

■ コンピュータサイエンス教科書シリーズ 7

オペレーティングシステム

理学博士 大澤 範高 著

COMPUTER SCIENCE TEXTBOOK SERIES

コロナ社

コンピュータサイエンス教科書シリーズ編集委員会

編集委員長 曾和 将容 (電気通信大学)

編集委員 岩田 彰 (名古屋工業大学)

(五十音順) 富田 悦次 (電気通信大学)

(2007年5月現在)

刊行のことは

インターネットやコンピュータなしでは一日も過ごせないサイバースペースの時代に突入している。また、日本の近隣諸国も IT 関連で急速に発展しつつあり、これらの人たちと手を携えて、つぎの時代を積極的に切り開く、本質を深く理解した人材を育てる必要に迫られている。一方では、少子化時代を迎え、大学などに入学する学生の気質も大きく変わりつつある。

以上の状況にかんがみ、わかりやすく体系化された、また質の高い IT 時代にふさわしい情報関連学科の教科書と、情報の専門家から見た文系や理工系学生を対象とした情報リテラシーの教科書を作ることを試みた。

本シリーズはつぎのような編集方針によって作られている。

- (1) 情報処理学会「コンピュータサイエンス教育カリキュラム」の報告、ACM Computing Curricula Recommendations を基本として、ネットワーク系の内容を充実し、現代にふさわしい内容にする。
- (2) 大学理工系学部情報系の 2 年から 3 年の学生を中心にして、高専などの情報と名の付くすべての専門学科はもちろんのこと、工学系学科に学ぶ学生が理解できるような内容にする。
- (3) コンピュータサイエンスの教科書シリーズであることを意識して、全体のハーモニーを大切にするとともに、単独の教科書としても使える内容とする。
- (4) 本シリーズでコンピュータサイエンスの教育を完遂できるようにする。ただし、巻数の制限から、プログラミング、データベース、ソフトウェア工学、画像情報処理、パターン認識、コンピュータグラフィックス、自然言語処理、論理設計、集積回路などの教科書を用意していない。これらはすでに出版されている他の著書を利用していただきたい。

ii 刊 行 の こ と ば

- (5) 本シリーズのうち「情報リテラシー」はその役割にかんがみ、情報系だけでなく文系、理工系など多様な専門の学生に、正しいコンピュータの知識を持ったうえでワープロなどのアプリケーションを使いこなす、なおかつ、プログラミングをしながらアプリケーションを使いこなせる学生を養成するための教科書として構成する。

本シリーズの執筆方針は以下のようである。

- (1) 最近の学生の気質をかんがみ、わかりやすく、丁寧に、体系的に表現する。ただし、内容のレベルを下げることはしない。
- (2) 基本原理を中心に体系的に記述し、現実社会との関連を明らかにすることにも配慮する。
- (3) 枝葉末節にとらわれてわかりにくくならないように考慮する。
- (4) 例題とその解答を章内に入れることによって理解を助ける。
- (5) 章末に演習問題を付けることによって理解を助ける。

本シリーズが、未来の情報社会を切り開いていけるたくましい学生を育てる一助となることができれば幸いです。

2006年5月

編集委員長 曾和 将容

まえがき

オペレーティングシステムは、アプリケーション・プログラムや利用者に対して共通的な機能を提供すると共に、ハードウェアを効率良く利用するための基盤ソフトウェアである。本書では、オペレーティングシステムを基本的な機能を提供する仮想計算機と、より高度なサービスを提供するサブシステムに分類し、その分類に基づいてオペレーティングシステムの基本的な概念と機構およびそれらの関係が理解できるよう解説している。

まず、基本的な仮想計算機の機能をプロセッサ、デバイス、メモリに分け、それぞれを解説している。高度なサービスを提供するサブシステムには多様なものがあるが、多くのオペレーティングシステムで実現されており、重要性の高いファイルシステム、ネットワーク、並列分散処理、ユーザインタフェース、保護とセキュリティを本書では取り上げた。

また、オペレーティングシステムは複雑なソフトウェアであり、複雑さを克服するために抽象化、構造化、仮想化が行われている。オペレーティングシステムは実用的なシステムであるので、厳密解を求めるのではなく、現実的な効果が得られるようトレードオフが考慮されている。これらの考え方がオペレーティングシステムにどのように繰返し適用されているかがわかるように、注意を払った。

現在のオペレーティングシステムにおいて重要な概念と機構を理解することに重点を置き、歴史的経緯にはこだわらなかった。このため、項目の選択や項目の解説順序が他の類書と異なっている点がある。重要な項目については従来のテキスト同様に解説している。また、特定のオペレーティングシステムに依存した理解を防ぐために、具体的なオペレーティングシステムについては必要最小限しか触れなかった。オペレーティングシステムの発展や具体的なオペレ

