



دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

# اصول و مفاهیم سیستم عامل

(ترم بهمن ۹۷)

## فصل اول: مقدمه

نستوه طاهری جوان

[nastoooh@aut.ac.ir](mailto:nastoooh@aut.ac.ir)



## جایگاه سیستم عامل

✓ دو جنبه اصلی کامپیوترها

○ سخت افزار

- مانند پردازنده، حافظه اصلی، دستگاههای ورودی و خروجی و ...

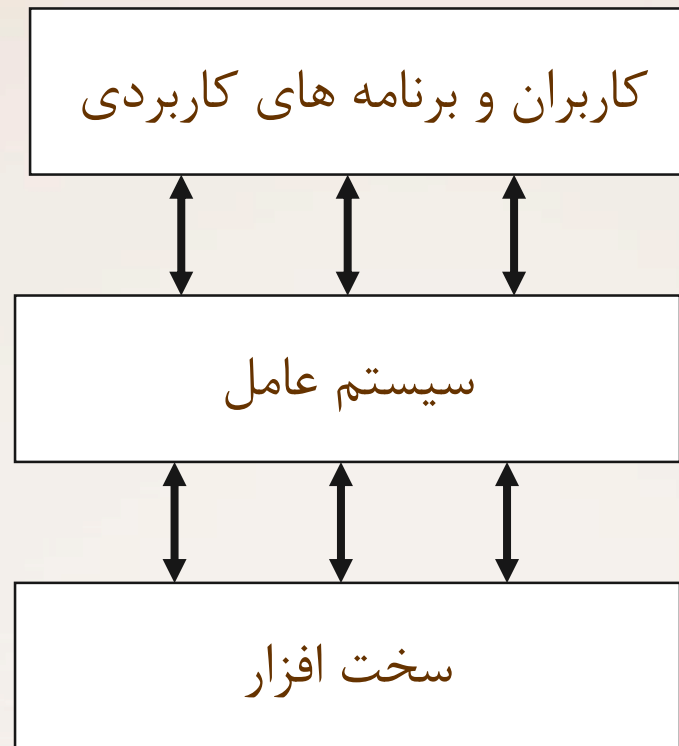
○ نرم افزار

- نرم افزارهای کاربردی
  - برای رفع نیاز کاربران
  - نرم افزارهای سیستمی
- برای بهره برداری از سخت افزار و نرم افزار ها



## جایگاه سیستم عامل

✓ سیستم عامل: مهم ترین نرم افزار سیستمی و واسط بین سخت افزار و انسان





## وظایف سیستم عامل

✓ ایجاد یک ماشین توسعه یافته (ماشین مجازی)

پنهان کردن جزئیات دست و پا گیر سخت افزار از دید کاربر (برنامه نویس)

✓ مدیریت منابع

مانند پردازنده، حافظه، دستگاههای ورودی/خروجی و ...



## روند تکامل سیستم عامل ها

### ✓ نسل اول کامپیوترها (۱۹۴۵ تا ۱۹۵۵)

- نوشتن برنامه ها به زبان ماشین (صفر و یک)
- استفاده کامپیوتر فقط توسط طراحان و سازندگان آن
- نیاز به آشنایی کامل کاربران با جزئیات سخت افزار
- در واقع عدم وجود سیستم عامل!

### ✓ نسل دوم کامپیوترها (۱۹۵۵ تا ۱۹۶۵)

- نوشتن برنامه ها به زبان هایی مانند اسمبلی (یا حتی فرترن)
- پانچ برنامه ها بر روی کارت ها و تحویل به اپراتورها
- سیستم عامل: اپراتورهای سیستم و بعد برنامه ساده همواره مقیم در حافظه



## روند تکامل سیستم عامل ها

✓ سیستم های دسته ای (Batch System) (عموماً مربوط به نسل دوم)

