%	67%
Tempo total de processamento	30 min	15 min
Taxa de throughput	6 prog./h	12 prog./h

Tabela: Comparação entre monoprogramação e multiprogramação

Quando os programas são executados concorrentemente:

Há um ganho considerável na utilização do processador, memória, periféricos e também no tempo de resposta (sistema multiprogramável).

Sumário

Interrupções e Exceções

UNIR

Alguns erros inesperados podem acontecer....

Desvio forçado no fluxo de execução do programa

Interrupção ou exceção:

- Consequência da sinalização de algum dispositivo de hardware externo ao processador
- Ou da execução de instruções do próprio programa

A diferença entre interrupção e exceção é dada pelo tipo de evento ocorrido, porém alguns autores e fabricantes não fazem esta distinção.

Interrupções

UNIR

Interrupção

- Mecanismo que tornou possível a implementação da concorrência nos computadores;
- Fundamento básico dos sistemas multiprogramáveis;
- Sistema operacional sincroniza a execução de todas as suas rotinas e dos programas dos usuários;
- sempre gerada por algum evento externo ao programa: independe da instrução que está sendo executada

Exemplo

- Quando um dispositivo avisa ao processador que alguma operação de E/S está completa;
- O processador deve interromper o programa para tratar o término da operação.

Interrupções

Ao final da execução de cada instrução:

Unidade de Controle – UC

- Verifica a ocorrência de algum tipo de interrupção;
- O programa em execução é interrompido e o controle desviado para uma rotina responsável por tratar o evento ocorrido
- rotina de tratamento de interrupção

Figura: Mecanismos de interrupção e exceção.

Interrupções

no momento da interrupção, um conjunto de informações sobre a sua execução seja preservado:

Via hardware
1 – Um sinal de interrupção é gerado para o processador. **2 –** Após o término da execução da instrução corrente, o processador identifica o pedido de interrupção. **3 –** Os conteúdos dos registradores PC e de status são salvos. **4 –** O processador identifica qual a rotina de tratamento que será executada e carrega o PC com o endereço inicial desta rotina.
Via software
5 – A rotina de tratamento salva o conteúdo dos demais registradores do processador na pilha de controle do programa. **6 –** A rotina de tratamento é executada. **7 –** Após o término da execução da rotina de tratamento, os registradores de uso geral são restaurados, além do registrador de status e o PC, retornando à execução do programa interrompido.

Tabela: Mecanismo de interrupção.

Exceções

UNIR

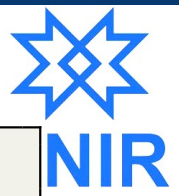
Exceções

- Resultado direto da execução de uma instrução do próprio programa;
- Divisão de um número por zero
- A ocorrência de overflow em uma operação aritmética

Principal diferença:

- **Exceção**: Síncrono.
- **Interrupção**: Assíncrono.

Interrupções e Exceções




```java
// Programa interrompido
public class DivisaoZero {
    public static void main(String[] args) {
        int a, b, c;
        a = 2;
        b = 0;
        c = a / b;
        System.out.println("Divisao de a por b = " + c);
    }
}
```

```java
// Programa com exceção
public class DivisaoZeroTratamento {
    public static void main(String[] args) {
        int a, b, c;
        a = 2;
        b = 0;
        try {
            c = a / b;
            System.out.print("Divisao de a por b = " + c);
        } catch (ArithmeticException arithmeticException) {
            System.out.println("Erro: divisao por zero");
        }
    }
}
```

Sumário

Operação de Entrada/Saída

Controlador ou interface: torna o processador independente dos dispositivos de E/S.

