

情報処理技術者試験

出題範囲

■試験の対象者像.....	1
■試験の出題形式と試験時間.....	11
■出題範囲.....	12
〔午前の試験〕.....	12
(試験区分別出題範囲一覧表).....	12
コンピュータ科学基礎.....	13
コンピュータシステム.....	13
システムの開発と運用.....	14
ネットワーク技術.....	15
データベース技術.....	15
セキュリティと標準化.....	15
情報化と経営.....	16
監査.....	16
〔午後の試験〕.....	17
システムアナリスト試験.....	17
プロジェクトマネージャ試験.....	17
アプリケーションエンジニア試験.....	17
ソフトウェア開発技術者試験.....	18
テクニカルエンジニア(ネットワーク)試験.....	19
テクニカルエンジニア(データベース)試験.....	19
テクニカルエンジニア(システム管理)試験.....	20
テクニカルエンジニア(エンベデッドシステム)試験.....	20
テクニカルエンジニア(情報セキュリティ)試験.....	20
情報セキュリティアドミニストレータ試験.....	21
上級システムアドミニストレータ試験.....	21
初級システムアドミニストレータ試験.....	22
システム監査技術者試験.....	22
基本情報技術者試験.....	23
■試験で使用する情報技術に関する用語・プログラム言語など.....	24
■アセンブラ言語の仕様.....	26
■表計算ソフトの機能・用語.....	34
■プログラム言語 Perl の用例・解説.....	37

■試験の対象者像

試験の対象者像、役割と業務、期待する技術水準を次に示します。

(1) システムアナリスト試験：Systems Analyst Examination

対象者像	経営戦略に基づく情報戦略の立案，システム化全体計画及び個別システム化計画の策定を行うとともに，計画立案者の立場から情報システム開発プロジェクトを支援し，その結果を評価する者
役割と業務	<p>情報システムの企画・計画の段階で，情報戦略立案，システム化全体計画策定，個別システム化計画策定を行い，また開発・導入の段階で，戦略と計画に基づく情報システム構築の実施に関する推進支援や，システム化と同時に進める業務革新の推進支援を行うとともに，それらの結果に関する評価を行う業務に従事し，次の役割を果たす。</p> <p>①経営戦略と一貫性のある情報戦略を立案し，それに基づくシステム化全体計画を策定する。</p> <p>②情報システムに関する資源及び組織の運営方針を策定し，システム化全体計画に反映させる。</p> <p>③システム化全体計画に基づいて，業務革新に貢献するシステム化案を提案し，個別システム化計画を策定する。</p> <p>④戦略と計画に基づく情報システム構築と情報システムサービス運営の実施を支援し，併せてシステム利用部門が主体で実施する業務革新を支援する。</p> <p>⑤計画立案者の立場から，システム化の結果に関して，合目的性・有効性・効率性・達成品質水準を評価する。</p>
期待する技術水準	<p>情報戦略が経営戦略の一環として重要な位置を占めつつあるなか，システムアナリストは，情報戦略立案，情報技術を活かした業務革新提案，及びシステム化計画策定の担い手として，次の幅広い知識・経験・実践能力が要求される。</p> <p>①経営戦略を正しく理解し，その中から経営上の課題に対して情報システムによる支援要素を抽出し，具体的な情報システムの構築構想と情報戦略を立案できる。</p> <p>②業務モデル・情報システム全体体系を定義する技術をもち，必要な体制を整え，情報システムの開発課題を分析したうえで，システム化全体計画を策定できる。</p> <p>③情報技術動向やシステム製品動向を把握し，最適な情報システム基盤構成の方針を策定できる。</p> <p>④現状分析，システム化要件の整理，システム概要設計，効果・コスト・リスク評価などを行い，業務革新に貢献する個別システムの開発計画を策定できる。</p> <p>⑤システム化全体計画，個別システム開発計画に対する評価ポイントを作成できる。また，システム化の結果を，有効性や達成品質水準などの観点から評価できる。</p>