- جمع آوری کارهای مختلف توسط یک کامپیوتر ارزان قیمت
- ذخیره کارها بر روی نوارهای مغناطیسی
- انتقال نوارها توسط اپراتور به کامپیوتر اصلی
- پردازش کارها به ترتیب توسط کامپیوتر اصلی (گران قیمت)
- ذخیره خروجی توسط کامپیوتر اصلی بر روی نوار
- انتقال نوارها توسط اپراتور به کامپیوتر ارزان
- چاپ خروجی ها توسط کامپیوتر ارزان



## روند تکامل سیستم عامل ها

### ○ مزیت سیستم های دسته ای

- جبران اتلاف وقت کامپیوترها در نتیجه راندمان بالاتر
- همزمانی عملیات ورودی/خروجی در کامپیوتر ارزان با پردازش در کامپیوتر گران
- عملیات ساده تر
- سهولت برای استفاده راه دور

### ○ عیب سیستم های دسته ای

- افزایش زمان گردش کار (زمان بین تحویل کار تا دریافت نتایج)
- به دلیل تحویل چندین کار به صورت دسته ای به سیستم
- عدم وجود قابلیت اولویت بندی (پارتی بازی!)
- به دلیل استفاده از نوار (معمولاً)

❖ گاهی به سیستم های دسته ای، Offline SPOOLing هم می گویند.



## روند تکامل سیستم عامل ها

❖ تعریف SPOOLing

مخفف: Simultaneous Peripheral Operation On Line

عملیات جانبی آنلاین به صورت همزمان

❖ تفاوت Offline SPOOLing و On line SPOOLing

- استفاده از نوار در آفلاین برای انتقال کارها
  - عدم وجود اولویت بندی
  - زمان گردش کار بالا تر
- استفاده از دیسک در آنلاین برای انتقال کارها
  - امکان اولویت بندی
  - زمان گردش کار پایین تر





## روند تکامل سیستم عامل ها

✓ نسل سوم کامپیوترها (۱۹۶۵ تا ۱۹۷۵) و سایر نسل ها

- سیستم عامل چند برنامه ای (Multi Programming)
- سیستم عامل اشتراک زمانی (Time Sharing)
- سیستم عامل چند وظیفه ای/چند نخی (Multi Tasking/Threading)
- سیستم عامل چند پردازنده ای/پردازشی (Multi Processing/Processor)
- سیستم عامل بلادرنگ (Real Time)
- سیستم عامل شبکه (Network)
- سیستم عامل توزیع شده (Distributed)



## روند تکامل سیستم عامل ها

○ سیستم های چند برنامه ای (Multi Programming)

- می توان گفت: نوع خاصی از سیستم های دسته ای
- استقرار چندین برنامه به طور همزمان در حافظه
- استفاده مفید از زمان پردازنده
- کماکان اجرای یک برنامه در آن واحد

وضعیت حافظه

سیستم عامل
<b>Job</b>
بلا استفاده

تک برنامه ای

وضعیت حافظه

سیستم عامل
<b>Job 1</b>
<b>Job 2</b>
<b>Job 3</b>
<b>Job 4</b>

چند برنامه ای



## روند تکامل سیستم عامل ها

❖ انواع برنامه ها از نظر منابع مصرفی

❖ CPU Bounded (CPU Limited)

صرف عمده زمان برنامه برای پردازش (مانند برنامه های محاسباتی و مهندسی)

❖ I/O Bounded (I/O Limited)

صرف عمده زمان برنامه برای ورودی/خروجی (مانند برنامه های تجاری یا حسابداری)

❖ تکنیکی برای افزایش کارایی سیستمهای چند برنامه ای:

دعوت ترکیب مناسبی از برنامه های CPU Limited و I/O Limited به حافظه برای استفاده بهینه از پردازنده و دستگاههای ورودی خروجی



