

付録1 ISBOKの簡易版(第4階層まで)

#	第1階層
##	第2階層
###	第3階層
####	第4階層

1	情報技術
1.1	コンピュータアーキテクチャ
1.1.1	基本的なデータの表現(非数値, 数値(整数, 実数, 誤差, 精度))
1.1.1.1	数値データの基本的な機械表現
1.1.1.2	非数値データの基本的な機械表現
1.1.1.3	整数と浮動小数点表現の有限精度
1.1.1.4	コンピュータ算術における誤差とそれに関連した移植性の問題
1.1.1.5	コンピュータアーキテクチャの基本概念
1.1.2	デジタル化された情報の物理的な表現(例: データ, テキスト, イメージ, 音声, ビデオ)
1.1.3	CPUの構成(CPU, メモリ, レジスタ, アドレス指定モード, 命令セット)
1.1.3.1	基本構成(フォンノイマン, ブロックダイアグラム, データバス, コントロールバス, 機能ユニット, 命令サイクル)
1.1.3.2	命令及びアドレス指定モード(命令セットとタイプ)
1.1.3.3	命令及びアドレス指定モード(アセンブリ-機械語)
1.1.3.4	アドレス指定モード
1.1.3.5	制御ユニット(命令取り出しと実行, オペランド取り出し)
1.1.3.6	CISC, RISC
1.1.3.7	コンピュータの構成
1.1.3.8	メモリシステム
1.1.4	コンピュータシステムの構成要素(バス, コントローラ, 記憶システム, 周辺装置)
1.1.4.1	周辺装置(入出力と割込み)
1.1.4.2	周辺装置(入力/出力の制御方式, 割込み)
1.1.4.3	周辺装置(外部記憶, 物理的な構成や装置)
1.1.4.4	補助記憶, テープ, 光学式
1.1.4.5	記憶システムと技術
1.1.4.6	空間の割付け, 階層
1.1.4.7	主メモリの構成, バス動作, 選択とアドレス指定のサイクルタイム
1.1.4.8	キャッシュメモリ, 読み/書き
1.1.4.9	仮想メモリ
1.1.4.10	コンピュータと他の装置(センサ, エフェクタ)間とのインタフェース
1.1.5	マルチプロセッサアーキテクチャ
1.1.5.1	システムアーキテクチャ(単一多重処理, 分散処理, スタック, 配列, ベクトル, マルチプロセッサ, ハイパーキューブアーキテクチャ, スーパーコンピュータ)
1.1.6	デジタル論理とシステム
1.1.6.1	論理要素とスイッチング理論(最小化の概念と関数の実装)
1.1.6.2	伝達遅延とハザード
1.1.6.3	デマルチプレクサ, マルチプレクサ, 複号器, 符号器, 加算器, 減算器, 比較器, シフトレジスタ, カウンタ
1.1.6.4	ROM, PROM, EPROM, EAPROM, RAM
1.1.6.5	同期回路の分析と統合, 非同期回路対同期回路
1.1.6.6	レジスタ転送の表現, 条件付き及び無条件
1.1.6.7	アルゴリズム状態機械, ステアリングネットワーク, ロード転送信号
1.1.6.8	3値状態とバス構造
1.1.6.9	ブロックダイアグラム, タイミングダイアグラム, 転送言語
1.2	アルゴリズムとデータ構造
1.2.1	形式的な問題と問題解決
1.2.1.1	欲張りアルゴリズムを用いた問題解決戦略
1.2.1.2	分割統治アルゴリズムを用いた問題解決戦略
1.2.1.3	バックトラックアルゴリズムを用いた問題解決戦略
1.2.1.4	ソフトウェア設計プロセス(仕様から実装まで)
1.2.1.5	問題認識文とアルゴリズムの決定(手続き)
1.2.1.6	実装戦略(トップダウン, ボトムアップ, チーム対個人, 管理タスク)
1.2.1.7	形式的検証概念

1.2.1.8	計算の形式モデル
1.2.2	基本的なデータ構造（リスト，配列，記号列，レコード，集合，リンク付きリスト，スタック，キュー，木，グラフ）
1.2.3	複合データ構造（例：データ，テキスト，音声，画像，ビデオ，ハイパーメディア）
1.2.4	抽象データ型
1.2.4.1	抽象データ型の目的と実装
1.2.4.2	非形式的仕様
1.2.4.3	形式的仕様，事前条件と事後条件，代数的仕様
1.2.4.4	モジュール，凝集度，結合度，データフローダイアグラム，及び階層チャートへの変換
1.2.4.5	正確さ，確認と検証，事前条件と事後条件，不変表明，コードの基本的な検査と設計の理解，構造化ウォークスルー
1.2.4.6	制御構造（選択，反復，再帰），データ型と問題解決のための使用
1.2.5	ファイル構成（順，直接アクセス，ハッシング，索引付）
1.2.5.1	ファイル（構成，アクセス方式）：ファイルレイアウト，基本的なファイルの概念（シーケンシャルファイル，シーケンシャル以外のファイル）
1.2.5.2	ファイル（構成，アクセス方式）：ディレクトリ内容と構成，命名，検索，アクセス，バックアップ
1.2.5.3	ファイル（構成，アクセス方式）：システムセキュリティの概要，セキュリティの方式及び装置，保護，アクセス，認証
1.2.6	ソートと探索のデータ構造とアルゴリズム
1.2.6.1	ソートのアルゴリズム（シェルソート，バケットソート，基数ソート，クイックソート），編集，報告，更新
1.2.6.2	探索アルゴリズム（順探索，二分探索，及び二分探索木）
1.2.6.3	探索，ハッシング，衝突の解消
1.2.7	アルゴリズムの効率，複雑さとメトリクス
1.2.7.1	上界及び平均界での漸近的分析；big"O"，little"o"
1.2.7.2	アルゴリズムにおける時間対空間のトレードオフ
1.2.7.3	複雑性クラス P，NP，P-空間；扱いやすい問題とそうでない問題
1.2.7.4	下界分析（ソートのための）
1.2.7.5	NP-完全性
1.2.7.6	$O(n^2)$ のソートアルゴリズム
1.2.7.7	$O(n \log n)$ のソートアルゴリズム
1.2.7.8	バックトラック，構文解析，離散型シミュレーションなど
1.2.7.9	アルゴリズムの分析の基礎
1.2.8	再帰的アルゴリズム
1.2.8.1	数学的帰納法と結びついた再帰的アルゴリズム
1.2.8.2	繰り返すと再帰的アルゴリズムの比較
1.2.9	ニューラルネットワークと遺伝的アルゴリズム
1.2.10	その他の重要なこと
1.2.10.1	計算可能な関数：チューリングマシンから選んだ計算可能な関数のモデル，RAM（部分的な）再帰的関数，ラムダ計算法，チャーチの定立
1.2.10.2	機械（例：万能チューリングマシン）
1.2.10.3	決定問題：再帰及び再帰的計算問題；決定不能な問題
1.2.10.4	並列アーキテクチャのモデル
1.2.10.5	並列アーキテクチャのためのアルゴリズム
1.2.10.6	数学的問題（良条件での問題，悪条件での問題）
1.2.10.7	数学的問題（数学問題に対する反復的近似，ニュートン法，ガウスの消去法）
1.2.10.8	数学的問題（誤りの分類：計算的，表現的，方法的，差異）
1.2.10.9	数学的問題（科学と工学における反復近似解法の応用）
1.2.10.10	計算の限界（計算可能性とアルゴリズムの至難性）
1.2.10.11	確率と統計
1.3	プログラミング言語
1.3.1	基本的なプログラミング言語の構造（言語間の機能比較）
1.3.2	機械語とアセンブリレベルの言語
1.3.3	手続き型言語
1.3.3.1	手続き型プログラミングの利点と欠点
1.3.3.2	基本的な型宣言（算術演算子，代入，条件文，ループと再帰）
1.3.3.3	手続き，関数とパラメータ；配列とレコード
1.3.4	非手続き型言語（論理型，関数型，イベントドリブン）
1.3.5	第4世代言語
1.3.6	言語のオブジェクト指向への拡張
1.3.7	プログラミング言語，設計，実装と比較