ーティングシステムの詳細については参考文献をあたらりたい。

紙幅の関係から省いた内容や詳しい参考文献、演習問題で利用するプログラム等の補足情報を Web で公開している。検索エンジンで「OS コロナ社 大澤」の 3 語で検索して、サイトを見つけ、そのサイトの内容を参照していただきたい。

本書を執筆する機会を与えていただき、原稿に対する貴重なコメントをいただいた電気通信大学大学院の曾和将容先生に深く感謝いたします。また、原稿を読んでいただき、数多くの有益なご指摘をいただいた豊橋技術科学大学の市川周一先生、独立行政法人メディア教育開発センターの浅井紀久夫先生、そして、筆者の遅筆を辛抱強く待っていただき、ひとかたならぬお世話をいただいたコロナ社の皆様に心よりお礼申し上げます。

2008 年 1 月

大澤 範高

目 次

1 オペレーティングシステムとは

1.1 重要な概念	2
1.2 コンピュータの構造	3
1.3 オペレーティングシステム構造	6
1.4 仮想計算機としてのオペレーティングシステム	7
1.4.1 仮想プロセッサ	7
1.4.2 仮想デバイス	8
1.4.3 仮想メモリ	9
1.5 サービス提供システムとしてのオペレーティングシステム	10
1.6 オペレーティングシステムの構造, 機能, 運用法	12
演習問題	13

2 実行管理

2.1 実行主体	14
2.2 入出力処理とアイドル状態	15
2.3 マルチタスキング	16
2.4 実行状態	18
2.5 実行モード	19
2.6 割 込 み	20
2.6.1 ベクタ割込みとポーリング	21
2.6.2 割込み処理の流れ	23

2.6.3	割込みレベル	24
2.6.4	タイマ割込み	24
2.7	ディスパッチャ	25
2.8	実行スケジューリング	26
2.8.1	タイミング	27
2.8.2	優先度付与基準	28
2.9	実行スケジューリング・アルゴリズム	29
2.9.1	先着順	29
2.9.2	時間順	30
2.9.3	ラウンドロビン	31
2.9.4	実時間スケジューリング	31
2.9.5	締切順	32
2.9.6	レート・モノトニック	33
	演習問題	33

3 同期・通信

3.1	同期・排他制御	34
3.1.1	相互排除の条件	35
3.1.2	割込み禁止	35
3.1.3	フラグ	36
3.1.4	不可分命令	37
3.1.5	ビジーウェイト	39
3.2	セマフォ	40
3.3	モニタ	41
3.4	優先度逆転	43
3.4.1	優先度継承アルゴリズム	45
3.4.2	優先度上限アルゴリズム	45
3.5	デッドロック	45
3.6	デッドロック解決法	47

3.6.1	防 止	47
3.6.2	回 避	47
3.6.3	回 復	48
3.6.4	楽観的同期	49
3.7	情 報 交 換	49
3.7.1	メッセージ通信	50
3.7.2	同期式・非同期式通信	51
3.7.3	メモリ上の情報交換方式	52
	演 習 問 題	54

4 デバイス管理

4.1	デ バ イ ス	55
4.1.1	デバイスコントローラ	55
4.1.2	デバイスの入出力	56
4.1.3	ダイレクトメモリアクセス	56
4.1.4	バ ス	57
4.1.5	状態変化の検出	59
4.2	デバイスの分類	60
4.3	デバイスドライバ	60
4.4	バッファリング	62
4.4.1	ダブルバッファ	62
4.4.2	バッファプール	63
4.4.3	リングバッファ	63
4.5	ブロッキング	65
4.6	スプーリング	65
4.7	ディスク装置	66
4.7.1	ボリューム	66
4.7.2	ハードディスクの構造	67
4.7.3	アクセス時間	68

4.8 ディスクアクセススケジューリング	69
4.8.1 先 着 順	70
4.8.2 最短シーク時間	70
4.8.3 SCAN	70
4.8.4 C-SCAN	71
4.8.5 LOOK と C-LOOK	71
4.8.6 ディスクアクセススケジューリング例	71
4.8.7 ハードディスク装置の発展	72
4.9 予測に基づいたアクセス高速化技法	73
4.9.1 プリフェッチ	73
4.9.2 キャッシング	73
4.10 表 示 装 置	74
4.11 プラグ・アンド・プレイ	75
演 習 問 題	76

5 記憶領域管理

5.1 ア ド レ ス	77
5.2 記憶領域の基本管理手法	78
5.2.1 可変長領域管理	78
5.2.2 固定長領域管理	80
5.2.3 ブロック管理方式	81
5.3 コンパクション	83
5.4 ガーベジコレクション	84
5.5 ページング	87
5.5.1 ページテーブル	88
5.5.2 多段ページテーブル	90
5.5.3 逆ページテーブル	91
5.5.4 アドレス変換キャッシュ	92
5.5.5 ページ共有	94

