

システムプログラム概論

OS の役割と構成

第1講: 平成20年10月6日 (月) 1限 S1教室

中村 嘉隆(なかむら よしたか)
奈良先端科学技術大学院大学 助教
y-nakamr@is.naist.jp
<http://narayama.naist.jp/~y-nakamr/>

講義概要(中村担当分)

- 内容
 - オペレーティングシステム(OS)
- 参考書
 - A.S. Tanenbaum, Modern Operating System second edition, Prentice Hall, 2001.
- 講義情報, 講義資料
 - <http://narayama.naist.jp/~y-nakamr/lecture/systemsprograming2008/>
 - 資料は各自でダウンロード・印刷すること

2008/10/6

第1講 OSの役割と構成

2

講義スケジュール

- 10/6 OS の役割と構成…(参考書 1 章)
- 10/8 プロセス…(2 章)
- 10/9 スレッド…(2 章)
- 10/15 メモリ管理(1)…(4 章)
- 10/20 メモリ管理(2)…(4 章)
- 10/22 入出力制御とファイルシステム…(5~6 章)
- 10/27 試験

2008/10/6

第1講 OSの役割と構成

3

計算機システムの構成

- 計算機システムは以下の階層構造になっている
 - アプリケーション(応用)プログラム
 - システムプログラム
 - ハードウェア



2008/10/6

第1講 OSの役割と構成

4

オペレーティングシステムとは

- オペレーティングシステム(Operating System, OS)は、コンピュータにおいて、ハードウェアを抽象化したインターフェースをアプリケーションソフトウェアに提供するソフトウェアであり、基本ソフトウェアの一種である。

出典: フリー百科事典『ウィキペディア(Wikipedia)』

2008/10/6

第1講 OSの役割と構成

5

OS の役割

- ハードウェアの抽象化
 - ハードウェアを操作するための複雑な手続きを隠し、ユーザ、アプリケーションプログラムに対して、使い勝手の良いインターフェースを提供
- 資源(CPU, メモリ, ディスク, etc)の管理
 - 複数のプログラムが動作している時に、各プログラムが各資源を使う**時間**や**量**を調整
 - 計算機全体の利用効率向上のための資源割当法

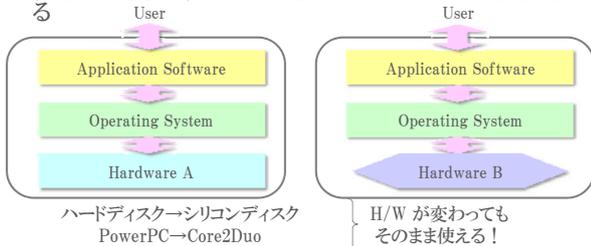
2008/10/6

第1講 OSの役割と構成

6

ハードウェアの抽象化

- ▶ ハードウェアごとの細かい違いにとらわれず、アプリケーションから同じように利用できる
- ▶ I/O レジスタに対する読み書きやタイミングなどの細かな操作を気にせず、単純化されたインターフェースで利用できる

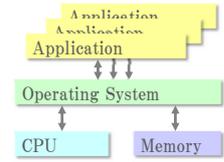


2008/10/6 第1講 OSの役割と構成 7

資源の管理

- ▶ ハードウェア資源は有限→競合が発生
 - ▶ メモリ, プロセッサ時間, 入出力装置, etc.
 - ▶ 複数のアプリケーションがこれら資源を取り合う
- ▶ ハードウェア資源の適切な割り当てが必要
 - ▶ メモリ領域分割, プロセッサの時分割, etc.

▶ プロセッサが複数あるように見せかけたり, メモリがより多くあるように見せかけることも可能



2008/10/6 第1講 OSの役割と構成 8

コンピュータの利用効率の向上

- ▶ 複数のアプリケーションを動作させるとき、資源を割り当てる順番の工夫で性能向上

➡ スケジューリング

▶ スケジューラ

- ▶ 資源割り当てを行う機構
- ▶ スケジューラの善し悪しがスループットに大きく影響
- ▶ スケジューリングアルゴリズムの例
 - ▶ FIFO (FCFS), LIFO, ラウンドロビン, レートモニタック, EDF (early deadline first)

2008/10/6 第1講 OSの役割と構成 9

OS の歴史

- ▶ 第1世代計算機 1945 - 1955
 - ▶ vacuum tubes, plug boards
- ▶ 第2世代計算機 1955 - 1965
 - ▶ transistors, batch systems
- ▶ 第3世代計算機 1965 - 1980
 - ▶ ICs and multiprogramming
- ▶ 第4世代計算機 1980 - 現在
 - ▶ personal computers