(2) プロジェクトマネージャ試験：Project Manager Examination

対象者像	情報システム開発プロジェクトの責任者として，プロジェクト計画の作成，要員などプロジェクト遂行に必要な資源の調達，プロジェクト体制の確立及び予算・納期・品質などの管理を行い，プロジェクトを円滑に運営する者
役割と業務	<p>情報システム開発プロジェクトの責任者として，当該プロジェクトを計画，推進・管理する業務に従事し，次の役割を果たす。</p> <p>①個別システム化計画に基づいて，当該プロジェクトの実行計画をプロジェクト計画</p>

	<p>として作成するとともに、必要な資源を調達し、プロジェクト体制を確立する。</p> <p>②予算、納期、品質などを管理し、プロジェクトを円滑に運営する。進捗状況を把握し、問題や将来見込まれる課題を早期に把握・認識し、適切な対策・対応を実施することによって、プロジェクトの目的を達成する。</p> <p>③プロジェクトの上位責任者及び関係者に、適宜、プロジェクトの取組方針、進捗状況、課題と対応策などを報告し、支援・協力を得て、プロジェクトを円滑に推進する。</p> <p>④プロジェクトの大工程及び全体の終了時、又は必要に応じて適宜、プロジェクトの計画と実績を分析・評価し、プロジェクトのその後の運営に反映するとともに、ほかのプロジェクトの参考に資する。</p>
期待する技術水準	<p>プロジェクトマネージャの業務を円滑に遂行するため、組織経営及び情報システム全般に関する基本的な事項を理解し、次の幅広い知識・経験・実践能力が要求される。</p> <p>①システム化計画及び期待されているプロジェクトを正しく認識し、実行可能なプロジェクト計画を立案できる。</p> <p>②与件・制約条件の中で、プロジェクトの全体意識を統一し、プロジェクトの目標を確実に達成できる。</p> <p>③人・資材・予算・納期・品質を管理し、プロジェクトを推進できる。</p> <p>④プロジェクトの進捗状況や将来見込まれる課題を早期に把握し、適切に対応できる。</p> <p>⑤プロジェクトの計画・実績を適切に分析・評価できる。また、その結果をプロジェクトの運営に反映するとともに、ほかのプロジェクトの参考に資することができる。</p>

(3) アプリケーションエンジニア試験：Application Systems Engineer Examination

対象者像	<p>情報システム開発プロジェクトにおいて、プロジェクト計画に基づいて、業務要件分析からシステム設計、プログラム開発、テストまでの一連のプロセスを担当する者</p>
役割と業務	<p>情報システム開発プロジェクトにおいて、業務要件を分析し、システムとして実現する業務に従事し、次の役割を果たす。</p> <p>①利用者側の業務要件を分析し、要求仕様をまとめる。</p> <p>②要求仕様に基づいて、新規開発・パッケージ導入などのシステム実現方法、システム構成、システム移行・運用などについて検討し、システム設計を行う。</p> <p>③プログラム開発要員を指導して、プログラム開発を実施させる。</p> <p>④総合テストを計画し、実施する。また、システム移行及び運用テストを支援する。</p> <p>⑤個別のハードウェア技術、ソフトウェア技術、ネットワーク構成、データベース構成、システム運用などについては、必要に応じて専門家の支援を受ける。</p>
期待する技術水準	<p>情報システム開発の中核技術者として業務要件を分析し、システムとして実現するため、次の幅広い知識・経験・実践能力が要求される。</p> <p>①経理、生産管理などの業務知識をもち、利用者側の業務要件を分析し、要求仕様書を作成できる。</p> <p>②新規開発、パッケージ導入などのシステム実現方法の評価・選択ができる。</p> <p>③ハードウェア、ソフトウェア、ネットワーク、データベースなど情報技術に関する全般的な知識を有し、必要に応じて特定分野の情報技術の専門家から支援を受け、</p>

	<p>要求仕様に合った情報システムが設計できる。</p> <p>④要求仕様書に基づいて、標準的な記法によって、外部設計書・内部設計書を作成できる。</p> <p>⑤プログラム開発、単体・結合テストに際し、プログラム開発要員を指導できる。</p> <p>⑥総合テストの計画と管理が行え、また、システム移行及び運用テストで利用者側の要員を支援できる。</p> <p>⑦計画された品質・工程の実行管理ができる。</p>
--	--