5.5.6 ページ保護	95
5.6 セグメンテーション	97
演習問題	99

6 仮想記憶

6.1 オーバーレイ	100
6.2 仮想記憶システム	102
6.3 プロセス・スワッピング	102
6.4 デマンドページング	103
6.4.1 処理の流れ	104
6.4.2 ページアウト・デーモン	106
6.5 参照の局所性	107
6.5.1 ワーキングセットモデル	107
6.5.2 ページフォールト率	108
6.6 置換えアルゴリズム	109
6.6.1 FIFO	110
6.6.2 OPT	111
6.6.3 LRU	111
6.6.4 クロック・アルゴリズム	112
6.7 スラッシング	114
演習問題	115

7 ファイルシステム

7.1 ファイル	116
7.2 ファイルシステムの機能	118
7.3 名前空間	118
7.3.1 ディレクトリ	120

7.3.2 階層ディレクトリ	121
7.3.3 ハッシュ表	122
7.4 ファイルの属性	123
7.5 ファイルの内容領域管理	124
7.5.1 FAT ファイルシステム	126
7.5.2 UNIX ファイルシステム	127
7.6 リンク	129
7.7 メモリ・マップト・ファイル	130
7.8 ログ構造ファイルシステム	131
7.9 耐故障性	132
7.9.1 ジャーナリング・ファイルシステム	133
7.9.2 耐故障性ディスク	133
演習問題	134

8 ネットワーク

8.1 データ交換モデル	135
8.1.1 回線交換モデル	135
8.1.2 パケット交換モデル	137
8.2 ネットワーク・トポロジ	138
8.2.1 代表的トポロジ	138
8.2.2 物理構造と論理構造	140
8.3 プロトコル	140
8.4 インターネットプロトコル	142
8.4.1 アドレス	143
8.4.2 カプセル化	145
8.4.3 ヘッダ構造	146
8.5 グループ通信方式	147
8.6 URI	149

演習問題	150
------	-----

9 並列分散処理

9.1 分散処理	151
9.2 クライアント・サーバ・モデル	152
9.3 遠隔手続き呼出し	153
9.3.1 データ表現	155
9.3.2 遠隔手続き呼出しセマンティクス	156
9.4 遠隔共有	157
9.4.1 デバイス遠隔共有	157
9.4.2 分散共有ファイル	158
9.4.3 分散共有メモリ	158
9.5 分散キャッシュ	159
9.5.1 キャッシュ・コヒーレンス	159
9.5.2 メッセージ通信とメモリ共有の双対性	161
9.6 並列処理	161
演習問題	162

10 ユーザインタフェース

10.1 キャラクタユーザインタフェース	163
10.2 グラフィカルユーザインタフェース	164
10.3 ウィンドウシステム	165
10.4 国際化・地域化	166
10.4.1 ロケール	167
10.4.2 文字コード	168
10.4.3 符号化方式	171
10.5 インプットメソッド	172

10.5.1	フロントエンド方式とバックエンド方式	173
10.5.2	変換エンジン	174
演習問題		175

11 保護とセキュリティ

11.1	保 護	176
11.1.1	アクセス制御行列	177
11.1.2	ケーパビリティリスト	178
11.1.3	アクセス制御リスト	179
11.1.4	ロック・キー機構	179
11.2	束 モ デ ル	179
11.3	暗 号	180
11.3.1	共通鍵暗号	181
11.3.2	公開鍵暗号	182
11.4	認 証	183
11.4.1	データ認証	183
11.4.2	ユーザ認証	184
11.4.3	公開鍵基盤	186
11.5	セキュリティ	186
11.6	悪意のあるソフトウェア	187
11.6.1	トロイの木馬	187
11.6.2	スパイウェア	187
11.6.3	コンピュータウィルス	187
11.6.4	ワ ー ム	188
11.6.5	ルートキット	188
11.7	攻 撃	189
11.7.1	ソーシャル・エンジニアリング	189
11.7.2	サービス不能	190
11.7.3	バッファオーバーラン	190

11.8 防 御	192
11.8.1 ワ ク チ ン	192
11.8.2 ネットワーク・ファイアウォール	193
11.8.3 サンドボックス	194
11.9 事後の検証	195
演習問題	195

12 構成法と事例

12.1 関心の分離	196
12.2 構成法	196
12.3 仮想計算機	198
12.3.1 ハードウェア支援による仮想計算機	199
12.3.2 ソフトウェアシミュレーションによる仮想計算機	200
12.4 運用と管理	201
12.4.1 ブートストラップ	201
12.4.2 オペレーティングシステムの初期化	202
12.4.3 オペレーティングシステムの終了	203
12.5 オペレーティングシステム事例	204
12.5.1 Linux	204
12.5.2 Plan 9	204
12.5.3 Windows	204
12.5.4 MacOS X	205
12.5.5 ITRON	205
12.5.6 特殊化された OS	205
演習問題	206

