

# ICT リテラシー (情報技術論) A

-- 第 12 回：オペレーティングシステム --

栗野 俊一

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く  
禁じます

2025/12/15 ICT リテラシー (情報技術論) A

# 伝言

---

## 私語は慎むように !!

### □ 席は自由です

- できるだけ前に詰めよう
- コロナ対策のために、ソーシャルディスタンスをたもとう

### □ 色々なお知らせについて

- 栗野の Web Page に注意する事

<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino>

- google で「kurino」で検索

# 前回 (第 11 回) の復習

---

ICT リテラシー (情報技術論) A

# 前回 (第 11 回) の復習

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 前回 (第 11 回) の復習 (1)

---

## □ 前回 (第 11 回) の復習 (1) : コンピュータの基本構造

### ○ コンピュータの歴史 : 数の表現手段と実現手段の変化

- ▶ 電子化 : 数値を「電子操作」で表現 -> 速度が桁違い改良される
- ▶ ノイマン型 : プログラムとデータがメモリ上に -> 万能性の獲得
- ▶ 半導体の導入 : 更に、小型化、高速化、安定化を得る

### ○ コンピュータの種類 : 規模と用途

- ▶ コンピュータは色々な用途で用いられている

### ○ コンピュータの機能 : 演算, 制御, 記憶, 入力, 出力

### ○ パソコンの内部構成 : M/B, CPU, Main Memory

### ○ 記憶装置 : 情報を記録する装置 (キャッシュ/メモリ/ハードディスク)

- ▶ 速度と容量のトレードオフ (cf. パレートの法則)

### ○ 演算装置 : 演算機能と制御機能を持つ

# 前回 (第 11 回) の復習 (2)

---

## □ 前回 (第 11 回) の復習 (2) : コンピュータの動作原理

### ○ チューリングマシン (TM) : 「計算」の数学的なモデル

- ▶ (無限の)記憶装置(テープ)と、簡単な機構 \*だけ\* で「計算」ができる
- ▶ 万能 TM の存在 -> ノイマン型

### ○ 電気回路とスイッチ : 「電気」を利用した「計算」の表現

- ▶ 真空管/トランジスタ : 増幅装置(アナログ)から、スイッチ(デジタル)へ

### ○ 二進法と論理回路

- ▶ 二進法 : On/Off による数表現 ( On -> 1 / Off -> 0 )
- ▶ 回路による、論理演算 ( スイッチを利用して、スイッチができる [自己言及] )
- ▶ 2の補数 / 丸め誤差 ( 10 進数 -> 2 進数変換 )

### ○ 演算の組み合わせによる複雑な機能の実現

- ▶ LSI 化による、集積度の飛躍的な向上 (印刷による微細化)

# 今週 (第 12 回) の概要

---

ICT リテラシー (情報技術論) A

## 今週 (第 12 回) の概要

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 今週 (第 12 回) の予定

---

## □ 今週 (第 12 回) の予定

### ○ 講義：オペレーティングシステム

- ▷ ソフトウェア (Text p.73, 6 章)
- ▷ OS とは (Text p.73 6.1.1 節)
- ▷ OS の種類 (Text p.74 6.1.2 節)
- ▷ OS の機能 (Text p.75 6.1.3 節)

# 今週 (第 12 回) の目標

---

## □ 今週 (第 12 回) の目標

### ○ オペレーティングシステム(OS)の機能

- ▷ OS の必要性

- ▷ OS の役割

### ○ ビジネスにおける OS の位置づけ

- ▷ デファクト・スタンダード : ビジネス戦略における OS 支配の意味



# 今週 (第 12 回)

---

## □ 前回 (第 11 回) の課題

- 振り返り課題-11

- 小テスト-11

## □ 今週 (第 12 回) の課題

- 振り返り課題-12

  - ▶ 回答期限は、講義実施から 1 week

- 小テスト-12

# ソフトウェア

---

## ICT リテラシー (情報技術論) A

# ソフトウェア

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# ソフトウェア

---

## □ソフトウェア ( Text p.73, 6 章 )

### ○ソフトウェアの定義

- ▶ 広義 : 情報の形で記述されたサービス ( 楽曲、映画、テレビ番組、ホームページなどのコンテンツである創作物 )
- ▶ 狭義 ( コンピュータ ) : CPU への命令を記述するプログラム

## □ハードウェアとソフトウェア

○ハード、ソフトが無ければただの箱 / ソフト、ハードが無ければただの紙

○ハードウェアだけあれば、サービスの提供は可能 ( 専門装置 )

- ▶ 柔軟性 ( 汎用性 / 万能性 ) に欠ける

○ソフトウェアは、ハードウェアの指示情報 ( ハード無しでは、本当に無意味 )

- ▶ ソフトウェアの存在は、システム(ハード+ソフト) に柔軟性を提供する
- ▶ コンピュータの万能性は、「ソフトの交換」で実現 : 万能 TM

## □ソフトウェアの必要性 : 機能性を高める仕組み

○商品を「組み合わせ可能な基本機能」を持つハードと、「組み合わせ事例」であるソフトで構成する

- ▶ 音楽 : オルゴール(完全な自動演奏) / ピアノ(楽譜[ソフト]と演奏が必要)
- ▶ 料理 : ピーラー(簡単に皮剥き可能) / 包丁(技術[ソフト]が必要)

# ソフトウェアの分類

---

- ソフトウェアの分類とそれらの関係 (Text p.74, 図 6.1)
- アプリケーションソフト：コンピュータを使って何か(応用)をするソフト
  - 必要に応じて、(外部)補助記憶装置から、主記憶装置に読み込まれる
- システムソフト：アプリケーションソフトを動かすための環境となる (常駐)
  - ミドルウェア：アプリケーションソフトに基本的な機能を実現
  - プラットフォーム：アプリケーションソフトが動く土台 (広義の OS)
    - ▷OS：ハードの管理を行う (ハードと独立 [狭義の OS])
    - ▷BIOS：ハード操作の基本機能の実現 (ハードに依存：ファームウェア)
    - ▷ドライバ ( OS と周辺装置を仲立するソフト )