2008/10/6 第1講 OSの役割と構成 10

第1世代:真空管&プラグボード

- ▶ 真空管等で構成, OS は存在せず
- ▶ プログラムは機械語(プラグボード)で与える
- ▶ プログラム実行の間, マシン全体が専有される

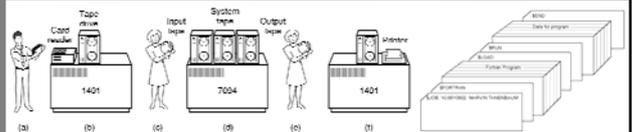


ENIAC (1946年)
1746本の真空管等で構成
出典: フリー百科事典『ウィキペディア (Wikipedia)』

マシンが空いている時間を見計らってプログラムを実行しなければならない

2008/10/6 第1講 OSの役割と構成 11

第2世代:トランジスタ&バッチシステム



▶ バッチシステム

- 一定時間、ジョブ(パンチカードで入力)をユーザから集める
 - 集まった全てのジョブをテープに書き込み
 - テープを読み込ませて、集めたジョブを一気に計算し、計算結果をテープに出力
 - テープから各ジョブの計算結果を取り出し、パンチカードにコピー
- ※(c)-(e)の機能は現在のOSに該当

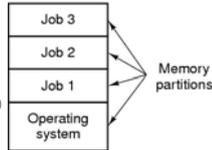
ジョブの例

ジョブはいつでも投入できる
ただし、結果が出るのは遅い

2008/10/6 第1講 OSの役割と構成 12

第3世代:IC & マルチプログラミング

- ▶ マルチプログラミング可能なOSが登場
 - ▶ 第2世代では、ジョブを順次処理→入出力待ちが発生すると、CPU時間が無駄になっていた
 - ▶ 第3世代では、マルチプログラミングが可能になった
 - ▶ 複数プログラムをメモリに読み込んでおき、入出力待ちが発生すると、その時間を他のジョブに切り替えて実行→CPU利用率が向上
 - ▶ 計算機への入力、出力にスプーリングが使えるようになった
 - ▶ バッチ処理だと結果が遅い→時分割処理



- ▶ 時分割処理を行うOSが登場
 - ▶ CTSS by MIT (1962年)
 - ▶ MULTICS project by MIT, Bell, GE (1960年代)
 - ▶ 数百人のユーザを同時にサポートすることを目標
 - ▶ UNIXに発展

2008/10/6

第1講 OSの役割と構成

13

第4世代:パーソナルコンピュータ

- ▶ パーソナルコンピュータ(PC)の登場
 - ▶ 最初の8ビット汎用CPUの登場(Intel 8080, 1974年)
- ▶ PC用OS
 - ▶ CP/M(1970年代)
 - ▶ DOS/MS-DOS(1980年代) } シングルタスク OS
 - ▶ GUI(Graphical User Interface)の登場
 - ▶ Mac OS(Apple Macintosh)
 - ▶ MS Windows, OS/2
 - ▶ CPUの高性能化に伴い、OSも高度化(マルチタスク, etc)
 - ▶ Windows NT → 2000 → Windows XP → Windows Vista
- ▶ UNIX + X Window
 - ▶ ワークステーション向け → …… → PCでも利用可能に
 - ▶ Linux

2008/10/6

第1講 OSの役割と構成

14

OSの基本機能

- ▶ プロセス管理
- ▶ メモリ管理
- ▶ 入出力制御(I/O)
- ▶ ファイルシステム
- ▶ セキュリティ
- ▶ 上記のOSの機能をアプリケーションプログラムから使うインターフェースの集合をAPI(Application Program Interface)と呼ぶ

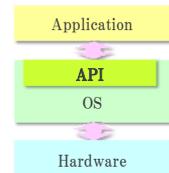
2008/10/6

第1講 OSの役割と構成

15

API(Application Program Interface)

- ▶ OSが管理する資源を利用するためのプログラムインターフェース(命令や関数の集合)
- ▶ APIに含まれるもの
 - ▶ OSが提供するシステムコール
 - ▶ OSが提供するGUI操作用ライブラリなど
- ▶ 標準化されたAPIを使うことでOSに依存せず、容易にアプリケーションを実装できる



File management	
Call	Description
fd = open(file, how, ...)	Open a file for reading, writing or both
s = close(fd)	Close an open file
n = read(fd, buffer, nbytes)	Read data from a file into a buffer
n = write(fd, buffer, nbytes)	Write data from a buffer into a file
position = lseek(fd, offset, whence)	Move the file pointer
s = stat(name, &buf)	Get a file's status information