1.3.7.1	初期の言語の歴史
1.3.7.2	手続き型言語の発展
1.3.7.3	非手続き型言語の発展
1.3.7.4	バーチャルコンピュータ
1.3.7.5	基本と構造化データ型
1.3.7.6	ユーザ定義のデータ型の生成と応用
1.3.7.7	式, 評価の順序, 及び副作用
1.3.7.8	式とステートメントの抽象化としてのサブプログラムとコールチェーン
1.3.7.9	例外の取り扱い
1.3.7.10	データ共有と制限付きアクセスの仕組み
1.3.7.11	静的対動的スコープ, 生涯, 可視性
1.3.7.12	パラメータのパスメカニズム (参照, 値, 名前, 結果など)
1.3.7.13	型検査の規則の多様性とそれらのしくみ
1.3.7.14	記憶領域のスタックベースのアプリケーション
1.3.7.15	記憶領域のヒープベースのアプリケーション
1.3.7.16	計算の制限付きモデル及び正規表現 (regular expression) のアクセプタとしての有限オートマトン
1.3.7.17	プログラミング言語解析に対する正規表現の応用
1.3.7.18	プログラミング言語の構文法のための形式的記述デバイスとしての文脈自由型文法の使用
1.3.7.19	文脈自由型文法とプッシュダウンオートマトンの同等性
1.3.7.20	プログラミング言語の構文解析におけるプッシュダウンオートマトンの応用
1.3.7.21	言語翻訳プロセス, コンパイラからインタプリタまで
1.3.7.22	プログラミング言語の意味論
1.3.7.23	関数型言語のパラダイムと言語
1.3.7.24	並列プログラミングの構成概念
1.3.7.25	手続き型言語 (実装, 性能の改善, デバッグ, バグ発見)
1.3.7.26	コンパイラとトランスレータ
1.3.7.27	最高水準言語: SQL, 第 4 世代言語
1.3.7.28	オブジェクト指向設計, 言語, 及びプログラミング
1.3.7.29	論理プログラミング言語 (LISP, PROLOG, 論理指向プログラミング)
1.3.7.30	コードジェネレータ
1.3.7.31	エキスパートシステムシェル
1.3.7.32	ソフトウェア設計言語
1.4 オペレーティングシステム	
1.4.1	アーキテクチャ, オペレーティングシステムの目標と構成 (構成法, レイヤーモデル, オブジェクトサーバモデル)
1.4.2	オペレーティングシステムとハードウェアアーキテクチャの相互作用
1.4.3	プロセス管理 (並行プロセス, 同期化)
1.4.3.1	タスク, プロセス, コンテキストスイッチャのディスパッチング, 割込みの役割
1.4.3.2	構造, 実行可能リスト, プロセス制御ブロック
1.4.3.3	並行プロセスの実行
1.4.3.4	共用アクセス, 競合条件
1.4.3.5	デッドロック (原因, 条件, 予防)
1.4.3.6	モデルとメカニズム (例: ビジー待機, スピンロック, デカーのアルゴリズム, セマフォア, mutex ロック, 領域, モニタ)
1.4.3.7	優先的な, 及び非優先的なスイッチング
1.4.3.8	スケジューラとスケジューリング方策
1.4.4	メモリ管理
1.4.4.1	物理的メモリとレジスタ
1.4.4.2	オーバーレイ, スワッピング, 区分
1.4.4.3	ページとセグメント
1.4.4.4	配置と再配置ポリシー
1.4.4.5	スラッシング, 作業用セット
1.4.4.6	フリーリスト, レイアウト, サーバ, 割込み, 障害の回復
1.4.4.7	メモリ保護, 回復管理
1.4.5	資源の配分とスケジューリング
1.4.5.1	プロトコルスイート (通信とネットワーク接続); ストリームとデータグラム
1.4.5.2	インターネット接続とルーティング, サーバとサービス
1.4.5.3	OS のタイプ (単一ユーザ, 複数ユーザ, ネットワーク)
1.4.5.4	分散及びリアルタイムシステムにおける同期化とタイミング
1.4.5.5	リアルタイムシステムにおける特別な配慮 (障害, リスク, 回復)
1.4.5.6	OS コーティリティ
1.4.5.7	ハードウェアの進化 (経済力, 制約)

