

# 情報処理試験の基礎知識

smeokano

## はじめに

---

この本は、情報処理試験に必要と思われる基礎知識を、大雑把に全体像が見えるように、体系化して書くことを目標に作られています。ですから、すべての用語について挙げられているわけではないのですが、全体像は見通せると思います。

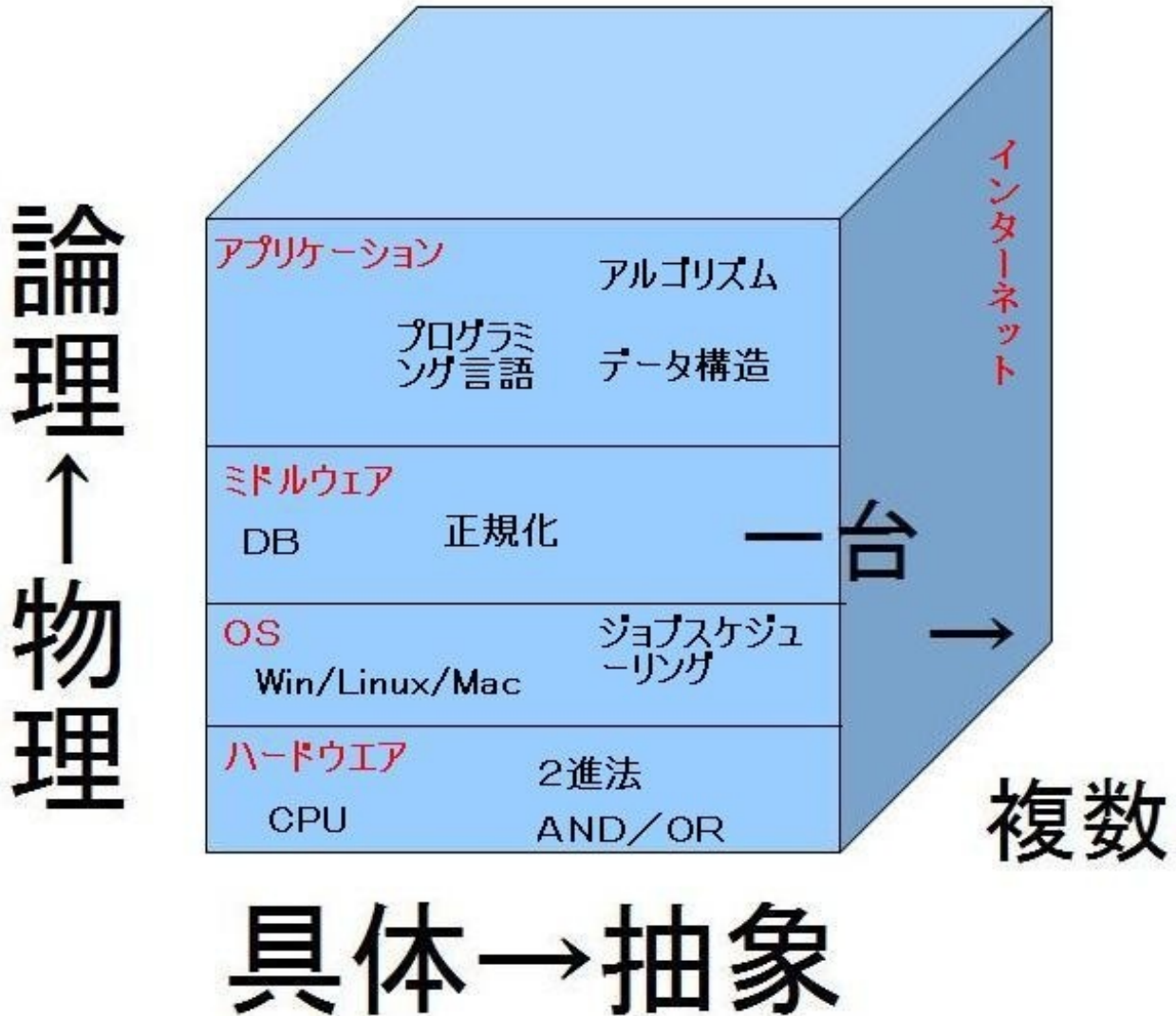
ITパスポートから、基本情報の午前中の試験に、対応していると思います。

内容は、ブログの「実践！情報処理試験講座」 ([http://d.hatena.ne.jp/sme\\_okano/](http://d.hatena.ne.jp/sme_okano/)) や、メルマガ「試験と実務の関連を説明する－実践！情報処理試験講座 メール版」 (<http://www.mag2.com/m/0001289711.html>) の中から抜粋、一部修正しています。

では、早速、はじめていきたいと思います。

情報処理試験は、大きく、テクノロジ系、マネジメント系、ストラテジ系の3つの分野に分かれます。

このうち、テクノロジ系関係の全体像、すなわち、コンピューターとネットワークについては、以下の図のような、3つの軸で考えるとわかりやすいと思います。



\* 黒字で書いてある用語は、例であり、これ以外もたくさんある

●ひとつは物理と論理の軸。

物理はハードウェア、その上にソフトウェアが載ります。ソフトウェアはOS、ミドルウェア、アプリケーションの順に載っていきます。

●もうひとつの軸は、具体的と抽象的の軸。

具体的なものとしては、ハードならCPU,パソコンなど、ソフトなら、ソースコードとかになってきます。

これが抽象的になると、基礎理論の話になります。

ハードとしては、  
具体的には、CPUの話や、スーパースカラとか、そういう話。

抽象論になってくると、  
データ表現という観点では、  
浮動小数点表示や、文字コード、さらにそれを表現  
する2進法とかになります。

処理という観点では、  
AND、OR演算とかになってきます。

ソフトウェアは、データ構造とアルゴリズムに分けられます。  
アルゴリズムは、ソートなど各種の技法、  
データ構造は、リストなどの各種構造、さらに、集合論にまでいきます  
データ構造とアルゴリズムをまとめて、処理を行う手順を記述したものがコンピューター  
言語であり、  
コンピューター言語は、  
記法の話、  
逆ポーランド記法、BNF記法という  
正規表現、  
(図として表現する)状態遷移図の話になり、  
オートマトンの話につながっていきます。  
という構造になります。

●もうひとつの軸は、1台か、複数台か？という軸です。

1台だとスタンドアロン、  
2台だと、冗長構成の話、具体的にはRAID,抽象的には稼働率  
複数台だと、ネットワーク、具体的には、インターネットなどがあります。  
インターネットにおいては、物理～論理までの軸として、OSIの7階層モデルがあり、  
具体的にはルーターやスイッチなどの機器の話、  
論理的には、プロトコルや規約の話などになります。

それでは、以降、ハードウェアから詳しく見ていきます。

## ハードウェアの体系

---

ハードウェアの話の前に・・・

ところで、コンピューターは、なぜ動くんでしょうね？

・・・電気を入れたからです

(あつたりまえじゃん！と思った人、あなたはこの話を最後まで読むと、驚愕の事実を知る)

そう、たしかに電気がはいったから、コンピューターは動いたのですが、  
電気が入りさえすれば、いいのでしょうか？

ドライバーは、計算しますか？

するわけねーじゃん、バカかお前？

と思った人・・・

もし、

1. 温風が当たったら電気が入るようにして→入力
2. その電気が入った回数を覚えておいて→記憶
3. 入った回数を足して行って→演算
4. 足した数だけ、電球を光らせたなら→出力

どうでしょう。ドライバーを使って、計算した・・・と、言えなくもない??

ということは、計算のような情報処理の本質は、

1. 入力
2. 記憶
3. 演算
4. 出力

そして、これらを結んでコントロールする

5. 制御

にあるといえます。これが、コンピューターの5大機能です。

以下、これらについて、もう少し詳しく、見ていきます。

## ●入力・出力

入力は、キーボード、マウスの時もあります。

各種センサーによって、電氣的に値を得ることもあります。

加速度センサーとか、温度センサとか。

センサーからの入力は、デジタルデータの場合もあるけど、アナログデータの場合もあります。

アナログデータの場合、コンピューターでは、デジタルデータで扱うので、アナログ→デジタル変換（AD変換）を行います。

標本化→量子化→符号化

という変換を経て、デジタルデータに変えていきます。

出力は、ディスプレイなどがありますが、ロボットなど、機械制御の場合、モーター等のアクチュエーターの場合があります。

また、アナログ機器に出力する場合は、デジタル→アナログ変換（DA変換）を行います。

入出力装置の具体的なものとして、ハードディスクがあります。

ハードディスクに関しては、昔は、平均アクセス時間などが、情報処理試験に出たのですが、いまは、それよりも、RAIDについて知っていたほうがいいでしょう（SANとかも）。

## ●記憶・演算・制御

さて、入力データで入ってきたデータは、記憶装置に保存されることになります。そして、演算装置で計算などを行います。

記憶装置には、主記憶とキャッシュがあり、この平均アクセス時間を求めるのが、情報処理試験で出たりします。

演算に関して、同じような計算問題として、命令の処理時間の話があります。  
MIPS等が出てくる計算問題です。

記憶・演算・制御は、具体的には、CPU,MPU、チップセットの話になります。  
逆に抽象的な、理論的な話になると、データのほうに関しては、データを  
コンピューター上でどう表現するかという話として

文字=文字コード

整数=二進数・2の補数表示（マイナスのとき）

実数=浮動小数点表示

という話になり、最終的には、1と0の二進数の話になります。

一方、データでなく、処理内容のほうをどう表現するか？という話になると、  
結局コンピューター処理は、AND,ORなどの論理演算の話になってきます。

制御の話として、演算を介さないでメモリアクセスするDMAなんていう話も  
ありますね。

●ちなみに・・・

コンピューターの5大機能があればいいのなら、別に電気を使わなくても  
いいことになります。

1. 5回回す、その後7回回す
2. 回されたことを覚えて置く
3. 元に戻る
4. 回った回数を数える。

これを機械仕掛けでやれば・・・計算機？

実は、そういう計算機、あります。

タイガー手回し計算機

というもので、コンピューターが出るよりも昔に使われていました。





## OSの体系

---

OSは、アプリケーションとハードウェアの間に立って、管理をするところです。

ハードウェアには、5大機能というのがありました。

- ・ 入力
- ・ 記憶
- ・ 演算
- ・ 制御
- ・ 出力

じゃあ、それぞれ管理して

- ・ 入力管理
- ・ 記憶管理
- ・ 演算管理
- ・ 制御管理
- ・ 出力管理

とすればいいかというと・・・

まず、「制御管理」って、何するんだか、よくわかんないので、これはない。

「入力管理」と「出力管理」は、まとめて「入出力管理」としましょう。

「演算管理」とは言わずに、ジョブ管理とか、タスク管理とか、スケジューラーとか言うのが、ふつうでしょうか？ここでは、ジョブ管理とよびますね。

「記憶管理」もメモリ管理とよびましょうか・・・

なので、まとめると、OSの機能は

- ・ 入出力管理
- ・ メモリ管理
- ・ ジョブ管理

となります。

ジョブ管理が指すものとしては、スケジューリングになります。

優先度をもとに、どのジョブを走らせるか、待ちにするかを決定します。

割り込みなんて話が、ここの話です。

メモリ管理に関しては、どこにデータを配置するかということで、ページングなどが話題となります。

ページイン、ページアウト、LRUなど・・・仮想メモリの話ですね。

入出力に関して。ハードのところでは、「データ」と「処理内容」と分けて話していましたが、入出力的には、同じ「ファイル」というもので表現されます。そこで、「ファイル管理」という話題が出てきます。

そして、ファイルは誰でも見れたり、実行できたりすると困りますので、「権限管理／ユーザー管理／認証」という話題が出てきます。

そうなってくると、セキュリティです。セキュリティは、ミドルウェアやネットワークでもよく出てきますが、OSでも、ちょこっと関連してきます。

また、実務的には、入出力装置とOSをつなぐ、ドライバというのが問題になりますが、出題的には少ないかな・・・？

## データベースの体系

---

順番で来ると、本来は、ミドルウェアの体系なのですが、試験に出るのは、データベースが主なので、ミドルウェアの代わりに、データベースについて書きます。

データベースは、なぜ、必要かについてまずはじめに、考えましょう。

データベースを持たず、アプリケーションごとにファイルをもつと、アプリケーション間で重複して使われるファイルが出ます。

たとえば給与と厚生年金、出退勤のアプリケーションでは、従業員ファイルが重複して使われます。

このような重複して使われるファイルは、一括して管理するほうが便利だし、間違いが少ないです。

そこで、このようなファイルを一括して管理するシステムとしてデータベースができました。

そして、一括して管理するために守るべき特性として、ACID特性があります。

(ACID特性についてはWikipediaなどを参照してください)

ただし、一括して管理するデータは、かならず、ACID特性が求められるか？

というと、そんなこともないわけで(サマリーデータ等で、そういうことがある)

そういう場合には、ACID特性をあえて満たさないような、NoSQLという形のデータベースも好まれます。

---

それはさておき、データベースには、表形式をベースとして、関係で表を結合して必要なデータを得る、リレーショナル型データベースや、親子関係をもとにデータをつリー状に配置する、階層型データベースなどがあります。

前者(RDB)のほうが、ACIDに向いています。

さてここで、RDBについて考えると、3つの世界観(スキーマ)が考えられます

1つめは、具体的にDBをどのように構築するかという内部スキーマ

2つめは、データを概念的に表現、DBにマッピングする、概念スキーマ

3つめは、DBを外部から、どう操作するかという、外部スキーマ

これが、3層スキーマ構造です。

●内部スキーマは、DBの物理的構造のお話です。

DBは

- ・ 早く検索するためのインデックス  
→ハッシュとB木
- ・ データそのもの
- ・ トランザクションのためのファイル  
ログファイル（実際には、undoファイルとか）

などがあります。データベースに障害が起きた時の処理（ロールフォワード）  
何らかのアクシデントでなかったことにしたい処理(ロールバック処理)、  
などが重要です。

●概念スキーマにおいては、重要なのは、正規化です。

第一正規形：重複をなくす

第二正規形：主キーをきめて、主キーが複数の項目からなっている場合、  
そのうちの1項目が決まると、キー以外の他の項目が決まらないか？

第三正規形：

主キー以外で、1項目が決まると、キー以外の他の項目が決まらないか？

などのチェックをします。

ちなみに、正規形はこのほかに、3. 5、4、5とありますが、  
4、5正規形は別の視点からの話であり、3. 5はやりすぎです。

●外部スキーマは、アプリケーションから見える形という話ですが、  
（外部スキーマに入れるべきかどうかわかりませんが、とにかく）、  
外部から操作するといえば、SQLです。

SQLが、一通りできることが大事です。

データベースの話としては、こんなところです。

特に、SQLと正規化はできる必要があり、正規化のまとめとして描かれるER図も読める必要が  
あります。

なお、DBが、2つ3つとなってくると、分散データベースで、2相コミットの話も出てきます

。

## アプリケーションの体系

---

基本情報でアプリケーションの操作といえば、表計算が出題されますが（選択ですが）、応用情報の場合、アプリケーションの操作を聞く問題はないです。

具体的なアプリケーションの(操作ではなく)機能・内容を問う問題としては、SMTPやPOPなど、ネットワーク関係のもので、あります。これは、ネットワークの話ということで、今回、説明は割愛します。

また、アプリケーションの開発について問う問題は、当然出ます。

ただし、アプリケーションそのものではないので、これも、今回の話からは外します。

ここで話すのは、アプリケーションの基礎的な部分の話である言語とか、最近話題のオープンソースについて等の話です。

### ●コンピューター言語の話

アプリケーションを開発するには、コンピューター言語（JAVAとか、COBOLとか）で記述し、インタープリタ、またはコンパイル&リンクを経て実行ということになります。

このコンピューター言語ですが、具体的な内容は、基本情報の午後の話題です。しかし、上記のコンパイルから実行までの話や、コンピューター言語における特徴（オブジェクト指向の言語はどれか）とか、言語の背景にある思想（オブジェクト指向って何？構造化手法とは？とか）は午前の範囲です。

応用情報では、言語が成立するための元となる理論、すなわち、BNF記法や逆ポーランド記法、そして、正規表現、さらに正規表現がオートマトンで表現できる話などが出題範囲になります。

さらに遡り、オートマトンと状態遷移の話から、チューリングマシンの話というふうに、結局、計算論まで行ってしまう話が範囲となります（普通、教えるときは、オートマトンから、逆に教えていきますが・・・）。

いわゆるコンピューターサイエンスが、応用情報の範囲です。

### ●一方、オープンソースに関しては、契約関係が、主になります。

GPLとは、どんなライセンスで、どんなことは、してもよくて、どんなことをしてはいけないか・・・などです。

なお、リポジトリについての詳しいこととか、コミュニティでうまくやっていくにはどうしたらいいかとか、OSCの会場の行き方とか、そんなことは聞かれません（あたりまえです）。

## 2台以上のコンピュータ（＝ネットワーク）の体系

---

いままでは、1台のコンピュータでしたが、今回は、2台以上のコンピュータについて、考えます。

情報処理試験で、2台以上のコンピュータについての話題というと、

- ・ 数台の構成
  - ・ 冗長性の話：RAID等
  - ・ 信頼性のはなし：稼働率、MTBF、MTTRなど
  
- ・ たくさんの構成
  - ・ ネットワーク
    - ・ インターネット
    - ・ その他
      - 電話
      - HDLC
      - ：

ということになります。

ただし、ネットワークは、インターネット中心です。

また、数台の構成の話は、上記に書いた程度の話しかないので、以下、インターネットの話をしていきます。

●インターネットの世界はOSIの7階層モデルがすべてです。

アプリケーション層

プレゼンテーション層

セッション層

トランスポート層：TCP、ポート番号

ネットワーク層：IP、IPアドレス、機器はレイヤー

データリンク層：イーサネットの世界・MACアドレス、機器はスイッチ

物理層：電気の世界、電線など

とくにIP層に関しては、IPのかなりの知識、たとえば、ルーティング（ディスタンスベクタのRIP,リンクステートのOSPFなど）かなり深いところまでやらないといけません。

OSIに関しては、1～3層においては、装置とプロトコル両方、4層はプロトコル、5層目以上は、大雑把に見ておけば良いです。とくに、3層のIPが重要になります。

## 会社と開発などの体系

---

いままで説明してきた体系は、コンピューターシステムについてのものでした。  
しかし、コンピューターシステムというのは、ある日突然できるものではありません。

このようなコンピューターシステムを作るには、

要求仕様  
外部設計  
詳細設計  
実装（プログラミング）  
単体テスト  
結合テスト  
総合テスト

などの、ソフトウェア開発を行って、やっとできます。

これらのソフトウェア開発プロセスは、ソフトウェア工学といわれる分野であり、情報処理試験のシラバスのには、開発技術といわれる分野です。

そして、ソフトウェア開発を行うには、マネジメントを行います。

ソフトウェア開発の内容を決め（スコープを決めて）、それに基づき、人、モノ、カネをそろえていきます。これが、プロジェクトマネジメントです。

また、できたシステムは、保守・運用していく必要があります。

そして、ユーザーのクレーム等に対しても、対応する必要があります。

これが、サービスマネジメントとなります。

また、できたシステムは評価する必要もあるかもしれません。

これが、システム監査になります。

つまり、コンピューターシステムを作り、運用するには、

- ・ソフトウェア開発
- ・プロジェクトマネジメント
- ・サービスマネジメント
- ・システム監査

などが、必要になります。

しかし、よくよく考えると、「コンピューターシステムを作り、運用する」前に、システムを作ろう！ということを誰かが考えない限り、システムは作られないはずで



この、「システムを作ろう！」と考える前に、

- ・ 経営理念を元に、
- ・ 長期、中期計画、年度計画などが策定され、
- ・ その結果、システム開発が必要と決まって、

という意味決定がなされます。この部分を扱うのが、「経営戦略」であり、この経営戦略を、よりシステムの的に考えたものが、「システム戦略」になります。

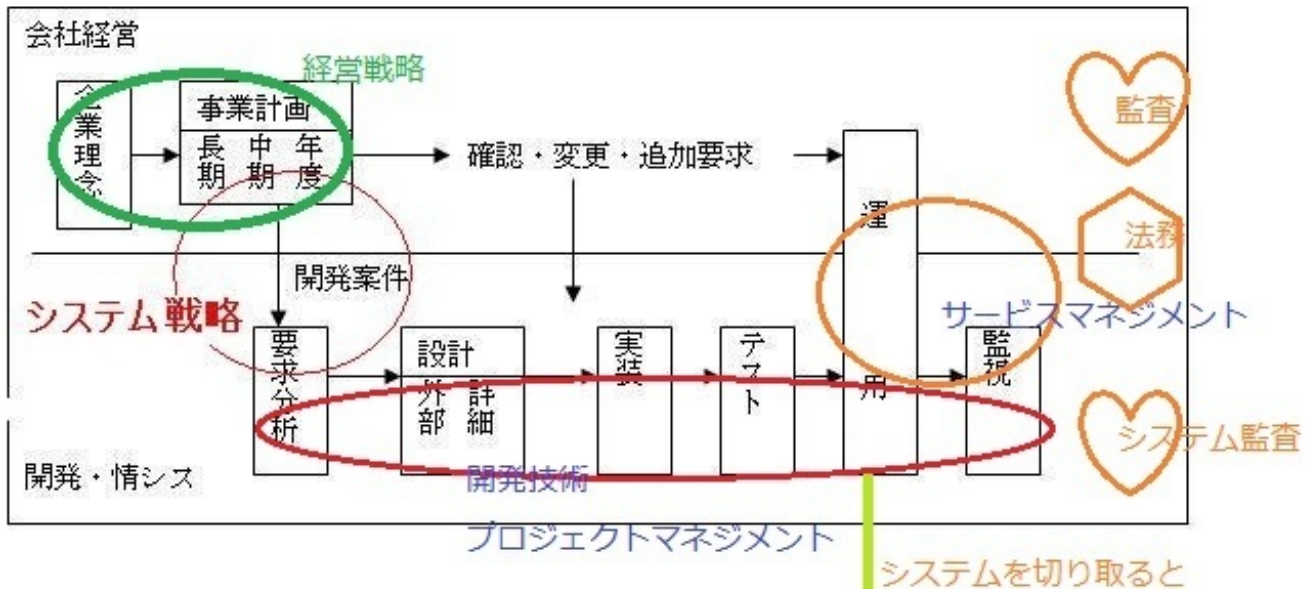
また、経営戦略に対する制約として、各種法規、すなわち、法務があります。

つまり、コンピューターシステムを作ると決めるのに、

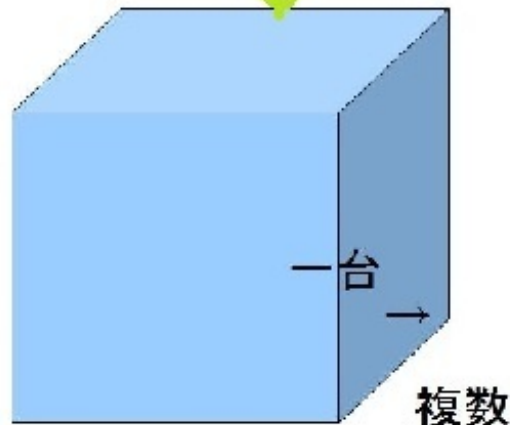
- ・ 経営戦略
- ・ システム戦略
- ・ 法務

などが、必要になります。

これらをまとめると、以下の図のようになります。



論理  
↑  
物理



具体→抽象

以降、開発技術から見ていきますが、ここ以降の分野は、今までとは少し違うので、それについて説明します。

応用情報処理試験のシラバスの大項目を、こんなふうに分けることもできます。

(1) なにか、基本的なドキュメントが存在し、それに沿って出題

開発技術：共通フレーム

プロジェクトマネジメント：PMBOK

サービスマネジメント

サービスマネジメント：ITIL

システム監査：システム監査基準、システム管理基準

(2) そういうものはない

→ほかの分野

今後話題とする「開発技術」等は、（１）に属します。そして、（１）の勉強法は、

- ・基本となっている文書を読む。
  - ・ ・のは、たいてい大変なので、簡単に書いてある本で、概略をつかむ
- ・自分の仕事等と関係していることと、基本文書との対応を考える
- ・足りない分野を、参考書などで補う
- ・過去問で総仕上げ

となります。

なので、以降、基本となっている文書との関係を中心に書いていきます。

情報処理試験の「開発技術」は、ソフトウェア工学といわれる分野です。

ソフトウェア工学は、ソフトウェア開発全体のプロセス等全体に関することと、ソフトウェア開発の各プロセスの詳細についてのお話です。

応用情報処理試験のシラバスの「開発技術」には、「システム開発技術」と「ソフトウェア開発管理技術」がありますが、これらは、

ソフトウェア開発の全体のプロセスなどを、「ソフトウェア開発管理技術」とし、

ソフトウェア開発の各プロセスに関することを「システム開発技術」として

分けています。

●「システム開発技術」は、開発の各工程に基づき、以下のように応用情報のシラバスでは、詳細化されています。

### 1. システム要件定義

- (1) システム要件定義のタスク
- (2) システム要件の定義
- (3) システム要件の評価

### 2. システム方式設計

- (1) システム方式設計のタスク
- (2) システムの最上位レベルでの方式確立
- (3) システム結合テストの設計
- (4) システム方式の評価

### 3. ソフトウェア要件定義

- (1) ソフトウェア要件定義のタスク
- (2) ソフトウェア要件の確立
- (3) ソフトウェア要件の評価
- (4) 業務分析や要件定義に用いられる手法

### 4. ソフトウェア方式設計・ソフトウェア詳細設計

- (1) ソフトウェア方式設計のタスク
- (2) ソフトウェア詳細設計のタスク
- (3) ソフトウェア方式設計
- (4) ソフトウェア詳細設計

- (5) インタフェース設計
- (6) ソフトウェアユニットのテストの設計
- (7) ソフトウェア結合テストの設計
- (8) ソフトウェア設計評価
- (9) ソフトウェア品質
- (10) ソフトウェア設計手法
- (11) コンポーネントの設計
- (12) モジュールの設計
- (13) 部品化と再利用
- (14) デザインパターン
- (15) レビュー

## 5. ソフトウェアコード作成及びテスト

- (1) ソフトウェアコード作成及びテストのタスク
- (2) ソフトウェアコード作成
- (3) ソフトウェアコード及びテスト結果の評価基準
- (4) コーディング基準
- (5) コードレビュー
- (6) デバッグ
- (7) ソフトウェアユニットのテスト

## 6. ソフトウェア結合・ソフトウェア適格性確認テスト

- (1) ソフトウェア結合のタスク
- (2) ソフトウェア適格性確認テストのタスク
- (3) ソフトウェア結合テスト
- (4) ソフトウェア適格性確認テスト
- (5) テスト結果の評価

## 7. システム結合・システム適格性確認テスト

- (1) システム結合のタスク
- (2) システム適格性確認テストのタスク
- (3) システム結合テスト
- (4) システム適格性確認テスト
- (5) テスト結果の評価

## 8. ソフトウェア導入

- (1) ソフトウェア導入のタスク

- (2) ソフトウェア導入計画の作成
- (3) ソフトウェア導入の実施
- (4) 利用者支援

## 9. ソフトウェア受入れ

- (1) ソフトウェア受入れ支援のタスク
- (2) 受入れレビューと受入れテスト
- (3) ソフトウェア製品の納入と受入れ
- (4) 教育訓練
- (5) 利用者マニュアル

## 10. ソフトウェア保守

- (1) ソフトウェア保守の意義
- (2) ソフトウェア保守の形態
- (3) ソフトウェア保守の手順

このプロセスの順番は、共通フレーム2007（SLCP-JCF2007）に基づいています。

各プロセスのはじめに、

～のタスク

というのがあり、実際に行うアクティビティは、その下から、

～（の）評価

と書かれている間になります。その後があるものについては、手法、技法について書かれています。

●ソフトウェア工学などと言われる開発工程と対応させると、

3. ソフトウェア要件定義  
は、要求分析

4. ソフトウェア方式設計・ソフトウェア詳細設計  
の「ソフトウェア方式設計」が外部設計（基本設計）

の「ソフトウェア詳細設計」が詳細設計

5. ソフトウェアコード作成及びテスト  
がプログラミング&単体テスト
6. ソフトウェア結合・ソフトウェア適格性確認テスト  
が結合テスト
7. システム結合・システム適格性確認テスト  
が総合テスト
8. ソフトウェア導入  
が移行（導入）準備
9. ソフトウェア受入れ  
が導入・運用
10. ソフトウェア保守  
が運用・保守

となります。

そうすると、「1.システム要件定義」と「2.システム方式設計」が残りますが、これは何か？  
というと、ハードなどを含んでいる場合（組み込みなど）、ソフトハード全体の設計をする場合  
の話で、ソフトだけで考えれば、RFPに載ってくるような話です。

●一方、各論ではなく、全体の話、「ソフトウェア開発管理技術」には何が書いてあるかとい  
うと、こんなかんじです。

1. 開発プロセス・手法
  - (1) ソフトウェア開発手法
  - (2) 構造化手法
  - (3) 形式手法
  - (4) マッシュアップ
2. 知的財産適用管理
  - (1) 著作権管理
  - (2) 特許管理

### (3) ライセンス管理

## 3. 開発環境管理

- (1) 開発環境構築
- (2) 管理対象

## 4. 構成管理・変更管理

- (1) 構成管理
- (2) 変更管理

「ソフトウェア開発手法」のところに、ウォーターフォールとか、アジャイルなどの手法が挙げられています。

なお、「構造化手法」があっても、オブジェクト指向がないのは、オブジェクト指向は、各論である、「システム開発技術」の「3. ソフトウェア要件定義」の「(4) 業務分析や要件定義に用いられる手法」でUMLが、「4. ソフトウェア方式設計・ソフトウェア詳細設計」の「(10) ソフトウェア設計手法」で、オブジェクト指向設計が出てきているので、あえて、ここでは書いていないものと思われます。

●このように、情報処理試験では、ソフトウェア開発工程は、共通フレーム2007 (SLCP-JCF2007) をもとに、構造化手法、オブジェクト指向共に対応した内容となっています。



## プロジェクトマネジメントの体系

---

プロジェクトマネジメントについては、ベストプラクティスをまとめたPMBOKというものがあります。

プロジェクトマネジメントにおいては、このPMBOKが基本的な文書となっています。

PMBOKは、プロジェクトにおける5つの局面

- A. 立ち上げプロセス群
- B. 計画プロセス群
- C. 実行プロセス群
- D. 監視・コントロール・プロセス群
- E. 終結プロセス群

(各局面を「プロセス群」と呼んでいます) について、そこで行われる42個のプロセスを、42個のままだと扱いにくいので、9個のナレッジエリア

- 4. プロジェクト統合マネジメント
- 5. プロジェクト・スコープ・マネジメント
- 6. プロジェクト・タイム・マネジメント
- 7. プロジェクト・コスト・マネジメント
- 8. プロジェクト品質マネジメント
- 9. プロジェクト人的資源マネジメント
- 10. プロジェクト・コミュニケーション・マネジメント
- 11. プロジェクト・リスク・マネジメント
- 12. プロジェクト調達マネジメント

にわけ、そこに42個の各プロセスを割り当てて、(各プロセスは、必ずどれかのナレッジエリアに属する) 各プロセスにおけるインプットアウトプットとツールと技法について、説明したものです。

(数字が4から始まっているのは、PMBOKの章に対応させたからです。つまり、ナレッジエリアの説明は4章から始まっています)

●それぞれのプロセス群、プロセスと、ナレッジエリアの関係は、以下のとおりです(数字は上記のナレッジエリアの番号)。

・ 立ち上げプロセス群

4. プロジェクト統合マネジメント

4. 1 プロジェクト憲章作成

10. プロジェクト・コミュニケーション・マネジメント

10. 1 ステークホルダー特定

・ 計画プロセス群

4. プロジェクト統合マネジメント

4. 2 プロジェクトマネジメント計画書作成

5. プロジェクト・スコープ・マネジメント

5. 1 要求事項収集

5. 2 スコープ定義

5. 3 WBS作成

6. プロジェクト・タイム・マネジメント

6. 1 アクティビティ定義

6. 2 アクティビティ順序定義

6. 3 アクティビティ資源見積もり

6. 4 アクティビティ所要期間見積もり

6. 5 スケジュール作成

7. プロジェクト・コスト・マネジメント

7. 1 コスト見積もり

7. 2 予算設定

8. プロジェクト品質マネジメント

8. 1 品質計画

9. プロジェクト人的資源マネジメント

9. 1 人的資源計画書作成

10. プロジェクト・コミュニケーション・マネジメント

10. 2 コミュニケーション計画

11. プロジェクト・リスク・マネジメント

- 1 1. 1 リスク・マネジメント計画
- 1 1. 2 リスク特定
- 1 1. 3 定性的リスク分析
- 1 1. 4 定量的リスク分析
- 1 1. 5 リスク対応計画

## 1 2. プロジェクト調達マネジメント

- 1 2. 1 調達計画

### ・実行プロセス群

#### 4. プロジェクト結合マネジメント

- 4. 3 プロジェクト実行の指揮・マネジメント

#### 8. プロジェクト品質マネジメント

- 8. 2 品質保証

#### 9. プロジェクト人的資源マネジメント

- 9. 2 プロジェクト・チーム編成
- 9. 3 プロジェクト・チーム育成
- 9. 4 プロジェクト・チームのマネジメント

#### 1 0. プロジェクト・コミュニケーション・マネジメント

- 1 0. 2 情報配布
- 1 0. 4 ステークホルダーの期待のマネジメント

## 1 2. プロジェクト調達マネジメント

- 1 2. 2 調達実行

### ・監視・コントロール・プロセス群

#### 4. プロジェクト統合マネジメント

- 4. 4 プロジェクト作業の監視・コントロール
- 4. 5 統合変更管理

#### 5. プロジェクト・スコープ・マネジメント

- 5. 4 スコープ検証
- 5. 5 スコープ・コントロール

6. プロジェクト・タイム・マネジメント  
6. 6 スケジュール・コントロール・マネジメント

7. プロジェクト・コスト・マネジメント  
7. 3 コスト・コントロール

8. プロジェクト品質マネジメント  
8. 3 品質管理

10. プロジェクト・コミュニケーション・マネジメント  
10. 5 実績報告

11. プロジェクト・リスク・マネジメント  
11. 6 リスクの監視・コントロール

12. プロジェクト調達マネジメント  
12. 3 調達管理

・ 終結プロセス群

4. プロジェクト統合マネジメント  
4. 6 プロジェクトやフェーズの終結

12. プロジェクト調達マネジメント  
12. 4 調達終結

必ず、4章の「プロジェクト統合マネジメント」は各プロセス群にあるようになっています。

●一方、情報処理試験では、例えば応用情報のシラバスにおいては、プロジェクトマネジメントの範囲を以下のよう規定しています。

1. プロジェクト統合マネジメント
  - (1) プロジェクトマネジメントの目的と考え方
  - (2) プロジェクトの体制と自己管理

(3) プロジェクト統合マネジメントの目的と考え方

(4) プロジェクト統合マネジメントのプロセス

## 2. プロジェクト・スコープ・マネジメント

(1) プロジェクト・スコープ・マネジメントの目的と考え方

(2) WBS

(3) プロジェクト・スコープ・マネジメントのプロセス

## 3. プロジェクト・タイム・マネジメント

(1) プロジェクト・タイム・マネジメントの目的と考え方

(2) プロジェクト・タイム・マネジメントのプロセス

(3) 代表的な管理手法

## 4. プロジェクト・コスト・マネジメント

(1) プロジェクト・コスト・マネジメントの目的と考え方

(2) プロジェクト・コスト・マネジメントのプロセス

(3) 代表的なコスト見積手法とコスト管理手法

## 5. プロジェクト品質マネジメント

(1) プロジェクト品質マネジメントの目的と考え方

(2) プロジェクト品質マネジメントのプロセス

(3) 代表的な品質マネジメントの手法

## 6. プロジェクト人的資源マネジメント

(1) プロジェクト人的資源マネジメントの目的と考え方

(2) プロジェクト人的資源マネジメントのプロセス

(3) 外部人材の管理

## 7. プロジェクト・コミュニケーション・マネジメント

(1) プロジェクト・コミュニケーション・マネジメントの目的と考え方

(2) プロジェクト・コミュニケーション・マネジメントのプロセス

## 8. プロジェクト・リスク・マネジメント

(1) プロジェクト・リスク・マネジメントの目的と考え方

(2) プロジェクト・リスク・マネジメントのプロセス

## 9. プロジェクト調達マネジメント

(1) プロジェクト調達マネジメントの目的と考え方

(2) プロジェクト調達マネジメントのプロセス

●つまり、

・ シラバスはPMBOKのナレッジエリアに基づいている

・ 基本的に

(ナレッジエリアの) 目的と考え方

(ナレッジエリアの) プロセス

があり、いろいろと手法がある場合には

代表的な～手法

という項目になっている

・ これにそぐわないエリアは（統合マネジメントは全体の話がはじめにあるが、これは除いて考える）

「2. プロジェクト・スコープ・マネジメント」のWBS

「6. プロジェクト人的資源マネジメント」の外部人材の管理

である。したがって、この2つ、WBSと労働派遣法関連などの、外部人材の管理については、特に重視していると思われる

ということが判ります。

## サービスマネジメントの体系

---

サービスマネジメント、特に、サポートセンター等のマネジメントのベストプラクティスとして、ITIL(Information Technology Infrastructure Library)があります。

ITILは、海外で出来たものであり、それを日本で広めている団体が、itSMFです。じゃあ、itSMFのホームページを見れば、よいのだね・・・というので、ホームページ <http://www.itsmf-japan.org/itil/index.html> をみると、実は、情報処理試験的には、困ったことが起こります。

●そこに書いてあるのは、ITIL V3の説明です。

でも、情報処理試験のシラバスには「サービスデリバリ」といった、ITIL V2で使われていた言葉が出てきています。

一方、日経コンピューター2011年9月29日号によりますと・・・  
「ITIL」が4年ぶりに大改定  
クラウド利用のポイントを解説  
だそうです。

ということで、情報処理試験のためにお勉強するとしたら、いったいどれを、勉強したらいいの？ということになります。

●現状、まだ改定したITIL 2011 Editionは、最近出たばかりなので、これを勉強するという段階にはないと思います。

とすると、ITIL V3とV2についてですが、この違いなどについて、まとまっているサイトがあります。

ITIL V3はV2からどこが変わったか

<http://www.atmarkit.co.jp/im/cop/serial/itilv3/02/01.html>

この辺から、見ていくのが良いのではないかと思います。

●ITIL V2,V3の本は、買おうと思えば買えます。

<http://www.itsmf-japan.org/books/index.html>

でも、高いし、カードでしか買えないし・・・

ということで、情報処理試験の午前の数問のためには、ちょっと・・・だと思います。ITIL Foundationの参考書のほうが良いかな？とも思います。

たぶん、試験的には、コストパフォーマンスのよくない科目ですね。

●さらに、情報処理試験では、サービスマネージメントに、SLAを含みます。

SLAは、最近、SaaS向けのものに関しては、

「SaaS向けSLAガイドライン」

<http://www.meti.go.jp/press/20080121004/20080121004.html>

また、電子情報技術産業協会（JEITA）が出している

民間向けITシステムのSLAガイドライン 第三版（9800円）

<http://coin.nikkeibp.co.jp/coin/nc/books/sla3/>

ITコーディネーター協会から出している

RFP・SLAドキュメント見本

[http://www.itc.or.jp/foritc/useful/rfpsla/rfpsla\\_doui.html](http://www.itc.or.jp/foritc/useful/rfpsla/rfpsla_doui.html)

などが、参考になる・・・かも？かもしれません。

●ただ、情報処理試験のために勉強するのであれば、この分野は、範囲が広く、参考書は高い割りに、問題数が少なく、それも、そんなに勉強しなくても解けてしまう問題が多いです。

コストパフォーマンスから考えると、ここをがんばるより、他のところ（例えば次回説明するシステム監査とか）に力を入れたほうがよいかもしれません。



## システム監査の体系

---

システム監査に必要な知識は、

- ・ システム監査基準
- ・ システム管理基準
- ・ 情報セキュリティ監査基準
- ・ 情報セキュリティ管理基準

に、まとめられています。

システム監査を行う手順などが、「システム監査基準」にまとめられ、実際に監査を行うときに、評価する基準（尺度）が、「システム管理基準」に、チェックシートのような形（項目の羅列）として、まとめられています。

そのうちのセキュリティ部分だけをまとめたものが情報セキュリティ監査（管理）基準です。

### ●システム監査基準は

前文

システム監査の目的

一般基準

実施基準

報告基準

とわかれています。実施内容が書かれているのが、「実施基準」で、

その手順は、

監査計画

予備調査

本調査および評価

結論

の順に行うように書かれています。

監査を行う際には、監査結果を裏付けるのに充分かつ適切な「監査証拠」を入手し、監査手続きの結果と関連資料を「監査調書」としてまとめるように書かれています。

最終的には、監査報告書にまとめますが、その内容は

- ・ 監査の対象

- ・実施した監査の概要
- ・保証意見または助言意見
- ・制約または除外事項
- ・指摘事項
- ・改善事項
- ・その他特記すべき事項

からなります。

●この分野で覚えるべきことは、これくらいです。他分野に比べ、圧倒的に勉強する事柄がすくないです。

なので、試験的に見ると、この分野は、狙いどころといえます。

今まで、システム開発についてみてきました。

しかし、そもそも、どうして会社はシステム開発を行うのでしょうか。

●まず、会社は、「何かを成し遂げよう」という志を持って、立ち上げられるとされています。もちろん、街のラーメン屋さんが、「何かを成し遂げよう」と思って立ち上げるかどうかは、疑問なのですが、それでも、お店を出す、会社を興すには何らかの考えがあったはず（金持ちになるという考えでもOK）。

その志や考えを、「経営理念」と呼びます。

会社はこの経営理念を元に、運営されていくわけです。

この経営理念は、数年で変わるようなものでなく、何十年、場合によっては百年単位で動かないものにします。

●でも、会社は時代に合わせていかなければいけません。そこで、企業理念を元に、まずはざっくり、会社として、こういう方向で動くというのを決めます。これが経営戦略になります。

具体的には、社外の動き、業界とか、日本、世界の動きと、社内の動き、会社の様子とか問題点をまとめます。まとめ方としてはSWOTがあります。

●そして、SWOTなどで、社外の様子と社内の様子をまとめたら、その状況下において、自社は、どうするのかを決めます。

決め方としては、自社の強みを発展させ、他社を圧倒し、まねできないもの（コアコンピタンス）を見つけたり（なかったら作り出し）、それを、主要成功要因（クリティカルサクセスファクター：CSF）とします。

そうしたら、主成功要因を利用して、自分たちが成功するビジネスストーリーを作り出し、そのお話に基づき、数値目標（KGI）／評価指標（KPI）などをつけるのですが、このとき、

「財務の視点（過去）」、

「顧客の視点（外部）」、

「内部業務プロセスの視点（内部）」、

「イノベーションと学習の視点（将来）」

の“4つの視点”からまとめると、バランス（ト）・スコアカード（BSC）になります。

・・・なりますが、ここまではやっていないでしょう。普通。

情報処理試験的にはこうなるのですが・・・

なお、最近流行の成長ドライバーなんていっているのも、基本的には、主要成功要因と同じ

です。

また、今は全社的な話をしましたけど、大きな会社になると、会社をいくつかに分けて、事業ごとに話をしたいわけです。このとき、事業レベルで、どの事業を続けていくかについて考える場合は、PPM等を使います。

●という形で、戦略ができ、これが、中長期計画にまとめられます。

これをさらに落としていくわけですが、ここでは、ざっくりと3つに分けて考えます。

1つは、財務

2つめ、マーケティングと製造

3つめ、人材

いわゆる、人、もの、カネにわけます。

これらの分野について、それぞれの人が、以下に示すような、戦術的要素まで落として考え、年度計画（＝短期計画）にまとめていきます。

では、以降、詳細に見ていきます。

●まずは、財務。

財務は定常的な評価とプロジェクトレベルでの評価を行います。

非通用ならばお金を借り入れて。。。いと色々なことをします。

定常的な評価の方法としては、基本的にBS、PLを使ったものになります。

B/S（貸借対照表）により、その会社の安全性を見たりします。

具体的に見る数字は、流動比率（流動資産／流動負債＝100%より大きいほうがいい）などです。

そしてP/L（損益計算書）をあわせて使って、収益性を見ます。

もちろん、回転率などは高いほうがいいのですが、それよりも、投資家が関心があるのが、ROI、ROE、ROAなどです。

投資したお金などにたいして、どれだけのリターンがあったか、に関心がむきます。

このようなことをして、会社の状況を大きく判断していきます。

あまりにも上記指標が悪いと、銀行などから借り入れられなくなるので、要注意です。

そのほか、会社では、管理しやすいように、「管理会計」というのを行います。ABC (Activity-Based Costing、活動基準原価計算) などが有名です。

また、プロジェクトが行われるとき、採算が合うかどうかを確認します。

本当は、DCF (ディスカウントキャッシュフロー) 法の中の、NPV (正味現在価値) という手法で計算するのですが、めんどくさいので、そこまでやらないのでは・・・いや、やっているのかな？

損益分岐点計算ぐらいは、しますかね？

しかし、財務で一番重要なことは、上記のようなことではなく、交通費を精算し、仕訳をすること・・・でもなく (それもするけど) 会社が動くように、お金を回すことです。資金繰りです。

大きな会社は、キャッシュフロー計算書などをつくりませんが、小さい会社でも、資金繰り表などを作って、資金繰りを管理します。そして、足りなくなったら、お金を借りてきたりします。

指標に注意したり、税金や交通費精算という日常業務をベースに、資金繰りに注意しながら、プロジェクトなどでは、採算性を計算し、場合によっては、融資を受けるようにする。というのが、財務的な役割になります。

●一方、マーケティング、製造についてです。

製造については、作っているものや製造形態によって、かなり違います。一言でまとめてしまえば、生産管理なのですが・・・

一方、マーケティングとしては、まずはマーケット戦略を立てて、誰に、何を売るか決めていきます。

これを行うため、製品を分類し、競合関係を明確にする (製品の規定) と同時に、消費者を細分化していきます。

そして、消費者×製品のマトリックスを作り、どのセグメンテーションにたいして、どのようなターゲットを狙って、どのようなポジショニングに付くかということ (=STP) を決めます。

この際、5Fや3Cは情報処理試験的には役立つと考えられていますが、マーケットリサーチ

によって、多変量解析（因子分析やクラスタリング）を行ったり、データマイニングを行ったほうが、数値的に出るし、はっきりするかな？

そして、マーケティング戦略により、だれに何を売るかが決まったら、どのように売るかを考えていきます。その考え方に役立つのが、4 Pです。

4 Pとは

プライス：価格

プレイス：流通チャネル（かなり違う話だと思うが・・・）

プロモーション：CM，パブリシティ、セールスプロモーション

プロダクト：商品

の頭文字をとったものです。

これらについて、より具体的に考えていくことになります。

ここまでくると、あとは営業さんにがんばってもらうと・・・

●そして、人材です。

会社の理念が決まったら、必要な人材像を決めていきます。

そしたら、その人材をどのように調達するのか、育成するのか、採用するのか決めます。

育成するとしたら、育成できるようなカリキュラム、すなわちキャリアパスを構築し、そのキャリアパスに則って人材を育成していかなければなりません。

業界的には、これをITSSにしたいみたい？

そして、ITSSに対応しているのが、この情報処理試験となります。

ただし、必ずしも社内の生え抜きを活用しなければならないわけではなくBPO（ビジネスプロセスアウトソーシング）や派遣の利用なども考えられます。

●これら、財務、マーケティング、製造、人材などにおいて、コンピューター化したほうがいい部分が出てきます。

その、コンピューター化していく過程が、システム戦略となります

前回、会社全体の「経営戦略」について考えました。

今回は、その経営戦略を受けて、情報システム部としては、どのようにシステム化していくかという、「システム戦略」について考えます。

経営戦略によって、事業戦略という形で、戦略のモデルができます。

システム化するには、この事業戦略（戦略モデル）から、コンピューター化すべき情報を抜き出して、モデル化しないといけません。

この情報モデルがビジネスモデルとなります。ビジネスモデルレベルで、全社最適化を図ります。

このビジネスモデルを考えて、全社最適化の構想を練るのがC I Oの役目となってきます。

●事業戦略を元に、中長期計画が立てられ、それ詳細化する形で、戦術レベルの年度計画が立てられるように、ビジネスモデルも詳細化され、業務モデルとなってきます。

つまり、実際に業務でどのようにコンピューターが必要で、その必要を満たすために、どのようにシステム化をするかを考えていくわけです。

また、そうではなく、市場ニーズをもとに、技術開発をした結果、製品を作る上で、コンピューターが必要だから、システム開発するということもあります。

この場合は技術開発戦略と関わってくることになります。

技術開発計画に、システム開発が関わってくる形になります。

●ちなみに、上記で述べたシステム化の類型としては、情報処理試験的には、以下のように考えているようです

(応用情報シラバス 大分類8：経営戦略 中分類21：ビジネスインダストリより)

### 1. ビジネスシステム

- (1) 社内業務支援システム
- (2) 基幹業務支援システム及び業務パッケージ
- (3) 行政システム
- (4) 公共情報システム

### 2. エンジニアリングシステム

- (1) 生産の自動制御
- (2) 生産システム
- (3) コンピュータ支援システム

### 3. e-ビジネス

- (1) EC (Electronic Commerce : 電子商取引)
- (2) EDI

### 4. 民生機器

- (1) 組込みシステム
- (2) 民生機器

### 5. 産業機器

- (1) 産業機器

くわしい内容は、シラバスに書いてあるので、そちらに任せるとして、ここでは、こんな風に、いろいろあるんだということを、押さえて置いてください（試験を受けるときには、それぞれの重要用語ぐらいは知っておいたほうがいい）

-

●そして、いざ、それらのシステムを開発するとなると、いろいろと調達するわけですが、システムの調達の場合、

- ・ R F P を出す
- ・ 提案してもらう
- ・ 発注会社を決める
- ・ 契約する

という流れになるのが普通だと思います。

発注以降に要求分析となる・・・はずなのですが、実際には、R F P に記載する都合上、R F P 作成時点で、要求分析、要件定義をすることになります。



## 法務の体系

---

情報処理試験のシラバスでは、「企業と法務」というのがあります。

「企業」に関しては、「1. 経営・組織論」、「2. OR・IE」、「3. 会計・財務」なのですが、これらについては、「経営戦略の体系」で触れたので、置いておきます。

ということで、「法務」の分野について、考えて見ます。

●情報処理試験において、「企業と法務」法務の分野は、法律と業界標準に分かれます。

法律というと、普通は憲法が一番ですが、情報処理試験で、憲法はまず出ないと思います。

憲法を含め、基本的な法律として六法があります。

- ・ 憲法
- ・ 民法
- ・ 商法
- ・ 刑法
- ・ 民事訴訟法
- ・ 刑事訴訟法

です。

このうち、まず刑法については、データの改ざんなどが関係するのですが、刑法だと、改ざん等により、被害が出ないと罰せられないです。

それはまずいということで、不正アクセス禁止法などが出てきました。

刑法と一緒にこれら「セキュリティ関連法規」も見ておくといいと思います。

民法は

第1編 総 則（第1条～第174条の2）

第2編 物 権（第175条～第398条）

第3編 債 権（第399条～第724条）

第4編 親 族（第725条～第881条）

第5編 相 続（第882条～第1044条）

に分かれます（<http://www.houko.com/00/01/M29/089.HTM>より）

このうちの債権の契約の中で、世の中で使う契約の雛形について規定しています。

とくに情報処理試験で重要なのは、委任と請負です。この違いが重要です。

また雇用に関しては、民法だけでなく、労働関連の法規全般について問われます。

さらに、契約に関しては、契約関連の下請法はもちろんのこと、法律よりも、もっと具体的な

、開発契約や使用許諾権について聞かれます。

債権において不法行為自体は問われませんが、不法行為の立証上の問題から生まれた製造物責任については、聞かれますので、要注意です。

さて、この「債権」、「物件」とならんで、「財産権」に属する、無体財産権（知的財産権）は、情報処理試験では重要な分野です。

知的財産権に関しては、著作権などがよく聞かれます。

さらに関連法規としての「不正競争防止法」も要注意です。

商法に関しては、瑕疵担保責任の問題などがあります。

あと、税法とか、金融商品取引法とか会社法とか、関連する法律を挙げるときりがない・・・という状況です。シラバスにいろいろ書いています。

●標準化については、

J I S

I S O

その他標準（ANSI，IEEE等）

デファクトスタンダード

といろいろありますが、特に大事なものとしては、共通フレーム、データの標準などでしょうか？

●この分野は、やりはじめると、きりがありません。

過去問中心に、覚えていったほうが、効率的だと思います。