13 歴史と展望

13.1 オペレーティングシステムの歴史	207
----------------------	-----

13.2 将来展望	211
引用・参考文献	213
演習問題解答	216
索引	218

1 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) は、演算処理装置やハードディスク装置などのハードウェアを効率良く利用するための基盤ソフトウェアであり、コンピュータ利用者や表計算ソフトウェアなどの**アプリケーション・プログラム** (application program, 応用プログラム) に対して共通的な機能を提供する。図 1.1 に示すようにオペレーティングシステムは、コンピュータシステム内では、ハードウェアとアプリケーション・プログラムの中間に位置すると考えることができる。オペレーティングシステムは**カーネル** (kernel), **スーパーバイザ** (supervisor) と呼ばれることもある。以下では、誤解を生じる場合を除いて、オペレーティングシステムを**OS** と略記し、アプリケーション・プログラムを単にアプリケーションと表記する。

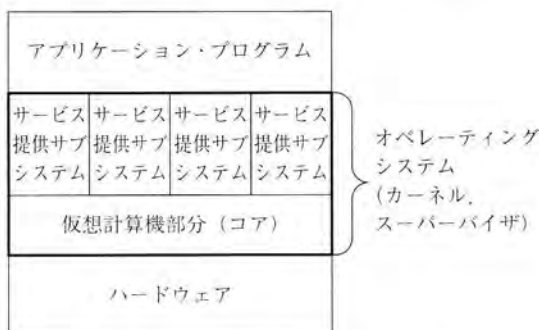


図 1.1 コンピュータシステムの構造

OS は、ハードウェアの機能を拡張し、利便性を高めた仮想コンピュータを実現するソフトウェアとも解釈できる。この観点から、OS を、ハードウェアの仮想化を行い、基本的な機能を提供する仮想計算機部分と、より高度なサービスを提供するサブシステムに分類することができる。本書では前者を**コア**

2 1. オペレーティングシステムとは

(core) と呼び、後者をサービス提供サブシステムと呼ぶ。この分類に基づいてオペレーティングシステムの基本的な概念と機構およびそれらの関係を理解できるように解説する。

1.1 重要な概念

OSを理解する上で重要な概念として、抽象化、仮想化、構造化、トレードオフがある。

抽象化 (abstraction) は、個々の具体的な機能や特性から、重要ではない性質や表象を捨て去り、共通した要素を取り出すことである。OSでは、ハードウェアの違いを隠蔽し、統一的に扱うためにハードウェア抽象化が行われる。さまざまな機能がアプリケーションからOSに求められるが、求められる機能のすべてを実現するとあまり利用されない機能が多くなり無駄が多くなるので、抽象化された重要な機能だけが提供される。アプリケーションは、その抽象化された機能を用いて、より具体的な機能を実現する。

仮想化 (virtualization) は、メモリや装置などの資源を物理的な構成に制約されることなく論理的な実体 (entity) として提供することである。物理的な演算処理装置や記憶装置の量には制限があるが、OSはこれらの制限を仮想化によって取り除き、論理的な実体として資源を提供する。

構造化 (structuralization) は、複雑な構造を整理し、より単純な要素に分割すると共にそれらの要素間の関係を明確化することである。OSは複雑なソフトウェアであるので、それを構造化によってできるだけ単純化し、見通しを良くし、システムの保守・管理を容易にすることが必要である。構造化の代表例が階層化である。階層化は、要素がなんらかの上下関係を持つように整理することである。

トレードオフ (trade-off) は、ある要求を満たそうとすると他の要求が満たせなくなる状態にあること、および、その状態で諸条件を考慮して調整を行うことをいう。コンピュータシステム全般にいえることであるが、同時にすべ

での要求を満たすことは一般に困難であり、そのような場合に要求のバランスをとり、妥協することによって、要求の折り合いを付けることが必要である。たとえば、OS で用いる **アルゴリズム** (algorithm, 処理手順) を検討する際には、性能などを向上させるために厳密な計算によって処理を行おうとすると計算に時間がかかり、そのためにかえって性能が低下してしまうことがある。そのような場合には、厳密解を求めず、近似によって高速に処理するトレードオフを考えることが必要である。

1.2 コンピュータの構造

OS の仮想計算機部分を考える前に、仮想化されていないコンピュータの構造を確認する。コンピュータを構成している要素は、**図 1.2** のように、入出力装置、内部記憶装置、演算処理装置に大きく分けることができる。各要素は一般にバスと呼ばれる伝送路で結ばれている。



図 1.2 コンピュータの構造

入出力装置 (input/output device) は、演算処理装置が内部記憶装置以外とデータ交換、同期、通信を行うための装置である。周辺機器 (peripheral device) や入出力デバイス、I/O 装置、I/O デバイスとも呼ばれる。本書では誤解を生じなければ単にデバイスと呼ぶ。デバイスには非常に多様なものが存在する。たとえば、ハードディスク装置のような**外部記憶装置** (補助記憶装置, 2次記憶装置), LAN アダプタのような通信装置, キーボードやマウスのような入力装置, コンピュータモニタのような表示装置等がある。