1.4.5.8	リアルタイムシステムと組込型システムのアーキテクチャ
1.4.5.9	組込型リアルタイムシステムにおける特別な配慮(タイミングの厳しい要求;信頼性,耐震性,耐故障性;入力と出力の考慮;時間に関する問題の認識;並行性;複雑なインタフェース(装置,装置間,ソフトウェア);リアルタイムシステムのためのテストの不適合性)
1.4.6	二次記憶装置の管理
1.4.7	ファイルとディレクトリシステム
1.4.8	保護とセキュリティ
1.4.9	分散型オペレーティングシステム
1.4.10	人間との対話のための OS 支援(例:GUI,対話型ビデオ)
1.4.11	OSの相互運用の可能性と互換性(例:オープンシステム)
1.4.12	オペレーティングシステム,ユーティリティ,ツール,コマンド及びシェルプログラミング
1.4.13	システム運営と管理
1.4.13.1	システムブートストラッピング/初期プログラムロード
1.4.13.2	システム生成
1.4.13.3	システム構成
1.4.13.4	性能分析,評価及び監視
1.4.13.5	システムの最適化とチューニング
1.4.13.6	システム運営機能(バックアップ,セキュリティ及び保護,ユーザの追加と削除)
1.5	通信
1.5.1	国際通信標準,モデル,傾向
1.5.1.1	コンピュータネットワークと制御(位相,公衆通信業者,機器構成,誤り検出と訂正,ポーリングとコンテンツプロトコル,セキュリティと暗号化)
1.5.1.2	ネットワーク設計と運営;ネットワークアーキテクチャ(ISO,SNA,DNA),プロトコル(X.25,ISO,など)
1.5.2	データの伝送(媒体,信号化技法,伝送上の損失,符号化,エラー検出,圧縮)
1.5.2.1	通信システム技術(伝送媒体,アナログ・デジタル,通信ハードウェア及びソフトウェア)
1.5.2.2	ネットワークオンチップ,システムエリアネットワーク,スイッチング方式
1.5.3	回線構成(誤り制御,フロー制御,多重化)
1.5.4	ローカルエリアネットワーク(LAN)
1.5.4.1	ネットワークトポロジ,媒体アクセス制御,多重化
1.5.4.2	LANとWAN(トポロジ,ゲートウェイ),利用(機能とOA),PBX
1.5.4.4	分散システムのアーキテクチャ
1.5.4.5	分散システムのハードウェアの概観
1.5.5	広域ネットワーク(WAN)(交換技術,一斉同報通信技術,ルーティング)
1.5.6	ネットワークアーキテクチャとプロトコル
1.5.7	インターネット接続
1.5.8	ネットワーク設定,性能解析及び監視
1.5.9	ネットワークのセキュリティ(暗号化,デジタル署名,認証)
1.5.10	高速ネットワーク(例:ISDN,SMDS,ATM,FDDI)
1.5.11	ネットワーク技術の最新の話(光通信網,無線通信,IPv6,NGN...)
1.5.12	通信アプリケーション(例:クライアントサーバ,EDI,電話網,e-mail,マルチメディア,ビデオ会議,付加価値通信網)
1.5.12.1	通信を使ったグラフィック情報やビデオ情報の伝達方式,データ圧縮,クライアントサーバ,ディスプレイ技術(例:AOLインタフェース,XWindows)
1.5.13	オープンシステムのプロトコル
1.5.13.1	転送プロトコル
1.5.13.2	アプリケーション支援プロトコル(暗号化,契約,同意)
1.5.14	情報分散
1.5.14.1	ネットワーク構造
1.5.14.2	クライアントサーバ/シンクライアントサーバ技術
1.5.14.3	ネットワーク,ルーティング,性能分析
1.5.14.4	通信システム
1.6	データベース
1.6.1	DBMS(特徴,機能,アーキテクチャ)
1.6.1.1	DBMS(特徴,機能,アーキテクチャ);データベースシステムの構成要素(データ,辞書,アプリケーションプログラム,ユーザ,管理運営)
1.6.1.2	DBMS(関係代数の概観)
1.6.1.3	論理設計(DBMS非依存設計):ER,オブジェクト指向
1.6.1.4	ANSI/SPARC 3層スキーマモデル
1.6.2	データモデル(関係,階層,ネットワーク,オブジェクト,意味オブジェクト,正規形)
1.6.2.1	リレーショナルデータモデル
1.6.2.2	概念モデリング(例:ER,オブジェクト指向)

1.6.2.3	モデルの相互変換（例：階層型からリレーショナルへなど）
1.6.2.4	正規形
1.6.3	トランザクション
1.6.3.1	ASID 特性
1.6.3.2	解離性水準(Isolation Level)
1.6.3.3	アクセス権限
1.6.3.4	データの保全
1.6.4	整合性（参照整合性，データ項目，内部関係）：参照の表現；エンティティと参照の整合性
1.6.5	データ定義言語（スキーマ定義言語，グラフィカルな開発ツール，辞書など）
1.6.6	アプリケーションインタフェース
1.6.6.1	典型的なデータベースシステムが提供する機能（アクセス方式，セキュリティ，デッドロック及び並列性の問題，第4世代言語環境）
1.6.6.2	データ操作言語，SQL，埋め込み SQL
1.6.6.3	アプリケーションとユーザインタフェース（DML，問合せ，SQL）
1.6.6.4	イベントドリブンのスクリーンオブジェクト（ボタン，リストボックスなど）
1.6.6.5	物理的トランザクション処理（クライアントサーバの考慮）
1.6.6.6	プロセスの考え方におけるクライアントとサーバの分散
1.6.6.7	OR マップ，DAO(Data Access Object)
1.6.7	知識問合せプロセッサと問合せ機構，OLAP ツール
1.6.8	分散型データベース，リポジトリとデータウェアハウス
1.6.9	DBMS 製品：データベースシステムの最新状況（例：ハイパーテキスト，ハイパーメディア）
1.6.10	データベースマシンとサービス
1.6.11	データとデータベースの管理
1.6.11.1	データの管理
1.6.11.2	データベース管理：データベースシステムの社会的な影響；セキュリティとプライバシー
1.6.11.3	データ及び応用システムの所有権とアクセス管理
1.6.11.4	役割と実行能力に基づくアクセスモデル
1.6.11.5	複製（レプリケーション）
1.6.11.6	システムキャパシティ計画，性能設計
1.6.11.7	冗長性，安全性及びバックアップ計画と運用管理
1.6.12	データ辞書，事典，リポジトリ
1.6.13	情報検索（例：インターネットツール，イメージ処理，ハイパーメディア）
1.6.14	データマイニング
1.7	人工知能
1.7.1	知識表現
1.7.1.1	人工知能の歴史，範囲と限界，チューリングテスト
1.7.1.2	人工知能の社会的，倫理的，法的及び哲学的側面
1.7.1.3	問題と状態空間
1.7.2	知識工学
1.7.3	推論処理
1.7.3.1	基本的な制御戦略（例：深さ優先，広がり優先）
1.7.3.2	前向き及び後向き推論
1.7.3.3	ヒューリスティック探索（生成とテスト，山登り法，横型探索，目的達成手段解析，図式探索，ミニマックス検索）
1.7.3.4	エキスパートシステムとシェル
1.7.4	その他の技術（ファジー論理，事例ベース推論，自然言語及び音声認識）
1.7.5	知識ベースシステム
1.7.5.1	自然言語，音声と映像
1.7.5.2	パターン認識
1.7.5.3	機械学習
1.7.5.4	ロボット工学

2	組織と管理概念
2.1	組織理論一般
2.1.1	組織の階層とフローモデル
2.1.2	組織上の作業グループ
2.1.3	組織のスパン(単一ユーザ，作業グループ，チーム，企業，グローバル)
2.1.4	企業内での IS の役割（戦略的，戦術的及び業務的）
2.1.5	組織構造における IS の影響，IS と継続的な改善
2.1.6	組織の構造（集権型，分権型，マトリクス型）
2.1.7	組織でのソフトウェア使用に関する組織的問題
2.2	情報システム管理

2.2.1	IS 計画
2.2.1.1	企業計画と IS 計画の連携
2.2.1.2	戦略的な IS 計画
2.2.1.3	短期的な IS 計画
2.2.1.4	リエンジニアリング
2.2.1.5	継続的な改善
2.2.2	IS 機能のコントロール (例: EDP 監査, アウトソーシング)
2.2.3	スタッフ配置と人的資源管理
2.2.3.1	スキル計画
2.2.3.2	スタッフの業績管理
2.2.3.3	権限委譲/仕事の所有者
2.2.3.4	教育と訓練
2.2.3.5	競争, 協力及び報酬の構造
2.2.3.6	2.2.15.6 へ移動
2.2.4	IS の機能構造 (企業内対アウトソーシング)
2.2.5	IS 組織の目的と目標の決定
2.2.6	ビジネスとしての IS 管理 (例: 顧客の定義, IS の任務, IS の決定的成功要因の定義)
2.2.7	CIO とスタッフの機能
2.2.8	サービス機能としての IS: 性能評価 (外部/内部), サービスのマーケティング
2.2.8.1	ISO-9000
2.2.8.2	Baldrige National Quality Program
2.2.9	IS の財政管理 (費用の投入と回収)
2.2.10	IS の戦略的な使用 (例: 競争優位と IS, プロセスリエンジニアリング, IS と品質, IS の世界的な影響と国際的な考慮)
2.2.11	知的作業, エンドユーザコンピューティング (支援, 役割, 生産性, 機能)
2.2.12	IS の方針, 運用手順の公式化及びコミュニケーション
2.2.13	バックアップ, 災害対策, 及び復旧の計画
2.2.14	新しい技術の管理
2.2.15	サブ機能の管理
2.2.15.1	通信管理
2.2.15.2	コンピュータ設備の管理 (例: 分散処理の自動操作, 処理能力の計画, サイト保守)
2.2.15.3	グループの意思決定支援システムの管理
2.2.15.4	データ管理
2.2.15.5	データとアプリケーションの所有権
2.2.15.6	創造性を育む風土の醸成
2.2.15.7	品質管理 (例: 信頼性と品質技術, QC チーム)
2.2.15.8	コンサルティングの管理, アウトソーシング
2.2.15.9	資源競合の調整
2.2.15.10	システムの設置, 移行, 運用, 及び廃棄に関する業務運用上の問題
2.2.15.11	ソフトウェアの改革と維持を支援する活動と訓練の管理
2.2.15.12	ソフトウェア工学の活動 (開発, 制御, 管理, 操作)
2.2.16	セキュリティと管理, ウィルスとシステムの安全性
2.2.17	コンピュータ運用の管理 (例: 媒体の管理, スケジューリング, 自動化と機能横断的な文脈)
2.3	意思決定理論
2.3.1	計測とモデル化
2.3.2	確実性, 不確実性及びリスクの下での意思決定
2.3.3	情報のコスト/価値, IS の競争可能な価値
2.3.4	意思決定モデルと IS (最適化, 満足化)
2.3.4.1	サイモンモデル
2.3.5	グループの意思決定プロセス
2.3.5.1	集団的浅慮(集団思考)
2.4	組織行動
2.4.1	ジョブ設計理論
2.4.2	文化の多様性
2.4.3	グループダイナミクス
2.4.4	チームワーク, リーダシップ及び権限委譲
2.4.5	影響力, 権限, 政策の行使
2.4.6	認知スタイル
2.4.7	交渉と交渉スタイル
2.4.8	合意の形成
2.6	スケジューリングの理論と概念

2.6.1	スケジューリングの目的
2.6.2	スケジューリングの方法
2.6.2.1	Johnson Rule
2.6.2.2	資源能力（引当可能容量，引当可能性，在庫）
2.6.2.3	山積み，山崩し
2.6.2.4	装置保全
2.6.3	TOC(制約理論)
2.6.3.1	DBR(ドラム，バッファ，ロープ)
2.6.3.2	CC(クリティカルチェーン)
2.7	変革プロセスの管理
2.7.1	変革に抵抗する理由
2.7.2	変革を動機づける戦略
2.7.3	変革の計画づくり
2.7.4	変革の管理
2.7.5	モデル化プロセスとシステム
2.7.6	データ取得手段の実験
2.7.7	プロセスや関係するソフトウェアの改良でのリーダーシップ
2.7.8	対処方略（衝撃，拒否，怒り，抑圧，契約，受諾）
2.7.9	グループ及びチーム学習
2.7.10	変革者の特性
2.7.10.1	傾聴と理解
2.7.10.2	調停と交渉
2.7.10.3	ファシリテーション
2.7.10.4	タイプの違いの認識（Myers Briggs，Rohm）
2.7.10.5	脅威と脅威の管理
2.7.10.6	システム思考
2.7.10.7	個人の専門的スキル
2.7.10.8	メンタルモデル
2.7.10.9	共有ビジョンの構築
2.7.10.10	変革の必要性和重要性の述べ方
2.8	ISの法的，倫理的側面
2.8.1	ソフトウェアの販売・使用許諾及び取次ぎ
2.8.2	契約の基礎
2.8.2.1	契約法
2.8.2.2	契約，請負契約，準委任契約
2.8.2.3	労働法規，労働者派遣法
2.8.3	プライバシー法
2.8.4	取次ぎと規制集団
2.8.5	知的財産権の保護と倫理
2.8.5.1	知的財産の保護
2.8.5.2	知的財産の形態，保護の意味，違反時の罰則
2.8.5.3	倫理（盗作，誠実，プライバシー，使用，誤用，コンピュータ技術の限界）
2.8.6	倫理（個人の責任，専門家の責任）と法律，倫理モデル，倫理的社会的分析
2.8.6.1	個人の責任（正直，公正，自律，正義の原則，及び公衆の福利）
2.8.6.2	専門家の責任（知識とスキルに対する期待と信頼）
2.8.6.3	責任あるコンピュータ技術者のための倫理綱領
2.8.6.4	倫理的行動への動機と重要性（ハッカーの精神と専門家の行動）
2.8.6.5	倫理的モデル（Benthamの行為の規範，kantのモラルの責務，Rawlesの社会的契約の交渉，Coveyの指導者中心の原則）
2.8.6.6	倫理的分析の要素（倫理的主張の議論，倫理的選択，疑問のある倫理的アプローチ（素朴な相対主義，エゴイズム，代行業と法律万能主義）
2.8.6.7	倫理的分析，事例からの議論，類例と反例，関係者の特定と倫理的問題，倫理綱領の適用，代替的な行動計画の特定と評価）
2.8.6.8	社会的分析（技術の使用と開発における社会的な文脈の影響，社会的な相互作用における力関係，開発者の価値観を具体化する技術，人間の多様性，設計と開発における経験的データの利用）
2.8.7	コンピュータアプリケーションのリスク，損失及び責任
2.8.8	保証
2.8.9	コンピュータ犯罪
2.8.9.1	ソフトウェアウィルスとその他の被害
2.8.9.2	ソフトウェアの詐欺，悪用，ハッカー
2.9	プロフェッショナリズム

2.9.1	現行の定期的, 専門的, 学術的刊行物
2.9.2	証明書の発行
2.9.3	専門組織(例: DPMA ,ACM ,DSI ,IEEE ,ASQC ,AIS ,INFORMS ,IPSJ ,ISSJ ,JASMIN)
2.9.4	専門家会議
2.9.6	IS 産業(製造業, OEMs, システムインテグレータ, ソフトウェア開発業)
2.9.7	コンピューティングの歴史的社会的な文脈
2.10	対人関係の能力
2.10.1	コミュニケーション能力
2.10.2	インタビュー, 質問, 傾聴
2.10.3	プレゼンテーションの技能
2.10.3.1	口頭及び文書によるコミュニケーション
2.10.3.2	グラフィックスとマルチメディアの利用
2.10.3.3	訓練(目的, 目標, コンピュータの基礎)
2.10.4	コンサルティングの能力
2.10.5	執筆能力
2.10.5.1	技術文書の基礎
2.10.5.2	文書作成の原則と標準
2.10.5.3	ソフトウェア文書の展開
2.10.5.4	文書作成ツール
2.10.5.5	生涯学習の手段としての執筆
2.10.5.6	観察結果を獲得する方法としての雑誌記事の執筆
2.10.5.7	知識の調査結果と問題解決案の執筆
2.10.6	積極的な態度と取組み
2.10.7	個人の目標の設定, 意思決定, 時間管理
2.10.8	原則を中心としたリーダーシップ
2.10.9	交渉の原則
2.10.10	創造性と機会発見力
2.10.11	クリティカル思考
2.10.11.1	情報の洗練/抽象化
2.10.11.2	タスクの優先順位付け
2.10.11.3	問題の案出
2.10.11.4	判断の一時停止
2.10.11.5	2.10.15.2 へ移動
2.10.11.6	発散的思考, 水平思考, 直線的思考などの戦略
2.10.11.7	問題点の質問
2.10.11.8	責任の引受, 意思決定, 権限委譲
2.10.11.9	多種多様な仕事と異なる見方
2.10.11.10	反論と寛容に係る処理
2.10.11.11	建設的な批判の受容と授与
2.10.12	データの測定と分析
2.10.12.1	文脈におけるデータの評価
2.10.12.2	データの測定/確定
2.10.12.3	データ変換と提示
2.10.12.4	データの代数的・機能的変換
2.10.12.5	データの分類と構成
2.10.12.6	プロセスモデリングとシミュレーションにおけるデータ使用
2.10.12.7	データ定義におけるサンプリング概念
2.10.12.8	データを使ったプロセス確認
2.10.14	個人の問題解決
2.10.14.1	問題のスコープと制約
2.10.14.2	解の予測, 正確さ, 時間制約, トレードオフ
2.10.14.3	完成と現実, 資質と見込みのギャップ
2.10.14.4	問題達成基準
2.10.14.5	創る, 推論する, 論理設備を見つける・取り消す
2.10.14.6	構成要素間の関係の決定
2.10.14.7	二者択一と関連諸解の考察
2.10.14.8	立案, 見積り, プロセス段階の文書化, 成果, 及び活動範囲
2.10.14.9	拘束時間, 予算, マルチタスキング, 課金作業, バランスのとれた仕事量
2.10.14.10	解の検証と結果の確認
2.10.15	発想法
2.10.15.1	KJ 法
2.10.15.2	発案を生み出すブレinstoーミング

2.11 基本的な組織の機能	
2.11.1	支払い(Payments)
2.11.1.1	キャッシュマネジメントの諸概念
2.11.1.2	デジタル紙幣
2.11.1.3	国を超えたキャッシュフロー
2.11.1.4	クレジットフロー, 取引, 承認
2.11.1.5	EDI (電子的データ交換) の処理セット
2.11.1.6	財務のマークアップ言語 (例: IFX)
2.11.1.7	会計システムのインタフェース (GL, AR, AP)
2.11.1.8	勘定残高監査
2.11.2	ビジネスの関係 (例: CtoB, CtoC, CtoG, BtoB, BtoG, GtoG)
2.11.3	ビジネスのモデル, 伝統的/電子的商取引
2.11.3.1	ショッピングモール
2.11.3.2	オークション
2.11.3.3	物々交換取引
2.11.3.4	地域共同体と協力
2.11.3.5	情報プロバイダ
2.11.3.6	サービスプロバイダ
2.11.3.7	ポータル
2.11.4	バリューチェーン (VC) の概念
2.11.5	サプライチェーンマネジメント (SCM) の概念
2.11.6	アテンション
2.11.7	マーケティングと広告
2.11.8	小売り
2.11.8.1	販売網
2.11.8.2	生産者直送販売
2.11.8.3	顧客特定化
2.11.8.4	ポータルと経路
2.11.9	製造業と製品
2.11.9.1	部品表, 製品データ管理, 製品ライフサイクル管理
2.11.9.2	生産計画, 生産管理, 品質管理
2.11.9.3	製造者責任, トレーサビリティ, 保守, サービス
2.11.9.4	生産管理方式 (MRP, 製番管理, カンバン)
2.11.10	人的資源管理とコンプライアンス
2.11.11	在庫管理
2.11.12	発送 (船積み)
2.11.13	調達
2.11.14	注文票と顧客サービス
2.11.15	会計監査と管理

3 システムの理論と開発	
3.1 システムと情報の概念	
3.1.1 一般システム理論	
3.1.1.1	チャーチマンの三位一体説
3.1.1.2	システムック表現 (X のために, Y によって, Z する)
3.1.2 システム概念 (例: 構造, 境界, 状態, 目的)	
3.1.2.1	情報理論(Shannon)の基本概念
3.1.2.2	組織システム, ソフトウェア製品とプロセスについての論考
3.1.2.3	システムに対するユーザと供給者の関係
3.1.2.4	データ, 情報, 知識, システム
3.1.2.5	複雑系 (例: カオス, 散逸構造, フラクタル)
3.1.2.6	システムダイナミクス
3.1.3 開放系と閉鎖系	
3.1.3.1	生圏システム
3.1.3.2	生命システム
3.1.4 システムの構成要素と関係	
3.1.5 システム管理 (標準, 管理理論, フィードバック, ループ, 測定, 品質)	
3.1.6 情報システムの特徴	
3.2 システム開発のアプローチ	
3.2.1 システム開発モデル (例: SDLC, プロトタイプング)	
3.2.1.1	システム開発のライフサイクル: ソフトウェアのライフサイクルモデル (反復による強化, フェーズ分け開発, スパイラル, ウォーターフォール)

3.2.1.2	プロトタイピングによる開発
3.2.1.3	パッケージ使用による開発
3.2.1.4	データ指向開発技術
3.2.1.5	プロセス指向開発技術
3.2.1.6	オブジェクト指向開発技術（ボトムアップ設計，再利用の支援）
3.2.1.7	システムエンジニアリングの考察
3.2.1.8	システムの構成要素としてのソフトウェア
3.2.1.9	ソフトウェアプロセス及びプロダクトライフサイクルモデル
3.2.1.10	ソフトウェアの生成方式とツール（スクラッチからの設計とコード作成，プログラム及びアプリケーション生成プログラム，高水準言語，再利用可能なコンポーネント）
3.2.1.11	システム設計の方式とツール
3.2.2	パッケージの取得と実装
3.2.3	ソフトウェア構成要素の統合
3.2.4	エンドユーザ開発のシステム
3.2.5	システム開発アプローチの選択
3.3	システム開発の概念と方法論
3.3.1	組織のモデル化及びソフトウェアプロセスのモデル化
3.3.1.1	モデル化の概念
3.3.1.2	非同期モデル及びパラレルモデルを含む高度なモデル化の概念
3.3.2	データモデリング（例：ER図，正規形）
3.3.3	データ指向方法論
3.3.4	プロセス指向方法論
3.3.5	行動指向（イベントモデリング）方法論
3.3.6	オブジェクト指向方法論
3.3.7	ソフトウェア工学のプロセスとプロダクト
3.4	システム開発ツールと技術
3.4.1	CASE
3.4.1.1	方法論（情報工学，ジャクソン法，ヨードン法，マーチン法など），ソフトウェア設計目標
3.4.1.2	ツール（CASEツール，コードジェネレータ），GDSS（グループ意思決定支援システム）
3.4.1.3	ツール（IDEF 他の仕様と設計ツール），データベース設計とスキーマ変換ツール，実装ツール，描画とグラフィックツール
3.4.2	グループベースの方式（例：JAD，構造化ウォークスルー，設計とコードレビュー）
3.4.3	ソフトウェア実装の概念とツール（例：データ辞書，リポジトリ，アプリケーション生成プログラム，再利用，プログラム生成プログラム，ソフトウェア実装言語）
3.4.3.1	統合開発環境（例：Eclipse）
3.4.3.2	統合開発環境の事例(Boehm, 1984)
3.5	アプリケーション計画
3.5.1	インフラストラクチャ計画（ハードウェア，通信，データベース，サイト）
3.5.2	ISアーキテクチャの計画
3.5.3	運用のための計画
3.5.4	システム規模，ファンクションポイント，COCOMO，複雑さ管理のメトリクス
3.5.4.1	見積もりの考え方，見積もりの方法
3.5.5	ISセキュリティ，プライバシー及び管理のための計画
3.5.6	発注管理
3.5.6.1	RFP(Request for Proposal)
3.5.6.2	発注と権利（著作権，使用权）
3.5.6.3	発注
3.5.6.4	検収
3.5.6.5	プロジェクトの打ち切り，損害賠償
3.6	リスク管理
3.6.1	実現可能性の評価
3.6.2	リスク管理の原則
3.6.2.1	リスク評価
3.6.3	不測事態対応計画
3.6.3.1	プロジェクト不測事態対応計画（例：バッファ，コンティンジェンシープラン）
3.6.3.2	運用時の不測事態対応計画（例：テロ，天災，事故，情報漏洩）
3.7	プロジェクト管理
3.7.1	プロジェクト計画と適切なプロセスモデルの選択，プロジェクトのスケジューリングとマイルストーン
3.7.2	プロジェクトの組織，管理，原則，概念，問題
3.7.2.1	SEPG（ソフトウェアエンジニアリングプロセスグループ）

3.7.2.2	PMO(Project Management Office)
3.7.3	作業構造 (WBS) とスケジュール
3.7.4	プロジェクトスタッフの考え方 (例: マトリクス管理, 人間の要因, チーム組織, 報告)
3.7.4.1	プロジェクトファシリテーション (動機付け, 励まし, 報告・連絡・相談)
3.7.5	プロジェクトの管理 (計画, コストの見積り, 資源配分, ソフトウェアの技術的レビュー, 分析, フィードバック, コミュニケーション, 品質の確保, スケジュールリング, マイルストーン)
3.7.5.1	プロジェクト管理の文書化
3.7.5.2	プロジェクトスケジュールリングの表現
3.7.5.3	プロジェクトの経済 (コスト見積り技術とツール, コスト/便益の分析)
3.7.5.4	プロジェクトスケジュールリングのツール
3.7.5.5	進捗管理, 予定実績管理, EVMS(アードバリュー管理)
3.7.6	複数プロジェクトの管理
3.7.7	管理上の概念, ストレスと時間管理
3.7.8	システム文書の作成
3.7.9	ユーザ文書の作成 (例: 参照マニュアル, 操作手順, オンライン文書)
3.7.10	システムのメトリクス
3.7.11	スコープとスコープ管理
3.7.11.1	要求変更管理
3.7.11.2	変更追跡
3.7.12	構成管理
3.7.12.1	構成管理の原理と概念
3.7.12.2	システム発展の管理における役割
3.7.12.3	プロダクト統合の保守における役割
3.7.12.4	文書 (変更管理, 版管理など)
3.7.12.5	構成管理のための組織構造
3.7.12.6	構成管理計画
3.7.12.7	構成管理ツール
3.7.13	システム開発の品質保証
3.7.14	プロジェクトの追跡 (例: PERT, ガント, CPM, CC)
3.7.15	プロジェクトの完了
3.7.15.1	収支報告
3.7.15.2	予定実績差異分析, 振り返りと知識化
3.8	情報とビジネスの分析
3.8.1	問題点と機会の認識 (例: サービスの要求, 計画のプロセス)
3.8.1.1	ソフトシステム方法論
3.8.1.2	問題点ネットワーク (因果図, 魚骨図)
3.8.1.3	因果ループ図
3.8.1.4	計画立案とフィージビリティスタディ
3.8.1.5	ワークデザイン, 品質機能展開, ものごと分析
3.8.1.6	ビジネスプロセスと表現手法
3.8.2	企業モデル
3.8.2.1	Zachman フレームワーク
3.8.2.2	企業モデル (ISO14258) へのアプリケーションの関連づけ
3.8.2.3	エンタープライズアーキテクチャ
3.8.3	要求定義と仕様化
3.8.3.1	機能要求
3.8.3.2	非機能要求
3.8.3.3	要件定義
3.8.3.4	要求の分析と理解
3.9	情報システム設計
3.9.1	設計 (論理, 物理)
3.9.1.1	システム設計方式とツール
3.9.1.2	ソフトウェア設計とシステム設計の役割
3.9.1.3	システム性能と柔軟性に関するハードウェアソフトウェアのトレードオフ
3.9.1.4	ハイレベルインタフェースの設計 (ハードウェアとソフトウェア, ソフトウェアとソフトウェア)
3.9.1.5	システム性能の予測
3.9.1.6	システムのモデル化技法と表現
3.9.1.7	オブジェクト指向のシステム設計技術
3.9.1.8	システム設計技術 (反復設計技術, モデル化など)
3.9.1.9	システム設計の柔軟性