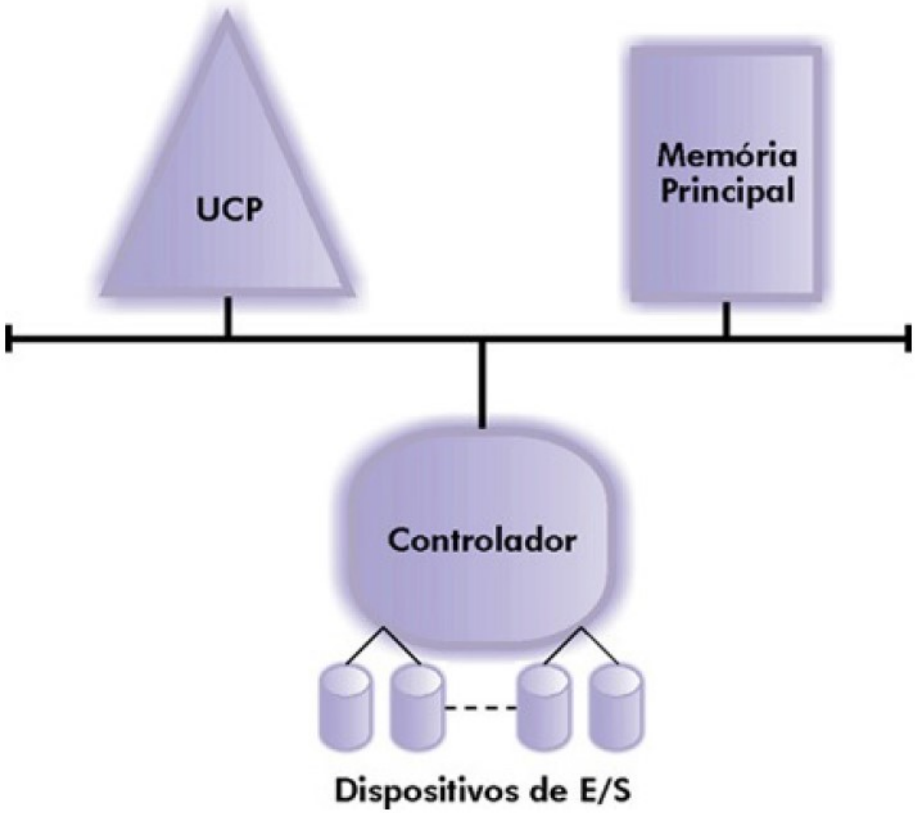


Figura: Controlador.

Instruções

- Processador não precisa de instruções específicas pra lidar com E/S.
- DMA (*Direct Memory Access*): permite que um bloco de dados seja transferido entre a memória principal e dispositivos de E/S sem a intervenção do processador, exceto no início e no final

Operação de Entrada/Saída

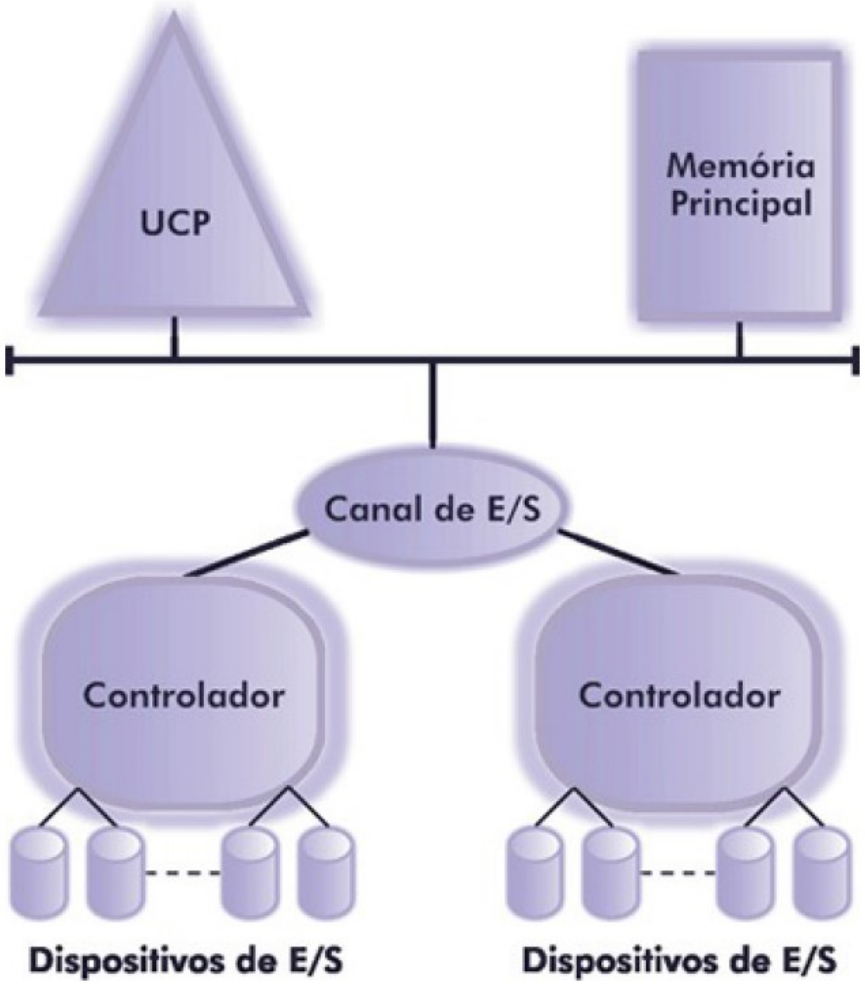


Figura: Canal de E/S.