内部記憶装置 (1次記憶装置) は、演算処理装置が利用するデータを保存

4 1. オペレーティングシステムとは

し、利用できるようにするための装置である。**メモリ** (memory), **主記憶** (main memory) とも呼ばれる。通常は、RAM (random access memory, ラム) と呼ばれる半導体メモリチップから構成される。メモリの各要素には識別のためのアドレス (address, 番地) が与えられ、演算処理装置はアドレスを指定して記憶要素へのデータの書出しや要素からのデータの読み込みをすることができる。メモリは、I/O デバイスに比べて高速アクセスが可能である。

演算処理装置は、一般に**プロセッサ** (processor) と呼ばれる要素である。**CPU** (central processing unit, 中央演算処理装置) とも呼ばれる。1 個のみのユニプロセッサ (uniprocessor) の場合もあるが、複数個から構成されるマルチプロセッサ (multiprocessor) の場合もあり、複数の場合には並列に処理を行うことが可能になる。また、複数のプロセッサから構成される場合には、プロセッサが同種のものばかりから成るホモジニアスな (homogeneous) 場合と、異なる種類のプロセッサが混在したヘテロジニアスな (heterogeneous) 場合がある。本書では、基本的な概念の理解をすることが中心のため、プロセッサが1 個のユニプロセッサの場合を中心に解説し、必要に応じてマルチプロセッサの基本的な場合も考える。

プロセッサにはさまざまな種類が存在するが、本書では図 1.3 に示すように高速アクセス可能な記憶領域である**レジスタ** (register) がプロセッサ内にあるとする。レジスタは、プログラムカウンタ、プロセッサ・ステータス・レジスタ、汎用レジスタ

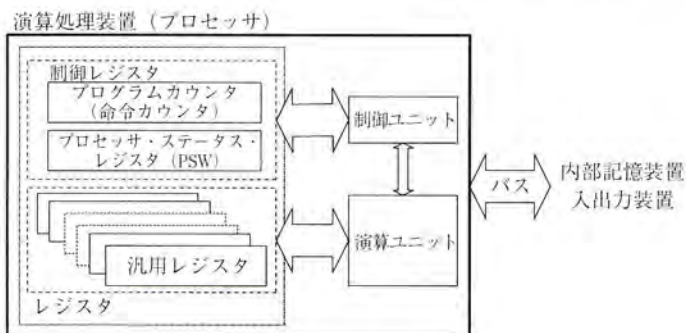


図 1.3 プロセッサの構造

スタなどの制御レジスタとデータの記憶、演算に利用できる汎用レジスタに大きく分類できる。

プログラムカウンタ (program counter, PC) は、次に実行する命令のアドレスを示すレジスタである。命令カウンタ (instruction counter) とも呼ばれる。**プロセッサ・ステータス・レジスタ** (processor status register) は、プロセッサの状態を表すフラグを集めたレジスタであり、**プロセッサ・ステータス・ワード** (processor status word, PSW) もしくは**プログラム・ステータス・ワード** (program status word) とも呼ばれる^{†1}。プロセッサには他にもプロセッサやメモリ管理のためのレジスタが含まれていることがある。これらは総称して**制御レジスタ** (control register) と呼ばれる。**汎用レジスタ** (general purpose register) は制御レジスタのように用途が決まっておらず、一般にデータを保存し、演算に利用するためのレジスタである。個数はプロセッサによって異なるが、一般には数個から数十個であり、プロセッサ外の記憶装置であるメモリと比較した場合には少数である。汎用レジスタは**データレジスタ**と呼ばれることもある。

ところで、プロセッサはさまざまな空間を持っていると考えることができ、空間の分類によってプロセッサの機能や役割を整理することができる。プロセッサの空間は、コンテキスト空間、メモリ空間、命令セット空間に大きく分類できる。メモリ空間以外にI/O空間を持っている場合もある。**コンテキスト空間** (context space) は、プログラムカウンタ、プロセッサ・ステータス・レジスタ、汎用レジスタなどのレジスタから構成されるプロセッサ内の記憶空間である。**メモリ空間** (memory space) は、内部記憶が存在する空間である。内部記憶はメモリアドレス指定によってアクセスされることから、メモリ空間はメモリアドレス空間とも呼ばれる。また、アドレスに重きがあり、誤解のおそれがない場合にはアドレス空間と呼ばれることもある。**命令セット空間**^{†2} (instruction set space) は、プロセッサの持つ命令セットによって構成