# オペレーティングシステム

---

ICT リテラシー (情報技術論) A

## オペレーティングシステム

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# オペレーティングシステム

---

## □ オペレーティングシステム [OS] (Text p.73, 6.1 節)

### ○ OS とは (Text p. 6.1.1 節) [What]

▷ 定義 : コンピュータのオペレーション(操作・運用・運転)を司るシステムソフトウェア(wikipedia)

### ○ OS を利用する目的 [Why]

▷ ハードウェアの利用効率の向上 (cf. Fortran monitor)

### ○ OS が提供する機能 [How to]

▷ ハードウェアの抽象化 (異なるものを同じものに見せる仕組み:質的相違)

▷ ハードウェア資源の管理 (有限な資源を適切に提供する仕組み:量的相違)

### ○ OS の位置づけ [Where/When]

▷ アプリケーションプラットフォーム ( アプリのベース : アプリは OS 毎に開発 )

▷ ユーザインターフェースの提供 ( 人間とコンピュータの仲立 [MMI/GUI] )

# OS の種類

---

## □ OS の種類 (Text p. 6.1.1 節)

- PC 以外の OS : ハードがあればソフトがあり、OS もある

- ▷ ゲーム機 / 乗物(車/飛行機/..) / 家電 / ガラ携 / etc...

- PC (スマホ/ タブレット) 用の OS

- ▷ unix 系 : Liux (android), MacOS (iOS)

- ▷ MS-Windows 系 : Windows 11 (Windows 10 Mobile)

## □ unix / MS-Windows

- 歴史 : 色々な人の手で改良がされた / MS 社が開発

- 公開性 : Open (誰でも自由に参照でき変更可能) / Close

- 保証性 : 無保証(自己責任) / 企業サポートあり

- 価格 : 無料 / 有料

# OS の機能

---

## □ OS の機能 (Text p.75 6.1.3 節)

### ○ ユーザインタフェース：コンピュータの操作性を定める

- ▶ マウスやキーボードの操作の仕方/ウィンドウのデザイン/ファイルの扱い方

### ○ ソフトとハードの仲介：ハードウェアの相違を吸収する

- ▶ OS の介在によって設計方法が共通化
- ▶ OS が定めた基準を満たしている限り、どのメーカーのソフト、ハードの組合せでも動く ( $M \times N \rightarrow M + N$ )

### ○ 記憶管理：キャッシュ/メモリ/ハードディスク等を管理 (速度の差)

- ▶ バッファリング：速度の違うメモリ間の速度の差を緩衝の役割を担う
- ▶ スワップ：一時的に情報を回避する仕組み

### ○ プロセス管理：CPU の管理

- ▶ タイムシェアリング：時分割により、複数の作業が同時に動いているように見せる
- ▶ マルチタスク：複数の仕事を効率良く行うよう、リソースを割り当てる (スケジューリング/割り込み)

### ○ ユーザ管理：利用者の管理

- ▶ マルチユーザシステム：複数の人が同じ PC を共有できる仕組み



# ビジネスにおける OS の位置づけ

---

ICT リテラシー (情報技術論) A

## ビジネスにおける OS の位置づけ

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# Microsoft (MS) 社の軌跡

---

□ 何故、MS はこんなに巨大企業になれたか？

- MS Windows を開発し、PC 上の OS を支配し、世界を席卷

□ Microsoft (MS) 社の軌跡

- Basic 処理系を開発して、「ライセンス」契約した

  - ▶ ソフトは「売り物」から、「貸す物」になった

  - ▶ コピーできる(情報の最も重要な性質の)権利は、売らない

- PC DOS (を他社から購入して.. MS DOS として) IBM にライセンス契約

  - ▶ IBM PC の興隆と同時に、OS の覇者としての位置づけを確立 (BIOS)

  - ▶ IBM 社は、互換機メーカーに、追い落されたのと対照的

- MS-Windows を開発 / MS-Office の売上で、世界一のメーカーに..

# ビジネスにおける OS の位置づけ

---

## □ビジネスにおける OS の位置づけ

- ユーザが使うのは OS ではなく、アプリケーションソフト
- OS はソフトウェアプラットフォーム
  - ▶ソフトウェア開発で、プラットフォームを握れば、開発に先行できる
- MS-Windows と MS-Office の関係
  - ▶MS 社の主力製品は、MS-Office

## □ビジネス戦略における OS 支配の意味

- 開発上の先行/有利な状況
- 売りたいアプリのプラットフォーム
  - ▶MS-Windows がデファクトスタンダードである事に価値

# デファクトスタンダード

---

## □ デファクトスタンダード (de facto standard)

- (市場における)結果として事実上標準化した基準 (標準化機関等が定めた規格ではない)

- ▷ 例: MS-Windows, MS-Office, PC/AT互換機, TCP/IP, TeX, QWERTY キーボード, VHS, etc..

## □ 規格の重要性

- 開発商品に規格を設ける事により、消費者に利益を与える(例:螺子)

- ▷ 企業にとっては、制約(規格を守る)だが、商機上のチャンス

## □ デファクトスタンダードの価値

- 企業が自分の都合で、規格を設ける事ができる

- ▷ 自社は制約無しで、他社に制約を与える事ができる

- ▷ 場合によっては、特許料/ライセンス料の徴収も.. (cf 任天堂)

おしまい

---

## ICT リテラシー (情報技術論) A

おしまい

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます