

Universidade Federal de Minas Gerais  
Departamento de Ciência da Computação  
Sistemas Operacionais  
Última alteração: 23 de abril de 2012

Programa	Calendário	Avaliação	Slides	Referências	Moodle	Versão PDF
----------	------------	-----------	--------	-------------	--------	------------

**AVISOS (confira também notícias no Moodle):**

- Imagem do Linux para Virtual Box disponível: SO2012-1  
A imagem foi comprimida com a ferramenta bzip2  
O login padrão da imagem é "dcc605" e a senha é "so2012-1" (sem as aspas).
  
- **Professor:** Dorgival Olavo Guedes Neto (Sala 4022)
- Local: ICEX, sala 2015
- Horário: segundas e quartas, 17:00 às 18:40
- **Estagiário em docência:** Pedro Bragioni Las Casas
- **Dúvidas:** Utilize o Moodle.  
**Não envie mail diretamente para o professor, exceto para o caso de querer discutir código!**

## Informações gerais

### Objetivo

Abordar os princípios básicos da área de Sistemas Operacionais com uma ênfase na noção de sistemas de computação, usando uma visão voltada para algoritmos e interfaces.

Ao final do curso o aluno deve ser capaz de identificar os principais elementos de um Sistema Operacional, analisar compromissos no desenvolvimento de soluções para problemas como contenção de recursos, escalonamento, gerência de memória e uso de armazenamento secundário, bem como descrever os princípios de operação de Sistemas Operacionais com relação a esses fatores.

Os conceitos da disciplina serão exercitados através de exercícios práticos voltados para a verificação de características específicas de sistemas operacionais e do desenvolvimento de versões a nível de usuário de alguns dos sistemas discutidos.

### Ementa

Conceitos de sistemas operacionais, redes de computadores, protocolos e serviços de comunicação. Arquitetura de redes de computadores. Camadas inferiores do modelo de referência de ISO: física, enlace, redes e transporte. Redes locais, interligação de redes; especificação de protocolos.

### Programa

O curso cobrirá a maior parte do livro texto adotado, com ênfase maior nos princípios básicos da área. Questões mais avançadas e de interesse mais recente, como segurança e aplicações multimídia, serão tratadas de forma mais superficial. Os alunos são encorajados a ler no livro as seções que não serão discutidas em sala de aula.

- Processos:
  - comunicação,
  - sincronização,
  - escalonamento;
- Gerência de memória:
  - *swapping*,

- paginação,
- segmentação,
- memória virtual;
- Sistema de arquivos;
- Gerência de entrada/saída (E/S).

## Calendário

(Calendário tentativo, sujeito a alterações)

Aula	Data	Assunto	Obs.
1	05/03	Introdução. Conceitos gerais.	
2	07/03	Conceitos gerais. Sistemas de computação	
—	12/03	Não houve aula	
3	14/03	Sistemas: interrupções, hardware de proteção	TP1 ->
4	19/03	Estruturas do Sistema Operacional	
5	21/03	Estruturas do Sistema Operacional	
6	26/03	Processos	<- TP1
7	28/03	Processos — escalonamento, trocas de contexto	
8	02/04	Processos — cooperação entre processos	
9	04/04	Processos. <i>Threads</i>	TP2 ->
10	09/04	Escalonamento	
11	11/04	Escalonamento	
12	16/04	Escalonamento. Problemas de sincronização	
13	18/04	Sincronização e comunicação entre processos	
14	23/04	Sincronização entre processos. Primitivas	
15	25/04	Problemas clássicos de sincronização	TP3 ->
—	30/04	Aula cancelada (SBRC)	<- TP2
16	02/05	Primeira prova	
17	07/05	<i>Deadlocks</i>	
18	09/05	Gerenciamento de memória	
19	14/05	Paginação e segmentação	
20	16/05	Memória virtual	
21	21/05	Memória virtual	
22	23/05	Interface do Sistema de Arquivos	TP4 ->
23	28/05	Implementação do Sistema de Arquivos	<- TP3
24	30/05	Implementação do Sistema de Arquivos	
25	04/06	Matéria a ser definida	
26	06/06	Matéria a ser definida	
27	11/06	Matéria a ser definida	
28	13/06	Matéria a ser definida	
29	18/06	Matéria a ser definida	
30	20/06	Matéria a ser definida	
—	25/06	Não há aula prevista	
—	27/06	Não há aula prevista	
—	02/07	Não há aula prevista	
—	04/07	Não há aula prevista	

# Avaliação

## Distribuição dos pontos

- **Provas:** três, *aproximadamente* 50 % da nota final.  
Não há prova suplementar, exceto para casos de ausência justificada.
- **Trabalhos Práticos:** quatro, *aproximadamente* 50 % da nota final

## Provas

Teremos três provas, cada uma valendo 17 pontos, com datas e matéria a serem definidas ao longo do curso. (Consulte o calendário para as datas corretas.)

Conteúdo das provas: **(a confirmar)**

- Primeira prova: capítulos 1 a 6.
- Segunda prova: capítulos 6 a 10.
- Terceira prova: capítulos 9 a 14.

Obviamente, cada prova pode ser composta sobre os conceitos abordados anteriormente em outras provas.