†1 ここでワードは、プロセッサで標準的な大きさの記憶領域の単位を意味する。

†2 命令セット空間は、ハードウェアの場合には一般に固定であるが、異種プロセッサから構成されるマルチプロセッサでは、プロセッサによって命令セット空間は異なる。

索引

【あ】

アクセス違反	97
アクセス権	177
アクセス制御	176
アクセス制御行列	177
アクセス制御リスト	179
アクセス方式	116
アドレス変換キャッシュ	93
アプリケーション・プロ グラム	1
アルゴリズム	3
暗号	180
アンチウィルス	192

【い】

一貫性モデル	160
イベント	165
インターネットプロトコル	142
インタフェース定義言語	155
インデックスブロック	125
インプットメソッド	173

【う・え】

ウィンドウシステム	165
ウォームスタート	203
エイリアス	129
エニーキャスト	149
遠隔手続き呼出し	153
演算処理装置	4
エンディアン	171
エンド・ツー・エンド原則	143

【お】

応答時間	28
大型汎用機	208
オーバーヘッド	26
オーバーレイ	100
オーバーレイ・ローダ	101
オープン	117
オペレーティング システム	1
親ディレクトリ	122

【か】

改竄	180
回線交換	135
階層型ファイルシステム	208
階層ディレクトリ	121
回転待ち時間	68
外部記憶装置	3
外部断片化	81
外部割込み	20
書出し	116
拡張子	121
仮想化	2
仮想記憶システム	102
仮想計算機	199
仮想計算機モニタ	199, 208
仮想時間	108
仮想プロセッサ	8
カーネル	1
カーネルモード	19
カプセル化	145
ガーベジコレクション	84
カレントディレクトリ	122

監査ログ	195
関心の分離	196
間接通信方式	50
完全修飾ドメイン名	144
慣用暗号方式	182

【き】

飢餓状態	29, 70
起床	26
逆ページテーブル	91
キャッシュ	73, 106
キャッシュ・コヒーレンス	160
キャッシング	73
キャラクタ型デバイス	60
キャラクタユーザインタ フェース	163
ギャングスケジューリング	162
脅威	186
共進化	207
共通鍵暗号	181
銀行家のアルゴリズム	47

【く】

空間的局所性	107
クライアント・サーバ・ モデル	153
クラスタ	152
グラフィカルユーザ インタフェース	164
グリッド・コンピュー ティング	152
クリティカルセクション	35
グループ通信	147
クローズ	117

クロック・アルゴリズム	
	112
【け・こ】	
計算量的安全性	183
計数セマフォ	40
携帯電話	211
軽量プロセス	14
ケーバビリティリスト	178
コア	1
公開鍵暗号	182
公開鍵基盤	186
構造化	2
公平性	28
国際化	166
コードポイント	168
コード領域	15
コネクション	136
コネクション指向	136
コピーイング	85
コマンドインタプリタ	163
コリジョン	91
コールドスタート	202
コンテキスト切換え	26
コンテキスト空間	5
コンパクション	83
コンピュータウイルス	187
【さ】	
最悪適合	79
最小権限の原則	178
最大1回セマンティクス	157
再配置可能	83
最良適合	79
作業ディレクトリ	122
先読み	73
サスペンド	203
サービス提供サブシステム	2
参照カウント	85, 130
参照の局所性	107
参照ビット	96

サンドボックス	194
【し】	
シェル	163
時間順スケジューリング	30
時間の局所性	107
シーク時間	68
資源	7
シーケンシャルアクセス	116
シーケンシャル・コン	
システム・モデル	160
資源割付グラフ	46
事象駆動型	31
システムコール	6
実行可能状態	19
実行スケジューリング	26
実行セマンティクス	156
実行中	18
実効メモリアクセス時間	108
実行モード	19
実時間処理	28
時分割多重化	200
締切	28
縮切順スケジューリング	32
ジャーナリング・	
ファイルシステム	133
主記憶	4
順アクセス	116
循環待ち	46
条件変数	42
衝突	91
ショートカット	129
シリンダ	68
シングルプログラミング	17
侵入検出システム	194
シンボリックリンク	129
【す】	
スケジューラ	26
スタック領域	15
スタブ	154

スパイウェア	187
スーパーバイザ	1
スピンドル	67
スピロック	39
スプーラ	65
スラッシング	114
スルーブット	28
スレッド	14
スワップアウト	103
スワップイン	103
【せ】	
制御レジスタ	5
生産者/消費者問題	62
脆弱性	187
生体認証	186
セキュリティ	176
セキュリティ・ホール	187
セクタ	68
セグメンテーション	97
絶対パス	122
セマフォ	40
セマンティクス	154
先着順スケジューリング	29
先頭適合	78
【そ】	
相互排除	34
相対パス	122
双対性	161
束モデル	179
疎結合	151
ソーシャル・エンジニア	
リング	189
ソフトウェアシミュレー	
ション	201
ソフトウェア割込み	21
存在ビット	95
【た】	
耐タンパー性	185
タイムシェアリング	18