## روند تکامل سیستم عامل ها

### ○ سیستم های اشتراک زمانی (Time Sharing)

- تقسیم زمان پردازنده بین چند برنامه
- سرویس دهی پردازنده به صورت گردشی بین فرآیندها تا اتمام فرآیند
- توسعه یافته سیستم های چند برنامه ای
- ارتباط آنلاین کاربران با برنامه ها
- سرویس دهی متناوب پردازنده به برنامه ها
- ورود مفهوم ترمینال از سیستم های اشتراک زمانی به بعد



## روند تکامل سیستم عامل ها

○ سیستم های چند وظیفه ای/چند سرخشی (Multi Tasking/Threading)

- توانایی تقسیم برنامه ها به چند Thread
- اختصاص یک P.C. و مجموعه رجیستر جداگانه به سرخ های یک فرآیند
- استفاده سرخ ها از منابع یکسان (مانند کد و داده)

○ سیستم های چند پردازنده ای/ چند پردازشی

- استفاده از چند پردازنده در یک سیستم
- توانایی اجرای موازی چندین عمل بر روی چندین پردازنده
- افزایش توان عملیاتی و تحمل پذیری خطا



## روند تکامل سیستم عامل ها

### ○ سیستم عامل شبکه

- اطلاع از وجود سیستم های دیگر
- توانایی به اشتراک گذاری منابع برای سیستم های دیگر
- توانایی استفاده از منابع دیگر سیستم ها
- استقلال سیستم ها از یکدیگر

### ○ سیستم های بلادرنگ

- نوع خاصی از سیستم با زمان پاسخ نهایی از پیش تعیین شده
- کاربرد: مثلا کنترل یک دستگاه خاص صنعتی
- هدف نهایی: پاسخگویی سریع و حتی الامکان تضمین شده
- سیستم های بلادرنگ سخت
- رعایت ضرب العجل ها تحت هر شرایطی (دیر پاسخ دادن به بدی پاسخ ندادن)
- سیستم های بلادرنگ نرم
- دادن اولویت بالا به برخی از فرآیندها در برابر دیگر فرآیندها



## روند تکامل سیستم عامل ها

○ سیستم های توزیع شده (Distributed Systems)

- توزیع منابع در شبکه
- عدم اطلاع کاربر از محل فایل ها و مکان اجرای فرآیندها
- اضافه شدن یک لایه به نام میان افزار (Middle Ware) بین کاربر و سیستم عامل
- یکپارچه بودن سیستم از دید کاربر





## اجزای اصلی سیستم عامل

### ✓ مدیر فرآیندها

- ایجاد و حذف فرآیندها
- توقف و از سر گیری اجرای فرآیندها
- کنترل همروندی فرآیندها
- ارائه مکانیزم مناسب جهت اداره بن بست

### ✓ مدیر حافظه

- مشخص کردن بخش های آزاد و در حال استفاده حافظه
- تخصیص و بازپس گیری حافظه به/از فرآیندها
- تبدیل و نگاشت آدرس ها





## اجزای اصلی سیستم عامل

### ✓ مدیر سیستم ورودی/خروجی

- مخفی کردن جزئیات دستگاه های ورودی/خروجی از دید کاربران
- فراهم کردن واسط های مورد نیاز برای استفاده دستگاههای ورودی/خروجی
- سعی در جبران کندی دستگاههای ورودی/خروجی نسبت به حافظه و پردازنده

### ✓ مدیر حافظه جانبی

- مدیریت و حسابرسی فضاهای تخصیص یافته و آزاد
- مدیریت و تبدیل آدرس ها
- زمانبندی رسانه



## ساختارهای سیستم عامل

- ساختار یکپارچه
- ساختار لایه ای
- ساختار ماشین مجازی
- ساختار مشتری-سرویس دهنده

• رجوع شود به تمرین شماره ۵



## وقفه ها

✓ راهکاری برای قطع اجرای دستورالعمل ها توسط پردازنده برای اجرای موقت سرویس های دیگر