Canal de E/S

Praticamente um processador a parte (controla os controladores)

Sumário

Buffering

UNIR

Memória temporária

Como mandar blocos de dados para um dispositivo que só aceita um bit por vez?

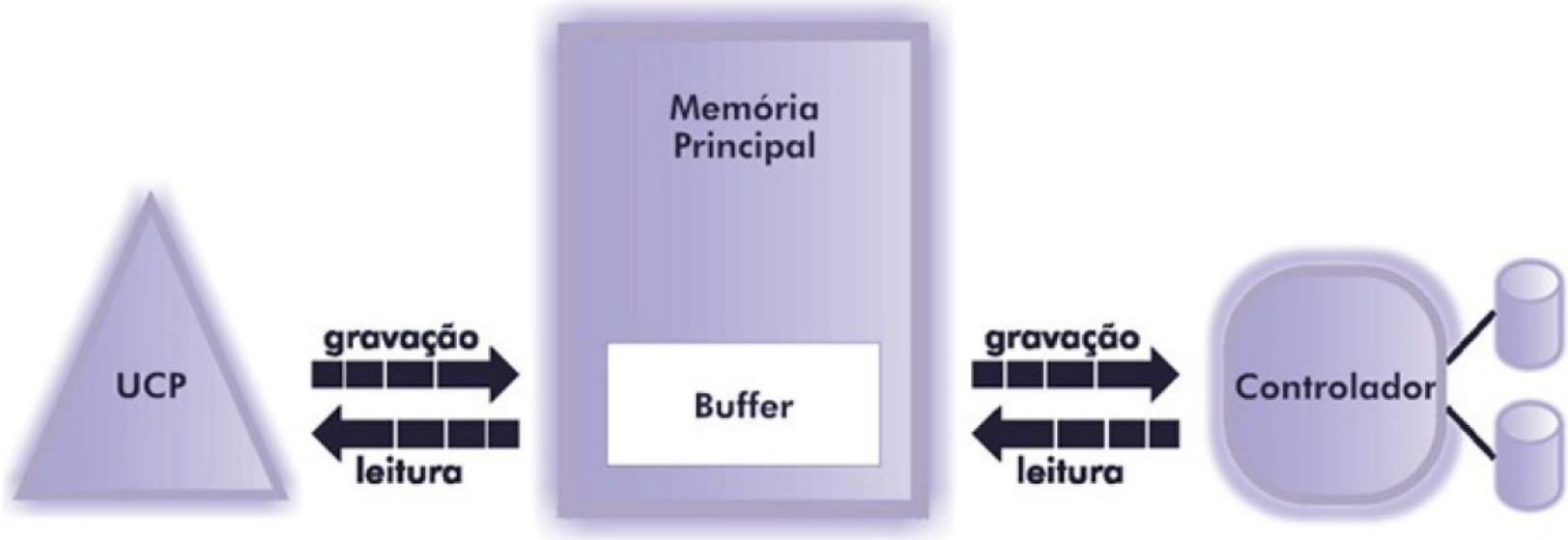


Figura: Operações de E/S utilizando *buffer*.

Sumário

Spooling

UNIR

Técnica controlada pelo SO

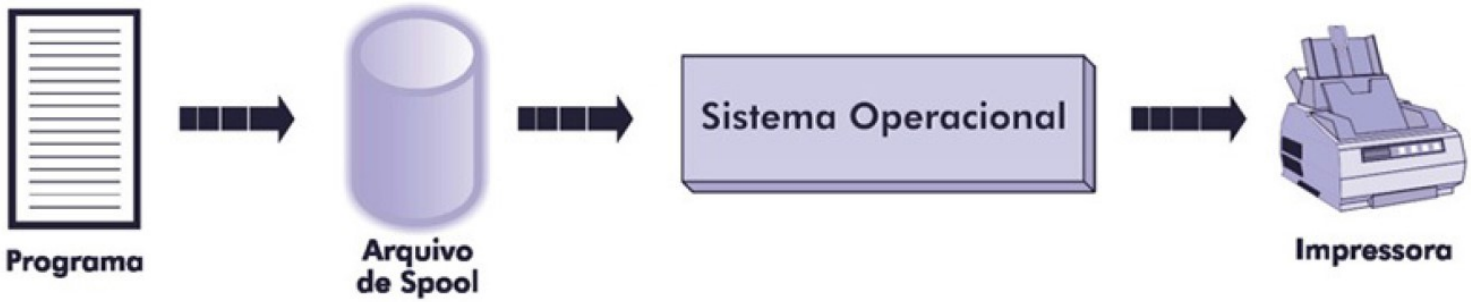


Figura: Técnica de spooling.