- | | | | | | |
|------------------|-------|-----------------|----------|--------------|---------|
| タイムシェアリングシステム | 208 | 転送時間 | 68 | パーソナルコンピュータ | 209 |
| タイムスライス | 31 | | | バッキングストア | 102 |
| ダイレクトメモリアクセス | 57 | 【と】 | | バックエンド | 173 |
| 多重度 | 114 | 同期 | 16 | バックドア | 188 |
| 多重レベル割込み | 24 | 同期式 | 51 | ハッシュ関数 | 91 |
| タスク | 14 | 特権命令 | 19 | ハッシュ表 | 123 |
| ゲーティビット | 96 | 特権モード | 19 | バッチ処理 | 28, 164 |
| ダブルバッファ | 62 | ドメイン | 176 | バッチ処理モニタ | 207 |
| 単位時間 | 24 | トラック | 67 | バッファ | 52, 62 |
| 単層カーネル | 197 | トレードオフ | 2 | バッファオーバーラン | 190 |
| 単層ディレクトリ | 120 | トロイの木馬 | 187 | バッファプール | 63 |
| 断片化 | 79 | | | パーティション | 66 |
| | | 【な・に】 | | ハードリアルタイム | 31 |
| 【ち・つ】 | | 内部記憶装置 | 3 | ハードリンク | 129 |
| 地域化 | 166 | 内部断片化 | 81 | ハード同期 | 162 |
| チェック・ポイント・リスタート | 48 | 内部割込み | 20 | 繁忙待機 | 39 |
| 逐次処理 | 17 | なりすまし | 180, 189 | 汎用レジスタ | 5 |
| 蓄積交換 | 138 | 2進セマフォ | 40 | | |
| 抽象化 | 2 | 入出力装置 | 3 | 【ひ】 | |
| 直接アクセス | 116 | 認証 | 180, 183 | ビジーウェイト | 39 |
| 直接通信方式 | 50 | | | ヒット | 94 |
| 通信手順 | 140 | 【ね・の】 | | ビットマップディスプレイ | 74 |
| 通信プロトコル | 140 | ネットワーク・アーキテクチャ | 135 | ビデオ RAM | 74 |
| | | ネットワーク透過 | 175 | 非同期式 | 51 |
| 【て】 | | ネットワーク・トポロジ | 138 | 非特権モード | 19 |
| デ이지チェーン | 57 | ネットワーク・ファイアウォール | 193 | 否認防止 | 181 |
| ディスクアクセススケジューリング | 69 | ノード | 138 | 非封鎖型 | 39 |
| ディスバッチ | 25 | ノンブロッキング型 | 51 | ヒープ領域 | 15 |
| ディスバッチャ | 25 | | | 平文 | 180 |
| ディレクトリ | 120 | 【は】 | | | |
| デジタル署名 | 184 | バイトオーダ | 171 | 【ふ】 | |
| データ領域 | 15 | ハイバネーション | 203 | ファイル | 116 |
| データレジスタ | 5 | ハイパーバイザ | 20, 199 | ファイルシステム | 118 |
| デッドロック | 45 | パケット | 137 | フィッシング | 190 |
| デバイスコントローラ | 55 | パケット型デバイス | 60 | 封鎖 | 35 |
| デバイスドライバ | 6, 61 | パケット交換 | 137 | 不可分命令 | 36, 37 |
| デフォルト | 170 | パケットフィルタ | 193 | 不揮発性 | 118 |
| デブロッキング | 65 | パス | 57 | ブートストラップ | 201 |
| デマンドページング | 104 | パス名 | 122 | ブートローダ | 201 |
| | | パスワード | 184 | プラグ・アンド・プレイ | 75 |
| | | | | プラッタ | 67 |