### ○ وقفه های خارجی

- صدور توسط دستگاههای خارجی مانند ورودی/خروجی

### ○ وقفه های داخلی

- صدور در داخل برنامه ها و توسط دستورات برنامه (مانند خطای تقسیم بر صفر، خطای دسترسی غیرمجاز حافظه)

### ○ وقفه های نرم افزاری

- یا همان System Call
- واسط بین فرآیندها و سیستم عامل (نوشته شده توسط طراحان سیستم عامل)
- دستورات فراخوانی توابع سیستمی (مانند درخواست یک ورودی/خروجی)



## وقفه ها

## ✓ برخی از وقفه های رایج

## ○ وقفه تایمر

- فایر شدن تایمرهای سیستم عامل (مثلا برای زمانبندی یا چک کردن وضعیتی خاص)

## ○ وقفه ورودی/خروجی

- صادر شده توسط دستگاه ورودی/خروجی هنگام تکمیل کار یا بروز خطا

## ○ وقفه نقص سخت افزار

- در اثر وقوع خرابی یا خطا در سخت افزار

## ○ وقفه Restart

- در اثر فشردن دکمه Restart

## ○ وقفه Program Check

- در اثر بروز خطا در برنامه ها (مانند تقسیم بر صفر، دسترسی به حافظه غیر مجاز)



## نهانی کردن

### ✓ فلسفه نهانی کردن (Caching)

- عدم تطابق سرعت حافظه و پردازنده
- محدود شدن نرخ اجرای دستورات توسط پردازنده به سرعت حافظه ☹️
- حالت ایده آل: ساخت حافظه اصلی با فناوری ثبات ها (هزینه گزاف!!!!)
- راه کار منطقی: استفاده از یک حافظه کوچکتر اما سریع بین پردازنده و حافظه اصلی
- روال کار:
- انتقال داده های مورد استفاده به صورت موقت به حافظه نهان
- در صورت نیاز به یک داده ابتدا حافظه نهان جستجو می شود
- اگر داده در حافظه نهان وجود نداشت، از حافظه اصلی خوانده می شود.
- در این حالت یک کپی از داده واکنشی شده، در حافظه نهان قرار داده می شود، با این فرض که احتمالاً در آینده نزدیک دوباره به آن نیاز داریم



## بافرینگ

### ✓ فلسفه بافرینگ

- استفاده از یک حافظهٔ میانگیر (بافر) جهت کاهش تاخیر ورودی/خروجی
- روال کار: ذخیرهٔ داده ها، قبل از نیاز در یک حافظهٔ میانی
- نتیجه: دسترسی به داده ها با سرعت بالاتر هنگام نیاز و افزایش کارایی.
- می توان گفت: با بافرینگ اختلاف بین سرعت عرضه و تقاضا یکنواخت می شود.
- استفاده از دو (یا چند) بافر جهت کاهش تاخیر و افزایش کارایی



## تمرین ها

✓ تمرین ۱: در مورد مکانیزم وقفه ورودی/خروجی تحقیق کنید.

○ راهنمایی: روال و مجموعه عملیاتی که هنگام وقوع وقفه تکمیل ورودی/خروجی صورت می گیرد را بررسی کنید.

✓ تمرین ۲: انواع روش های انتقال ورودی خروجی را بررسی کند.

○ راهنمایی: مرجع [3] بخش هفت از فصل اول. شامل روش های مبتنی بر وقفه، برنامه سازی شده و مولفه DMA

✓ تمرین ۳: در مورد دسته بندی سیستم های چند پردازنده ای تحقیق کنید.

○ راهنمایی: در کتاب استالینگز [3] این سیستم ها به چهار رده SISD، SIMD، MISD و MIMD تقسیم می شوند.



## تمرین ها

✓ تمرین ۴: در مورد دسته بندی انواع رجیسترهای پردازنده ها تحقیق کنید.

○ منبع: آزاد.

✓ تمرین ۵: در مورد انواع ساختارهای سیستم عامل ها تحقیق کنید.