Sumário

Reentrâncias

Reentrâncias

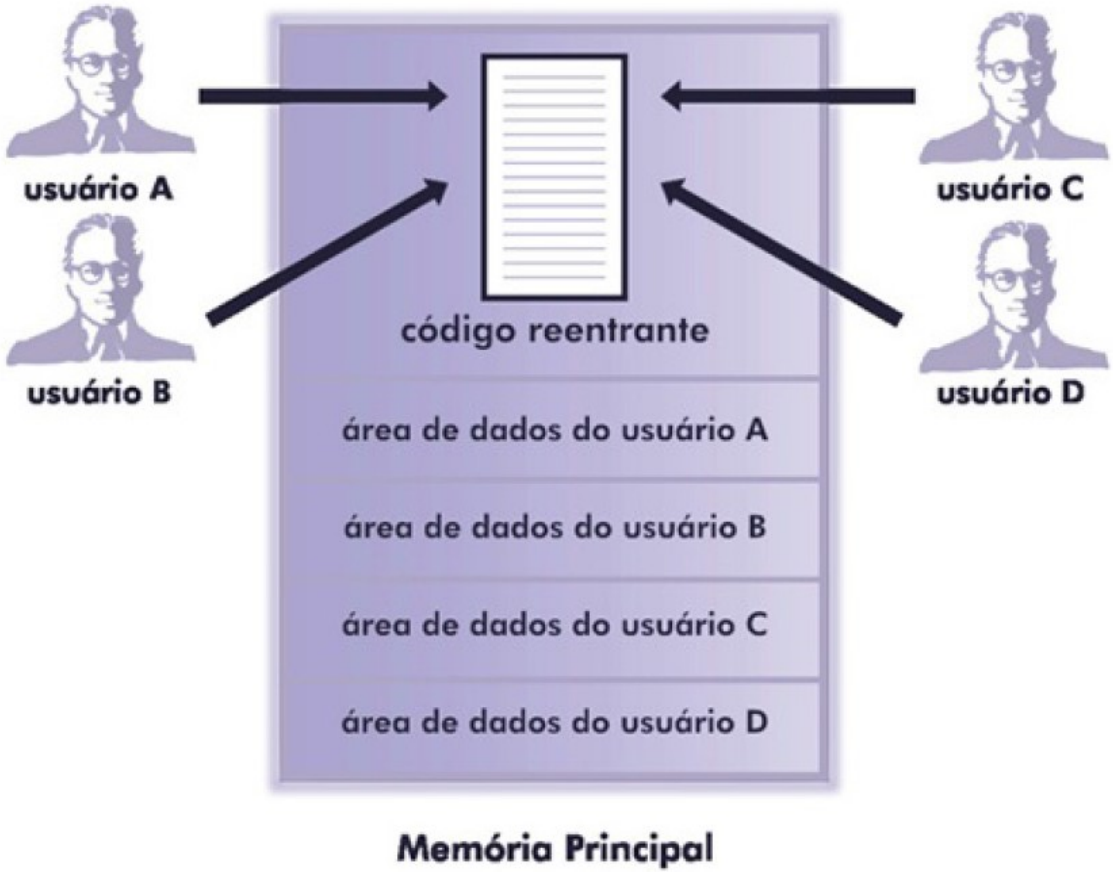


Figura: Reentrância.

Sumário

Exercícios

1. O que é concorrência e como este conceito está presente nos sistemas operacionais multiprogramáveis?
2. Por que o mecanismo de interrupção é fundamental para a implementação da multiprogramação?
3. Explique o mecanismo de funcionamento das interrupções.
4. O que são eventos síncronos e assíncronos? Como estes eventos estão relacionados ao mecanismo de interrupção e exceção?
5. Dê exemplos de eventos associados ao mecanismo de exceção.
6. O que é DMA e qual a vantagem desta técnica?
7. Como a técnica de *buffering* permite aumentar a concorrência em um sistema computacional?
8. Explique o mecanismo de *spooling* de impressão.
9. Em um sistema multiprogramável, seus usuários utilizam o mesmo editor de textos (200 Kb), compilador (300 Kb), software de correio eletrônico (200 Kb) e uma aplicação corporativa (500 Kb). Caso o sistema não implemente reentrância, qual o espaço de memória principal ocupado pelos programas quando 10 usuários estiverem utilizando todas as aplicações simultaneamente? Qual o espaço liberado quando o sistema implementa reentrância em todas as aplicações?

UNIR

FIM!

jonathan@unir.br