## Exercícios

Serão disponibilizadas (no moodle) listas de exercícios sobre o assunto de cada capítulo, com soluções. Apesar de não valerem pontos, essas listas visam exercitar os principais conceitos associados e servem como preparação para as provas.

## Trabalhos

Os trabalhos ainda estão sendo definidos, portanto esta lista vai evoluir. A princípio serão quatro trabalhos com valores como descritos a seguir.

- **TP1:** Sistemas operacionais históricos  
**(Trabalho individual ou em dupla).** 5 pontos.
- **TP2:** Implementação de uma chamada de sistema  
**Trabalho individual ou em dupla.**  
15 pontos.
- **TP3:** Um interpretador de comandos do usuário (*shell*) — 15 pontos.
- **TP4:** Um pacote de *threads* — 15 pontos.

As práticas devem ser desenvolvidas no ambiente Linux usando as linguagens ANSI C/ANSI C++, sem bibliotecas adicionais além das bibliotecas padrão de cada linguagem (e do Linux, obviamente). Uma boa opção, para aqueles que têm acesso a um computador em casa, é instalar uma versão de Linux. Isso pode ser feito com uma instalação como a Ubuntu a partir de um CD, ou você pode instalar um ambiente virtual como o VMware ou o Virtual Box e executar uma máquina virtual Linux. No caso do Ubuntu, certifique-se que os seguintes pacotes sejam instalados: `gcc`, `gcc-doc`, `glibc-doc`, `manpages`, `manpages-dev` e `ddd`. Isso pode ser feito com o comando:

```
sudo apt-get install gcc gcc-doc glibc-doc manpages manpages-dev ddd
```

## 0.1 Sobre prazos e formato de entrega:

Todos os trabalhos devem ser entregues eletronicamente, como um arquivo .zip ou .tar.gz, contendo todo o material relacionado ao trabalho em questão — não inclua arquivos objeto (.o) nem arquivos executáveis; um relatório deve fazer parte da entrega, detalhando aspectos principais do trabalho executado. **O relatório e todos os arquivos fonte devem conter o(s) nome(s) dos autores.**

A entrega eletrônica deve ser feita até o **meio-dia** do dia de entrega do trabalho.

Trabalhos entregues depois do prazo serão penalizados em 12,5 % se entregues ainda no dia marcado (mas após o horário definido), em 25 % se entregues até as 23:59:59 do dia seguinte e 50 % nas 24 horas seguintes. Após esse prazo, não serão mais aceitos.

Para evitar problemas, comece a fazer cada trabalho assim que ele for definido. O prazo planejado para cada entrega prevê trabalho contínuo durante o período; deixar para fazer tudo nos últimos dias **não vai dar certo**, acreditem.

Parte da avaliação dos trabalhos poderá ser feita na forma de questões escritas nas provas teóricas e/ou por entrevistas.

**Confira as datas dos trabalhos no calendário do curso**

## Referências e links úteis

### Livro-Texto

Fundamentos de Sistemas Operacionais, sexta edição, 2004, Silberschatz, Galvin e Gagne, Ed. LTC.

**Atenção:** há uma outra versão semelhante, porém baseada no uso de Java como ambiente de aplicação. Apesar do conteúdo ser bastante semelhante, a diferença de enfoque pode deixar de fora alguns aspectos importantes.

### Bibliografia suplementar

Existem outros bons livros sobre Sistemas Operacionais por aí, mas eles não trazem o material com a profundidade e organização desejadas para este curso, portanto não são recomendados como substituto para o livro texto adotado. Entretanto, alunos em busca de uma outra visão sobre o tema podem querer consultar esses livros eventualmente:

- Sistemas Operacionais Modernos, Tanenbaum  
A outra referência mais importante na área.
- Projeto e Implementação de Sistemas Operacionais, Tanenbaum  
O primeiro livro com enfoque mais moderno na área. Foca toda a discussão sobre o código fonte de um S.O. desenvolvido para o livro, o Minix.
- Stevens, W.R., *Advanced Unix Programming*, uma boa para se conferir todos os detalhes das interfaces de programação de processos, E/S e memória do Unix/Linux.  
É basicamente uma versão estendida do livro *Unix Network Programming, 2nd ed.* do mesmo autor, pela Prentice Hall (não sei se há tradução). Esse livro é tão famoso entre os programadores Unix que já apareceu até no cinema!

### Links úteis

- Como instalar o linux no HD ou usando VMWare

## Assuntos variados

Alguns links úteis que trazem mais detalhes sobre tópicos específicos discutidos durante o curso.

- **C:** o tutorial “Programming in C” tem muitas informações importantes, como o uso de sinais, etc.
- **Codificação:** detalhes de diferentes técnicas de codificação.

## Artigos

Eventualmente nesta seção serão incluídos artigos de extrema relevância para a área de sistemas operacionais que complementem o material do livro, para aqueles que se interessarem por mais detalhes. Muitos deles apresentam conceitos básicos de aplicação ampla, alguns detalham elementos importantes desenvolvidos recentemente, enquanto outros possuem um interesse histórico, discutem pesquisas recentes ou caminhos futuros para a área. Praticamente todos eles servem bem como exemplos de trabalhos com uma visão abrangente de sistemas, onde os problemas são analisados e as soluções definidas com uma preocupação constante com o sistema como um todo.

**Hints for Computer System Design** (ACM SOSP 1983)