2008/10/6

第1講 OSの役割と構成

16

プロセス管理の概要

- ▶ マルチプログラミング
 - ▶ 複数のプログラムを同時に実行する仕組み
- ▶ 複数のプロセス(実行中プログラム)の管理
 - ▶ プロセスの生成, 終了, 切り替え
- ▶ スレッド(並列実行処理の単位)の管理
 - ▶ プロセスとの違い?
 - ▶ スレッドの作成, 終了, 切り替え
 - ▶ 資源競合の回避方法

✓ 詳細は、第2～3回講義にて

2008/10/6

第1講 OSの役割と構成

17

メモリ管理の概要

- ▶ プロセスへのメモリの割当管理
 - ▶ 競合回避
- ▶ 仮想メモリと実メモリ
 - ▶ アドレス変換, MMU
 - ▶ ページング
 - ▶ セグメンテーション
 - ▶ ページ切り替えアルゴリズム

✓ 詳細は、第4～5回講義にて

2008/10/6

第1講 OSの役割と構成

18

入出力制御(I/O)

- ▶ 様々な入出力(I/O)デバイス
 - ▶ キーボード, マウス, プリンタ, ディスク, ビデオ, カメラ, ネットワーク, センサ, ...
 - ▶ キャラクタデバイスとブロックデバイス
- ▶ プロセッサとデバイス間のやりとり
 - ▶ Memory-mapped I/O, DMA
 - ▶ 割込み制御
- ▶ デバイスドライバの構成

✓詳細は, 第 6 回講義にて

2008/10/6

第1講 OSの役割と構成

19

ファイルシステム

- ▶ ファイルシステムの構成
 - ▶ ファイル, ディレクトリの構成
- ▶ ファイルへのディスクブロックの割付方法
 - ▶ 連続割付, リンク割付, i-node

✓詳細は, 第 6 回講義にて

2008/10/6

第1講 OSの役割と構成

20

セキュリティ

- ▶ ユーザが資源に何らかのアクセスするときに前もって認証
- ▶ ユーザに適切なアクセスレベルを付与
- ▶ 管理者の方針に基づいてアクセスを制限
- ▶ Ex. UNIX でのファイルパーミッションによる制御

✓詳細は, 参考書 9 章参照

2008/10/6

第1講 OSの役割と構成

21

様々な OS

- ▶ シングルタスク OS とマルチタスク OS
- ▶ リアルタイム OS

2008/10/6

第1講 OSの役割と構成

22

マルチタスク OS

- ▶ OS が複数のプロセスを切り替えながら実行
 - ▶ 同時に 1 つしか実行できない = シングルタスク
 - ▶ 十分短い時間間隔で切り替えができると, 複数のタスクが同時に動作しているように見える
- ▶ プリエンプティブ(preemptive)スケジューリング
 - ▶ 一定の時間が経過すると, 処理実行の途中で別のプロセスに CPU を切り替える
- ▶ ハンプリエンプティブスケジューリング
 - ▶ プロセス自身が処理終了後, 自発的に CPU 切替を要請する
 - ▶ OS は CPU 資源の管理をしなくていい

2008/10/6

第1講 OSの役割と構成

23

リアルタイム OS

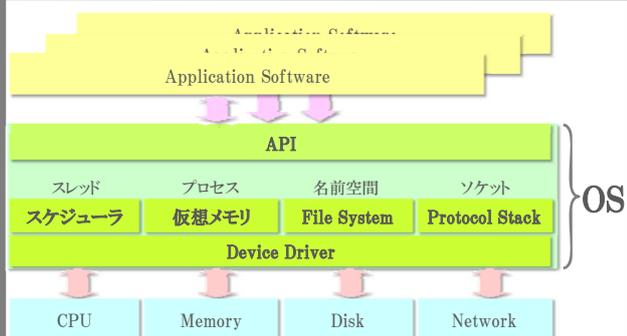
- ▶ リアルタイム = 実時間, 即時性
 - ▶ 応答が早いことではなく, 決められた時間を守ること
- ▶ 資源管理のうち, 時間資源の保護に特化
 - ▶ 各プロセスの優先度を元にスケジューリング
 - ▶ スケジューリングが肝要
- ▶ 組み込みシステムなどで利用
 - ▶ Ex. μ ITRON (国産 OS, 台数ベースでは日本で最も普及している OS)

2008/10/6

第1講 OSの役割と構成

24

OS の構成



2008/10/6

第1講 OSの役割と構成

25

まとめ

- OS の目的・役割
 - ハードウェアの抽象化, 資源の管理
- OS の歴史
 - 第 1 世代計算機～第 4 世代計算機で使われた OS
- OS の機能
 - プロセス・メモリ管理, 入出力制御, ファイルシステム, セキュリティ
- OS の種類
- OS の構成

2008/10/6

第1講 OSの役割と構成

26