○ راهنمایی: در مرجع [1] به چهار دسته زیر تقسیم شده است:

Client Server – Virtual Machine – Layered – Monolithic

✓ تمرین ۶: در مورد عملیات مود دوگانه (مود کاربر- مود کرنل) و مزایای آن تحقیق کنید. چه نوع دستوراتی باید در مود کرنل اجرا شوند؟ با مثال تحلیل کنید.

○ منبع: کتاب مرجع [1]، بخش ۱-۵-۱.





## تمرین ها

✓ تمرین ۷: در مورد وظایف کلی سیستم عامل پیرامون مدیریت فرآیند تحقیق کنید.

○ راهنمایی: منبع: کتاب مرجع [1]، بخش ۶-۱.

✓ تمرین ۸: دربارهٔ وقفهٔ تایمر، مزایا و کاربردهای آن در سیستم عامل تحقیق کنید.

○ راهنمایی: منبع [1]، بخش 1-5-1

✓ تمرین ۹: در مورد کاربرد بافرینگ در علوم کامپیوتری تحقیق کنید.  
○ راهنمایی: مثلاً کاربردهای مختلف در شبکه یا دیگر زمینه ها...



## تمرین ها

✓ تمرین ۱۰: در مورد فراخوانی های سیستمی (System Calls) تحقیق کنید.

○ مرجع [1]، بخش 2-3

✓ تمرین ۱۱: درباره روال کلی بررسی و پردازش وقفه ها تحقیق کنید.

○ مرجع [3]، بخش ۴ از فصل اول.

✓ تمرین ۱۲: در مورد PSW(Program Status Word) در سیستم عامل ها تحقیق کنید.

○ مرجع [2]، صفحه ۲۱.

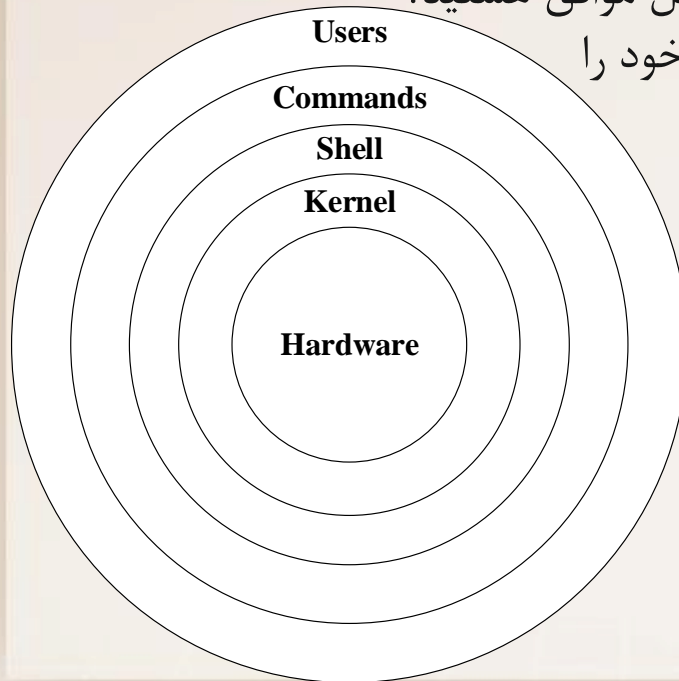


## تمرین ها

✓ تمرین ۱۲: با توجه به شکل زیر، به سوالات پاسخ دهید.

- الف، تک تک اجزای موجود در شکل را تشریح کنید.
- ب، به نظر شما در این شکل جایگاه اجزایی مانند برنامه های کاربردی، برنامه های سیستمی و فراخوانی های سیستم (System Calls) کجا هستند؟
- ج، آیا با این شما در مورد معماری سیستم عامل موافق هستید؟  
در صورت مخالفت، طرح شماتیک پیشنهادی خود را  
(با هر اجزایی) رسم کنید.

○ مرجع: آزاد.





## تمرین ها

✓ تمرین ۱۳: مفهوم API در سیستم عامل را به طور دقیق تشریح کنید. چند نمونه را با کاربرد آن مثال بزنید.  
○ منبع: [1] صفحه ۶۳.

✓ تمرین ۱۴: در سیستم عامل ها، Bootstap Program چیست؟  
○ منبع: مرجع [1]، بخش 1-2-1

✓ تمرین ۱۵: تفاوت کلاسترینگ متقارن و نامتقارن در چیست؟  
○ منبع، مرجع [1]، بخش 1-3-1



## تمرین ها

✓ تمرین ۱۶: پس از مرور فصل اول و دوم دوم کتاب سیلبرشاتس، به انتخاب خود دو سوال مفهومی را به عنوان تمرین مطرح کنید و پاسخ تشریحی آنها را بنویسید.

✓ تمرین ۱۷: پس از مرور فصل اول کتاب سیستم عامل های مدرن تنن بام، به انتخاب خود دو سوال مفهومی را به عنوان تمرین مطرح کرده و پاسخ آنها را بنویسید.

✓ تمرین ۱۸: پس از مرور فصل اول و دوم کتاب استالینگز، به انتخاب خود دو سوال مفهومی را به عنوان تمرین مطرح کرده و پاسخ آنها را بنویسید.



## نمونه تست های کنکور ارشد

✓ ارشد، دولتی، فناوری اطلاعات، ۸۹

○ کدام یک موارد زیر، نادرست است؟

- تغییر اولویت فرآیندها در مود کرنل انجام می شود.
- کامپیوتر در هنگام روشن شدن در مود کرنل قرار می گیرد.
- خواندن ساعت سیستم در مود کاربر انجام می شود.
- تنظیم زمان سیستم در مود کاربر انجام می شود.

✓ ارشد، آزاد، فناوری اطلاعات، ۸۶

○ کدام یک از مزایای سیستم عامل ماشین مجازی نیست؟

- سرعت بالا
- افزایش امنیت سیستم
- حل مشکل ناسازگاری سیستم
- استفاده چندین کاربر از یک سیستم



## نمونه تست های کنکور ارشد

✓ ارشد، کامپیوتر، آزاد، ۷۱

○ در یک سیستم عامل توزیع شده، کدام یک از موارد زیر درست نیست؟

- چندین پردازنده مستقل از نظر جغرافیایی با هم فاصله دارند و تحت یک سیستم عامل کار می کنند.
- در تبادل پیغام، کاربران می بایست آدرس ماشین های یکدیگر را بدانند.
- محل استقرار فایل ها در کنترل کاربران نیست.
- قابلیت اطمینان یک سیستم عامل توزیع شده از یک سیستم عامل متمرکز بیشتر است.

✓ ارشد، کامپیوتر، آزاد، ۸۲

○ کدام یک از عملیات زیر از تله (Trap) سیستم عامل استفاده نمی کند؟

- فراخوان سیستمی
- دسترسی غیر مجاز به حافظه
- مقدار دهی متغیرها
- نقص صفحه



## منابع

- [1]. A. Silberschatz, P. B. Galvin and G. Gagne, “**Operating System Concepts,**” 9<sup>th</sup> ed., John Wiley Inc., 2013.
- [2] A. S. Tanenbaum and H. Bos, “**Modern Operating Systems,**” 4<sup>rd</sup> ed., Pearson, 2015.
- [3] W. Stallings, “**Operating Systems,**” 9<sup>th</sup> ed., Pearson, 2018.
- [4] A. S. Tanenbaum, A. S. Woodhull, “**Operating Systems Design and Implementation,**” 3<sup>rd</sup> ed., Pearson, 2006.
- [5] نستوه طاهری جوان و محسن طورانی، “اصول و مفاهیم سیستم عامل”، انتشارات موسسه آموزش عالی پارسه، ۱۳۸۶.





# پایان