ブリフェッチ	73	ページング	87, 208	メモリ管理ユニット	88
フレーム	87	ベストエフォート	142	メモリ空間	5
プロキシ	194	ヘッダ	145	メモリ・マップト I/O	56
プログラムカウンタ	5	ヘッド	67	メモリ・マップト・	
プログラム割込み	20	変換エンジン	174	ファイル	130
プロセス	14	変換索引バッファ	93	文字コード系	168
プロセス間通信	49	変更ビット	96	文字集合	168
プロセス・スワッピング				モジュール	100
	102			モニタ	41
プロセッサ	4	【ほ】			
プロセッサ・ステータス・		保護	176	【ゆ・よ】	
レジスタ	5	保護ドメイン	176	有効ビット	95
プロセッサ・ステータス・		ポリウム	66	優先度逆転	43
ワード	5	ポーリング	22, 59	優先度キュー	27
プロセッサ利用率	28			ユーザインタフェース	163
ブロッキング	65	【ま】		ユーザモード	19
ブロッキング型	51	マイクロカーネル	197	ユニキャスト	147
ブロック	65, 81	マイクロプロセッサ	209	横取り	28
ブロック型デバイス	60	マーク・アンド・スweep	85	読み込み	116
ブロードキャスト	147			読み込み/書出し問題	54
プロトコル・スタック	145	待ち状態	19		
プロトコルデータユニット		マルウェア	187	【ら】	
	145	マルチキャスト	148	ライブラリ	12
フロントエンド	173	マルチスレディング	211	ライブロック	48
分散共有	157	マルチタスキング	16	ラウンドロビン	
分散共有メモリ	158	マルチプログラミング	17	スケジューリング	31
分散ファイルシステム	158	マルチプロセッシング	18	楽観の同期	49
		マルチモーダル・			
【へ】		インタフェース	211	【り・る】	
並行処理	17			リアルタイム	31
並列処理	18	【み】		リジューム	203
ペイロード	145	ミス	94	粒度	161
幂等セマンティクス	157	密結合	151	リロケータブル	83
ペクタ割込み	21	ミドルウェア	12	リンク	129, 138
ページアウト	106	ミニコンピュータ	208	リングバッファ	63
ページアウト・デーモン		ミューテックス	34	リング保護	19, 208
	107			ルック・アンド・フィール	166
ページイン	105	【め・も】		ルーティング	137
ページ置換え	106	命令セット空間	5	ルート	137
ページテーブル	88	メタデータ	124	ルートキット	188
ページフォールト	97, 104	メッセージダイジェスト	183	ルートモジュール	101
ページフォールト率	108	メッセージパッシング	49		
ページ枠	87	メモリ	4		

【れ・ろ】

例 外	20
レコード	116
レジスタ	4
レート・モニトニック・ スケジューリング	33
連想メモリ	92

ログ構造ファイルシステム	131
ロケール	167
ロック	34
ロック・キー	179
ロールバック	48
	【わ】
ワーキングセット	107

ワークステーション	209
ワクチン	192
ワーム	188
割込み	20
割込みハンドラ	21
割込みベクタ	22
割込みレベル	24
割出し	20

【A・B】

ASCII	168
Belady の異常	110

【C】

CAS 命令	38
C-LOOK	71
CORBA	155
CPU	4
C-SCAN	71

【D・E・F】

DDoS	190
DMA	57
DNS	144
DoS	190
EDF	32
FAT	126
FCFS	29, 70
FIFO	110
FQDN	144

【G・I】

GC	84
GUI	164
IANA	147
IEEE	143
i-node	128
I/O 空間	6
I/O 命令	56

IPC	49
IPL	202
IPv 6	142
IP アドレス	143
ISO	143
ISO/IEC 10646	170
ISO/IEC 646	168
ITU-T	143

【L・M】

LIFO	110
Linux	122, 204
LOOK	71
LRU	111
MMU	88

【O・P・Q】

OPT	111
OS	1
OSI	141
PDU	145
PKI	186
Plan 9	119, 204
POSIX	124, 167
PSW	5
P 操作	40
QoS	136

【R】

RAID	133
RMS	33

RPC	153
RR	31
RSA 暗号	182

【S】

SCAN	70
SPTF	30
SSTF	70
SVC	6

【T】

TAS 命令	38
TCP/IP	142
TLB	93
TSS	208

【U】

UCS	170
UDP	143
UFS	127
Unicode	170
UNIX	60
UNIX ファイルシステム	127

【V・W・X】

V 操作	40
WIMP インタフェース	165
WWW	209
XML	156
X ウィンドウシステム	165

— 著者略歴 —

- 1983年 東京大学理学部情報科学科卒業
1988年 東京大学大学院博士課程修了 (情報科学)
理学博士
1988年 パーソナルメディア株式会社勤務
1993年 電気通信大学助手
1998年 メディア教育開発センター助教授
2001年 総合研究大学院大学助教授 (併任)
2003年 メディア教育開発センター教授
2003年 総合研究大学院大学教授 (併任)
2009年 千葉大学大学院教授
現在に至る

オペレーティングシステム

Operating Systems

© Noritaka Osawa 2008

2008年3月21日 初版第1刷発行

2020年4月5日 初版第9刷発行

検印省略

著者	大 澤 範 高
発行者	株式会社 コロナ社
	代表者 牛来真也
印刷所	新日本印刷株式会社
製本所	有限会社 愛千製本所

112-0011 東京都文京区千石4-46-10

発行所 株式会社 コロナ社

CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替 00140-8-14844・電話 (03) 3941-3131 (代)

ホームページ <https://www.coronasha.co.jp>

ISBN 978-4-339-02707-5 C3355 Printed in Japan

(花井)



JCOPY <出版者著作権管理機構 委託出版物>

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつど事前に、出版者著作権管理機構 (電話 03-5244-5088, FAX 03-5244-5089, e-mail: info@jcopy.or.jp) の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。