

基本情報技術者 科目 B 試験攻略法

1. 科目 B 試験の出題形式 ～何が問われるか～

試験時間：100分(1時間40分)

出題形式：事例問題(応用問題)で多肢選択式(複数個の選択肢から1個以上を選択)
小問形式で20問 全問必須

合格基準：満点の60% (項目応答理論によるスコアで評価)

★ アルゴリズムと情報セキュリティ の二つの分野で構成されます！

科目 B 試験の出題内容： **2023年4月から**この形式で実施されています。

問番号	テーマ	出題数
1～16	アルゴリズムとプログラミング	16問 必須
17～20	情報セキュリティ	4問 必須

- 「アルゴリズムとプログラミング」の分野は、大きく次の三つのカテゴリで構成されます。

No	カテゴリ
①	プログラムの基本要素
②	データ構造及びアルゴリズム
③	プログラミングの諸分野への適用

(出典：基本情報技術者試験 科目 B 試験サンプル問題)

また、科目 B 試験の出題範囲については、次のように定められています。

[参考 基本情報技術者試験 科目 B の出題範囲]

1 プログラミング全般に関すること

実装するプログラムの要求仕様(入出力, 処理, データ構造, アルゴリズムほか)の把握, 使用するプログラム言語の仕様に基づくプログラムの実装, 既存のプログラムの解釈及び変更, 処理の流れや変数の変化の想定, プログラムのテスト, 処理の誤りの特定(デバッグ)及び修正方法の検討 など

注記 プログラム言語について, 基本情報技術者試験では擬似言語を扱う。

2 プログラムの処理の基本要素に関すること

型，変数，配列，代入，算術演算，比較演算，論理演算，選択処理，繰り返し処理，手続・関数の呼出し など

3 データ構造及びアルゴリズムに関すること

再帰，スタック，キュー，木構造，グラフ，連結リスト，整列，文字列処理 など

4 プログラミングの諸分野への適用に関すること

数理・データサイエンス・AI などの分野を題材としたプログラム など

5 情報セキュリティの確保に関すること

情報セキュリティ要求事項の提示（物理的及び環境的セキュリティ，技術的及び運用のセキュリティ），マルウェアからの保護，バックアップ，ログ取得及び監視，情報の転送における情報セキュリティの維持，脆弱性管理，利用者アクセスの管理，運用状況の点検など

（出典：基本情報技術者試験 科目B試験サンプル問題）

科目B試験は「小問」形式が20問並ぶ形式となります。IPAが公開しているサンプル問題を見る限りでは、1問のボリュームは**1～2ページ程度**で、従来の午前問題を少し拡張したような印象を受けます。

単純計算すると、1問当たり5分(=100分÷20問)で解くスピードが求められることとなります。ただし、サンプル問題の「アルゴリズムとプログラミング」をみると、カテゴリ①～③の順に問題が並んでおり、前半はやや軽めで、問番号が大きくなるに従い、難易度が高くなる傾向にあるようです。そのため、後半の③(プログラミングの諸分野への適用)の問題などは5分では解くのが厳しい問題となりますので、全体の時間配分も重要となってくるでしょう。

また、サンプル問題では、オブジェクト指向を扱った問題も数問見受けられますので、クラスやメソッド、コンストラクタなどのオブジェクト指向の基本的な知識も必要になります。

以下に、IPAが公表したサンプル問題の一部を掲載します。

① プログラムの基本要素：問 1～問 6

問 4 次のプログラム中の ～ に入れる正しい答えの組合せを、解答群の中から選べ。

関数 gcd は、引数で与えられた二つの正の整数 num1 と num2 の最大公約数を、次の(1)～(3)の性質を利用して求める。

- (1) num1 と num2 が等しいとき、num1 と num2 の最大公約数は num1 である。
- (2) num1 が num2 より大きいとき、num1 と num2 の最大公約数は、(num1 - num2) と num2 の最大公約数と等しい。
- (3) num2 が num1 より大きいとき、num1 と num2 の最大公約数は、(num2 - num1) と num1 の最大公約数と等しい。

[プログラム]

```

○整数型: gcd(整数型: num1, 整数型: num2)
  整数型: x ← num1
  整数型: y ← num2
  
  if (  )
    x ← x - y
  else
    y ← y - x
  endif
  
  return x
    
```

解答群

	a	b	c
ア	if (x ≠ y)	x < y	endif
イ	if (x ≠ y)	x > y	endif
ウ	while (x ≠ y)	x < y	endwhile
エ	while (x ≠ y)	x > y	endwhile

for 文や while 文を用いた繰り返し処理, if 文による分岐処理などの基本制御構造を使用し、数値や配列などを処理するプログラムを扱った問題が出題されています。

(出典：基本情報技術者試験 科目 B 試験サンプル問題)

② データ構造及びアルゴリズム：問7～問13

問8 次の記述中の に入れる正しい答えを、解答群の中から選べ。

優先度付きキューを操作するプログラムである。優先度付きキューとは扱う要素に優先度を付けたキューであり、要素を取り出す際には優先度の高いものから順番に取り出される。クラス PrioQueue は優先度付きキューを表すクラスである。クラス PrioQueue の説明を図に示す。ここで、優先度は整数型の値 1, 2, 3 のいずれかであり、小さい値ほど優先度が高いものとする。

手続 prioSched を呼び出したとき、出力は の順となる。

コンストラクタ		説明
PrioQueue()		空の優先度付きキューを生成する。
メソッド	戻り値	説明
enqueue(文字列型: s, 整数型: prio)	なし	優先度付きキューに、文字列 s を要素として、優先度 prio で追加する。
dequeue()	文字列型	優先度付きキューからキュー内で最も優先度の高い要素を取り出して返す。最も優先度の高い要素が複数あるときは、そのうちの最初に追加された要素を一つ取り出して返す。
size()	整数型	優先度付きキューに格納されている要素の個数を返す。

図 クラス PrioQueue の説明

(出典：基本情報技術者試験 科目B試験サンプル問題)

【プログラム】

```
○prioSched()
  PrioQueue: prioQueue ← PrioQueue()
  prioQueue.enqueue("A", 1)
  prioQueue.enqueue("B", 2)
  prioQueue.enqueue("C", 2)
  prioQueue.enqueue("D", 3)
  prioQueue.dequeue() /* 戻り値は使用しない */
  prioQueue.dequeue() /* 戻り値は使用しない */
  prioQueue.enqueue("D", 3)
  prioQueue.enqueue("B", 2)
  prioQueue.dequeue() /* 戻り値は使用しない */
  prioQueue.dequeue() /* 戻り値は使用しない */
  prioQueue.enqueue("C", 2)
  prioQueue.enqueue("A", 1)
  while (prioQueue.size() が 0 と等しくない)
    prioQueue.dequeue() の戻り値を出力
  endwhile
```

解答群

- ア “A” , “B” , “C” , “D”
- イ “A” , “B” , “D” , “D”
- ウ “A” , “C” , “C” , “D”
- エ “A” , “C” , “D” , “D”

キュー操作のプログラムです。キューは先入先出しの基本的なデータ構造の一つですが、今まで疑似言語になかった“オブジェクト指向”の概念が用いられているのが特徴的です。問題自体は、トレースするタイプの問題ですので、表にあるメソッドの内容を確実に理解できれば、解答は可能でしょう。

(出典：基本情報技術者試験 科目B試験サンプル問題)

③ プログラミングの諸分野への適用：問 14～問 16

問14 次の記述中の に入れる正しい答えを、解答群の中から選べ。ここで、配列の要素番号は 1 から始まる。

要素数が 1 以上で、昇順に整列済みの配列を基に、配列を特徴づける五つの値を返すプログラムである。

関数 summarize を summarize({0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1})として呼び出すと、戻り値は である。

[プログラム]

```
○実数型: findRank(実数型の配列: sortedData, 実数型: p)
  整数型: i
  i ← (p × (sortedDataの要素数 - 1)) の小数点以下を切り上げた値
  return sortedData[i + 1]

○実数型の配列: summarize(実数型の配列: sortedData)
  実数型の配列: rankData ← {} /* 要素数0の配列 */
  実数型の配列: p ← {0, 0.25, 0.5, 0.75, 1}
  整数型: i
  for (i を 1 から pの要素数 まで 1 ずつ増やす)
    rankDataの末尾 に findRank(sortedData, p[i])の戻り値 を追加する
  endfor
  return rankData
```

解答群

- ア {0.1, 0.3, 0.5, 0.7, 1}
- イ {0.1, 0.3, 0.5, 0.8, 1}
- ウ {0.1, 0.3, 0.6, 0.7, 1}
- エ {0.1, 0.3, 0.6, 0.8, 1}
- オ {0.1, 0.4, 0.5, 0.7, 1}
- カ {0.1, 0.4, 0.5, 0.8, 1}
- キ {0.1, 0.4, 0.6, 0.7, 1}
- ク {0.1, 0.4, 0.6, 0.8, 1}

探索や整列などの基本アルゴリズムやデータ構造の知識はあまり必要とせず、提示されたプログラムをその場で読解して解答を求めるタイプの問題といえます。応用力が求められるため、テーマにもよりますが難易度は高めになると予想されます。

(出典：基本情報技術者試験 科目B試験サンプル問題)

④ 情報セキュリティ：問 17～問 20

問17 製造業の A 社では、EC サイト（以下、A 社の EC サイトを A サイトという）を使用し、個人向けの製品販売を行っている。A サイトは、A 社の製品やサービスが検索可能で、ログイン機能を有しており、あらかじめ A サイトに利用登録した個人（以下、会員という）の氏名やメールアドレスといった情報（以下、会員情報という）を管理している。A サイトは、B 社の PaaS で稼働しており、PaaS 上の DBMS とアプリケーションサーバを利用している。

A 社は、A サイトの開発、運用を C 社に委託している。A 社と C 社との間の委託契約では、Web アプリケーションプログラムの脆弱性対策は、C 社が実施するとしている。

最近、A 社の同業他社が運営している Web サイトで脆弱性が悪用され、個人情報が漏えいするという事件が発生した。そこで A 社は、セキュリティ診断サービスを行っている D 社に、A サイトの脆弱性診断を依頼した。脆弱性診断の結果、対策が必要なセキュリティ上の脆弱性が複数指摘された。図 1 に D 社からの指摘事項を示す。

- | | |
|------|--|
| 項番 1 | A サイトで利用しているアプリケーションサーバの OS に既知の脆弱性があり、脆弱性を悪用した攻撃を受けるおそれがある。 |
| 項番 2 | A サイトにクロスサイトスクリプティングの脆弱性があり、会員情報を不正に取得されるおそれがある。 |
| 項番 3 | A サイトで利用している DBMS に既知の脆弱性があり、脆弱性を悪用した攻撃を受けるおそれがある。 |

図 1 D 社からの指摘事項

設問 図 1 中の各項番それぞれに対処する組織の適切な組合せを、解答群の中から選べ。

解答群

	項番 1	項番 2	項番 3
ア	A 社	A 社	A 社
イ	A 社	A 社	C 社
ウ	A 社	B 社	B 社
エ	B 社	B 社	B 社
オ	B 社	B 社	C 社
カ	B 社	C 社	B 社
キ	B 社	C 社	C 社
ク	C 社	B 社	B 社
ケ	C 社	B 社	C 社
コ	C 社	C 社	B 社

与えられた事例を読解し、設問に解答する形式の問題です。この問題はマネジメント色が強いですが、暗号化などの技術系の問題も出題されるでしょう。

（出典：基本情報技術者試験 科目 B 試験サンプル問題）

2. 試験の実施方式

基本情報技術者試験は、2023年4月から通年試験(いつでも受験できる試験)に変更され、C B T (Computer Based Testing) 方式で実施されています。

● CBT方式とは

コンピュータを利用して実施する試験方式のことで、受験者はコンピュータに表示された試験問題に対して、マウスやキーボードを用いて解答します。

紙の問題冊子は配布されません。

また、試験問題は非公開であり、「第三者に漏えいしないこと」について、受験申込み時に同意する必要があります。

① 受験会場

全都道府県において、それぞれ1か所以上の試験会場が用意され、会場ごとに、「複数の試験実施日時」が設定されます。

② 受験申込み

申込み時に、試験会場と試験日、開始時刻を自分で選択します。ただし、当該会場の当該日、当該時刻において、空席がなければ予約はできません。また、**科目 A 試験**と**科目 B 試験**は、同じ日に受験することになります。

科目 A 試験(90分) → 休憩(10分) → 科目 B 試験(100分)

③ 成績及び合格発表

各試験終了後に、画面上に科目評価点がそれぞれ表示されます。合格発表は、受験日の翌月中旬に情報処理推進機構ホームページから確認できます。また、受験者マイページでも確認することができます。

④ リテイクポリシー

項目	説明
一度受験申込をした試験区分の再申込が可能になる日時	申込済の試験の終了時刻を過ぎたら、再申込が可能になります。 注記 システムでの処理の都合上、申込済の試験の終了時刻を過ぎてから再申込が可能になるまでには数時間～1日程度掛かります。
一度受験した試験区分の再申込時に、受験日として指定が可能な日	前回の受験日の翌日から起算して30日を超えた日以降を、受験日として指定可能です。 注記 前回の試験を受験せずに欠席した場合、このリテイクポリシーは適用されません。

(情報処理推進機構(IPA)HP <https://www.ipa.go.jp> より)

3. 科目 B 試験の対策

① 情報セキュリティの攻略

過去本試験の午後試験で扱われたテーマを幾つか挙げてみると、技術系の「暗号方式の特徴」「ハッシュ関数」「デジタル署名」「ファイアウォール」、マネジメント系の「リスクアセスメント」「各種攻撃と対策」「ログ管理」「パスワードの数」「CIA」などがあります。

科目 B 試験のサンプル問題では、マネジメント系の問題がやや多めでしたので、対策においてもセキュリティ技術だけではなく、情報セキュリティマネジメント的な内容もしっかり抑えておく必要があります。

まずは、技術系の定番ともいえる**暗号化と認証**を中心に対策していきましょう。TLSやPKI、デジタル証明書を含めた認証局(CA)、VPNなどの知識も確実に身に付けておくべきです。このとき、単なる用語の暗記ではなく、**その目的、メリット、デメリットなどの特徴**も確実に抑えておきましょう。「暗号方式の特徴」「ハッシュ関数の特徴」などが論点として狙われる可能性があります。また、ネットワークセキュリティの出題に備えて、ファイアウォールのパケットフィルタリングの設定方法は必須とし、IP アドレスやサブネットといったネットワークの基礎知識も確認しておきましょう。

次に、SQL インジェクションなどの攻撃や犯罪及びその対策などの基本知識も重要です。また、演習問題を多く解くことで、事例を読解し現状システムのセキュリティ要件や問題点を洗い出し、インシデントの発生に対してどのような対策が必要か、原因はどこにあったのか、といった設問に答えられる力を養いましょう。そのためには、問題文や図表から「ヒントとなる部分」を確実に見極めることを常に心がけて演習するようにしてください。

また、CBT 試験では問題文に書込みができません。本試験では、メモ用紙とボールペンが用意されますので、

- ・重要な用語や文章を、メモ用紙に書き出して記録しておく
- ・現状のセキュリティルールやインシデント対応などで気付いた点・注意点などは、とにかくメモ用紙に書いて記録しておく

といったことを必ず練習しておきましょう。



② アルゴリズムの攻略

1問1問のボリュームが今までの午後試験(6~8ページ)より少なくなり、1~2ページの小問になったからといって、アルゴリズムの学習量が大きく減るわけではありません。