3.9.1.10	開発方式 (例: アジャイル, RAD, ラピッドプロトタイピングなど)
3.9.2	設計方法論 (リアルタイム, オブジェクト指向, 構造化, イベントドリブン)
3.9.3	設計目標 (例: ユーザビリティ, 性能)
3.9.4	創造的な設計プロセスを促進する技術
3.9.5	情報表現の代替案, 認知スタイル
3.9.6	人間とコンピュータの相互作用 (例: エルゴノミクス, グラフィカルユーザインタフェース, 音声, タッチパネル)
3.9.6.1	ユーザインタフェース (音声, タッチ ...)
3.9.6.2	エルゴノミクス
3.9.6.3	共通のユーザアクセス
3.9.6.4	ユーザインタフェース: メニューシステム, コマンド言語, 直接操作, 共通インタフェースのツールキット
3.9.6.5	グラフィックス出力装置とその特性
3.9.6.6	グラフィックスの原理とその特性
3.9.6.7	グラフィックスソフトウェアシステム: 汎用グラフィックス標準
3.9.6.8	ウィンドウマネージャとユーザインタフェースの構造
3.9.6.9	ツールボックスとプログラム支援環境の構造
3.9.6.10	グラフィックデータと音の表現法
3.9.6.11	人間とコンピュータのインタフェースに関する設計技術 (装置の独立性, バーチャルターミナルなど)
3.9.6.12	人間とコンピュータのインタフェースに関する人間工学 (ユーザのクラスについての仮説, 入力エラーの処理, 画面設計など)
3.9.7	ソフトウェア開発
3.9.7.1	ソフトウェアの要求: 原則; タイプ (機能, 性能, その他); 分析: 確認の技術 (プロトタイピング, モデル化, シミュレーション); 顧客とのコミュニケーション: ツール
3.9.7.2	ソフトウェア仕様: 目的; 標準; タイプ (機能, 性能, 信頼性, その他); 形式モデル; 表現; 文書 (標準, 構造, 内容, ユーザ, 完全性, 一貫性); 技術; 品質属性の仕様: 形式仕様言語とツール
3.9.7.3	ソフトウェア設計: 設計の原則 (抽象, 情報隠蔽, モジュール性, 再利用, プロトタイピング); よく知られたシステムの枠組み; 設計の水準; 文書; 設計の説明; サブシステムの設計; 設計の品質の評価; 言語とツール; 方式, 実施及び技法
3.9.7.4	ソフトウェアの品質保証 (問題点, 定義, 標準, 対象分野としての品質保証, 品質に影響を及ぼす要因, SDLC のフェーズにおける品質, 計測, 品質保証のための組織構造, 計画, 文書, 品質保証プロジェクトチーム, 品質とセキュリティ, 産業での実践)
3.9.7.5	ソフトウェアの正当性と信頼性: 原則, 概念, モデル化, 方式
3.9.7.6	ソフトウェア品質保証の検証と妥当性: 役割と方式, 形式モデル, 独立した検証と妥当性を確かめるチーム, ツール, 報告
3.9.7.7	ソフトウェアの実装 (実装とソフトウェア設計の関係, ソフトウェアの実装過程とプログラミング支援環境の関係, 設計の原則と実装言語の関係, ツール, 評価 (コーディング標準, メトリクスなど)), 実装におけるその他の検討事項 (言語構造とプログラミング技術, 再利用, アプリケーションジェネレータ, など)
3.9.7.8	ソフトウェアとハードウェアシステムの統合: 方式, 計画, テスト (開発時の増殖テストも含む), テスト結果の評価と文書化, システムの欠陥の診断, ハードウェアの欠陥のシミュレーション
3.9.7.9	ソフトウェアのテスト: 役割, 原則と標準; テストと品質保証の関係; 方式; テストの水準 (ユニット, システム, 統合, 受け入れなど); 計画, 監査; 限界; 統計的な手法; 形式モデル; 文書; ツール; テストと評価のチーム; テスト環境の構築; テストケースの生成: 回帰テスト; ブラックボックスあるいはホワイトボックスのテスト; 技術的なレビュー; 性能の分析; 結果の分析と報告
3.9.8	ソフトウェアアーキテクチャ
3.9.8.1	ソフトウェアパターン (例: デザインパターン, アーキテクチャパターン)
3.9.8.2	アナリシスパターン
3.9.8.3	ミドルウェア
3.9.8.4	クライアントサーバ
3.10	システムの実装とテスト戦略
3.10.1	システムの構築
3.10.2	ソフトウェアシステムの構築 (例: プログラミング, 単体テスト, ロードモジュールのパッケージ化)
3.10.3	ソフトウェアの統合 (例: パッケージ)
3.10.4	システム移行 (アプローチ, 計画, 実装)
3.10.5	システム統合とシステムテスト: 検証と妥当性, テスト計画の生成, テスト (受け入れテスト, 単体テスト, 結合テスト, 回帰テスト)

3.10.6	訓練（例：ユーザ，管理，運用，システム，訓練用教材）
3.10.7	ソフトウェアプロジェクトの管理：スコープの設定，スケジューリング，構成管理，品質保証：ソフトウェアの信頼性の問題（安全，責任，リスク評価）；メンテナンス
3.10.8	システムのインストール
3.10.9	実装後のレビュー
3.11	システムの運用と保守
3.11.1	サービス要請と変更管理
3.11.2	リバースエンジニアリングとリエンジニアリング
3.11.3	調整と均衡化
3.11.4	システムとソフトウェア保守の概念
3.11.4.1	ソフトウェア保守の種類（認知，修繕，修正）
3.11.4.2	保守のためのソフトウェア設計
3.11.4.3	ソフトウェア保守技術（プログラムの読み込み，リバースエンジニアリング）
3.11.4.4	ソフトウェアの保守モデル
3.11.5	情報システムの評価
3.12	さまざまな情報システムの開発
3.12.1	トランザクション処理システム
3.12.2	経営情報システム
3.12.3	グループ支援システム
3.12.4	意思決定支援システム/エキスパートシステム
3.12.5	エグゼクティブ情報システム
3.12.6	オフィスシステム
3.12.7	協調作業システム
3.12.8	画像及びワークフローシステム
3.12.9	機能的な支援システム（例：プロセス制御，マーケティング）
3.12.10	企業間連携システム
3.12.11	生産管理システム，BPR
3.13	IS 教育と環境
3.13.1	IS リテラシ，メディアリテラシ
3.13.1.1	人間の情報処理
3.13.1.2	人間の記憶構造
3.13.1.3	情報開示と情報保護
3.13.1.4	情報メディアの選択，ツールの選択
3.13.1.5	組織活動と IS
3.13.1.6	人間の科学
3.13.1.7	IS の事例，失敗の事例
3.13.1.8	電子図書館
3.13.1.9	e ラーニング
3.13.1.10	数学的基礎（集合論，パターン処理）
3.13.1.11	ポートフォリオ
3.13.1.12	シミュレーション（例：ゲーム，エージェント，モンテカルロ）
3.13.1.13	バーチャルリアリティ
3.13.1.14	コンピュータビジョン
3.13.2	IS の歴史と進化
3.14	IS 人材の育成
3.14.1	3.8.2.x へ移動
3.14.2	人材のスキル標準（ITSS，ETSS，UISS）
3.14.2.1	学習形態と学習環境
3.14.2.2	オンラインテスト（CBT）
3.15	教育方法論
3.15.1	インストラクショナルデザイン
3.15.1.1	教育評価
3.15.2	オーサリング
3.16	その他の参照学問
3.16.1	経済学
3.16.1.1	経済活動と市場原理
3.16.1.2	経済と政策