カテゴリ②「データ構造及びアルゴリズム」や③「プログラミングの諸分野への適用」の問題を解くためには、基本アルゴリズム及びデータ構造に関する基礎知識を含めた“アルゴリズムのスキル”を上げておく必要があります。このスキルが不足していると、プログラムの穴埋めなどに対応するのが難しくなります。

しかし、この“アルゴリズムのスキル”は簡単に上がりませんので、時間をかけて学習及び演習をしていく必要があります。

ただし、昨年までの基本情報技術者試験の受験者が行ってきたデータベースやネットワークなどの選択問題対策及びプログラム言語の学習が、科目B試験対策では必要無くなりました。その分しっかりとアルゴリズムの学習に集中することができます。



基本アルゴリズムの理解・習得!!

まず、テキストを用いて「基本アルゴリズム」の学習をしっかりと行うことが重要です。「基本アルゴリズム」には、

最大値(最小値)アルゴリズム, 線形探索, 2分探索

基本整列アルゴリズム(選択法, 交換法, 挿入法)

高速整列アルゴリズム(クイックソート, マージソートなど)

文字列照合, 文字列置換

などがあります。これらのアルゴリズムがカテゴリ②の「データ構造及びアルゴリズム」のテーマとして出題された場合、その基礎知識があれば、プログラムの理解がより速くなり、正答率も上げることができます。

また、これらの学習を通して、

配列処理(添字の初期化, ループの条件, 添字の更新など)

添字の特別な表し方, 図を描くことで整理・理解する

など、他のカテゴリ①や③の問題を解く基礎となる力を身につけることを意識しましょう。



データ構造の攻略!!

カテゴリ②の「データ構造及びアルゴリズム」で扱われるデータ構造には、

スタック, キュー, リスト, 木

の4つがあります。各データ構造の特徴として、その考え方は決して難しいものではないが、いざプログラム中で実装しようとする時、急に難易度が上がってしまいます。

このテーマを苦手とする方も多いのですが、逆に一度攻略しておけば、他の受験者より有利になるといえるでしょう。

そこで、以下の内容をしっかりと学習しておきましょう。

スタック	：配列での PUSH, POP の操作。特にスタックポインタの使い方
キュー	：配列のエンキュー, デキューの操作, 配列の循環利用
リスト	：リストを先頭からたどるループ。データの挿入・削除の処理手順
木	：2分探索木やヒープの考え方及びその実装



トレースの練習!!

科目 B 試験では、**トレースの問題も出題されます**。そこで、学習時及び演習時にも積極的にトレースを行ってみましょう。問題で提示された例や、自分で作ったデータ(配列など)を使って、プログラムの流れを追ってみるのです。

また、トレースにより、次のようなことを理解することができます。

初期値, ループの条件(どういった場合に終了するのか), 添字の更新, ループ後の結果となる変数や配列の状態, 具体的な処理手順



演習!! 演習!! そして問題に慣れること!!

スキルを上げるために必要なのは、なるべく多くのアルゴリズム(プログラム)を読むことです。問題集などを用いて、**多くの演習に取り組み, 問題に慣れることも重要です**。このとき、科目 A 試験に出てくるアルゴリズムの問題も効果的な演習材料になりますので、うまく活用していきましょう。

また、CBT 試験では問題文に書込みができません。本試験では、メモ用紙とボールペンが用意されますので、演習時にメモ用紙やノートを用意して、

- ・図の一部を抜き出したり、自分で図解してみたりすることで整理する
- ・プログラムの一部や大まかな流れをメモ用紙に書いてみる
- ・配列などの図を描いたり、トレースをしたり、手を動かしながら解く

という練習をしていきましょう。

演習を積み重ねることで、いくつかの解法パターン(このタイプの問題ならばこう解けばよい、といったパターン)が身につけられますので、その点も意識してみましょう。

なお、TAC の科目 B マスターコースでは、アルゴリズムについて、

講義(インプット)2回 → 演習1回 → 講義2回 → 演習1回 → …

というように、講義と演習を交互に行うスパイラル学習で、演習を確実に実施して頂くコース設計になっております。



4. CBT 試験の対策

・PCを使った演習を試みる

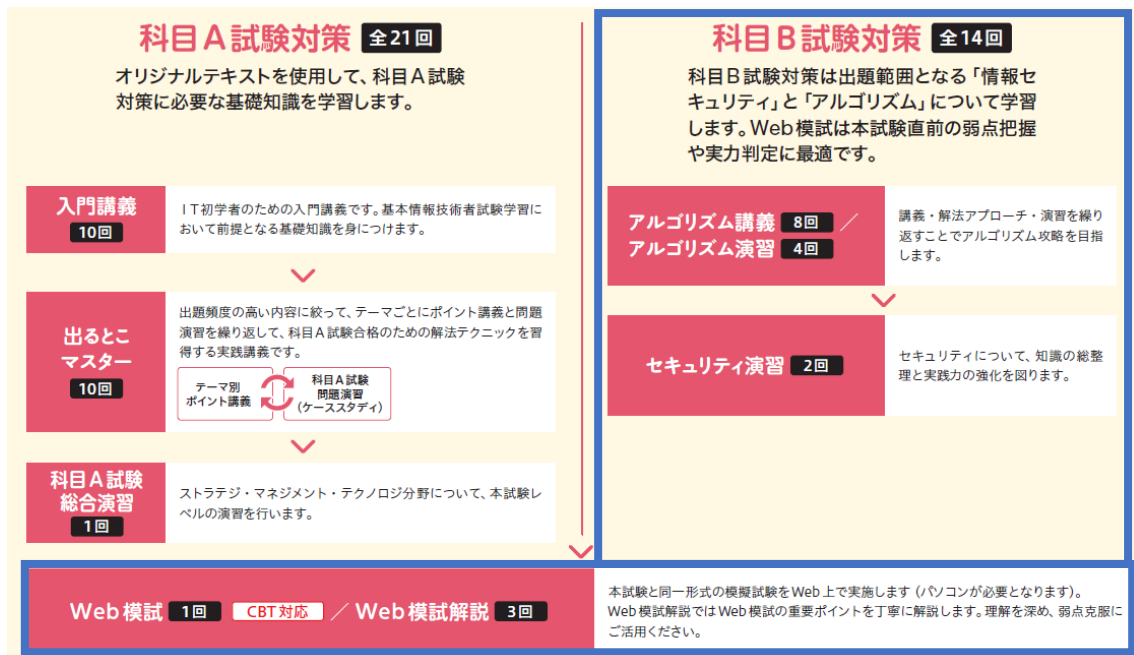
本試験の受験が近づいてきたら、IPA が公表したサンプル問題の PDF を用いて、PC を用いた演習を試みるとよいでしょう。画面に問題の PDF を表示させ、メモ用紙を用意し、画面を見ながら解く練習をします。

5. TAC 講座へのご案内

基本情報技術者は学習範囲が広く、覚えるべき項目(専門用語や公式など)が多い試験です。また、科目 B 試験対策では、学習に時間のかかる「アルゴリズム」を習得する必要があります。そのため短期間に合格を目指すなら、適切なロードマップに従って、しっかりとしたペースで歩みを進める必要があります。

TACで実施している「基本情報技術者講座」は、オリジナル教材を使用して、講師の迫力ある講義、テスト、質問フォローなど合格に必要なすべてがパッケージされていますので、短期間に無駄なく効果的に学ぶことができます。

(TAC 基本情報技術者 科目 B マスターコースのカリキュラム)



●合格のためのアドバイス

- ・毎回の講義をなるべく欠席しないこと。欠席した場合はフォロー制度を利用して、早めに挽回すること。
- ・復習を中心とした自己学習をテキスト・問題集でしっかり行うこと。
- ・疑問点は講師に積極的に質問しましょう。教室なら直接講師に、Web 通信講座などならば質問メールで。遠慮は禁物ですぞ。