(4) ソフトウェア開発技術者試験：Software Design & Development Engineer Examination

対象者像	<p>情報システム開発プロジェクトにおいて、内部設計書・プログラム設計書を作成し、効果的なプログラムの開発を行い、単体テスト・結合テストまでの一連のプロセスを担当する者</p>
役割と業務	<p>情報システム開発プロジェクトにおいて、外部仕様に基づいてソフトウェアを開発する業務に従事し、次の役割を果たす。</p> <p>①外部設計書の作成者との十分な意思疎通を図り、それらに基づく内部設計書・プログラム設計書を作成する。</p> <p>②高度なアルゴリズムやデータ構造に関する知識に基づいて、効果的なプログラムを作成する。</p> <p>③プログラムの単体テスト・結合テストを確実に実施する。</p> <p>④ソフトウェア開発に関して、基本情報技術者を指導する。</p>
期待する技術水準	<p>情報システム開発におけるソフトウェア開発技術者として、外部仕様に基づいて内部設計・プログラム設計・プログラム開発を行い、高品質なソフトウェアを開発するため、次の知識・技能が要求される。</p> <p>①ネットワーク、データベース、システム構成などの情報技術に関する全般的な知識をもち、上位技術者の指導のもとに情報システムの設計ができる。</p> <p>②内部設計書・プログラム設計書を作成できる。</p> <p>③プログラミングに必要な高度の論理的思考をもつ。</p> <p>④ネットワーク、データベースなどに関する実装技術をもつ。</p> <p>⑤一つ以上のプログラム言語の仕様を熟知しており、その言語の特徴を利用して効果的なプログラムの開発ができるとともに、基本情報技術者を指導できる。</p> <p>⑥プログラムのテスト手法を熟知しており、単体テスト・結合テストの計画と管理が行え、テストの実施についてはプログラム開発要員を指導できる。</p>

(5) テクニカルエンジニア（ネットワーク）試験：Network Systems Engineer Examination

対象者像	<p>情報システム基盤（業務システム共有のシステム資源）の構築・運用において中心的役割を果たすとともに、個別の情報システム開発プロジェクトにおいて、固有技術の専門家として開発・導入を支援する者。具体的には、ネットワークを固有技術とする。</p>
役割と業務	<p>ネットワークシステムを計画・設計・構築・運用する業務に従事し、次の役割を果たす。</p> <p>①ネットワーク管理者として、情報システム基盤であるネットワーク資源を管理する。</p>

	<p>②WAN/LAN に対する要求を分析し、効率性・信頼性・安全性を考慮した設計・構築・運用を行う。</p> <p>③個別システム開発の各工程（計画・分析・設計・運用・保守）において、ネットワーク関連の技術支援を行う。</p>
期待する技術水準	<p>ネットワークは情報システム基盤であり、個別のアプリケーションシステムの構成要素ともなる。また、マルチメディアへの展開も含め、ネットワークに要求される機能要件は急速に進展している。目的に適合したネットワークを構築・維持するため、次の幅広い知識・経験・実践能力が要求される。</p> <p>①ネットワーク技術・制度の動向を広く見通し、目的に応じて適用可能な技術を選択できる。</p> <p>②企業・組織全体又は個別アプリケーションのネットワークへの要求を的確に理解し、ネットワーク要求仕様を作成できる。</p> <p>③要求仕様に関連するネットワーク設計技法、プロトコル技術、信頼性設計、セキュリティ技術、通信サービス・料金などを選択して、最適な論理設計・物理設計ができる。</p> <p>④ネットワーク関連企業（通信事業者、ベンダ、工事業者など）を活用して、ネットワークの効率的な構築・運用ができる。</p>

(6) テクニカルエンジニア（データベース）試験：Database Systems Engineer Examination

対象者像	<p>情報システム基盤（業務システム共有のシステム資源）の構築・運用において中心的役割を果たすとともに、個別の情報システム開発プロジェクトにおいて、固有技術の専門家として開発・導入を支援する者。具体的には、データベースを固有技術とする。</p>
役割と業務	<p>情報資源及びデータベースを計画・設計・構築・運用・管理する業務に従事し、次の役割を果たす。</p> <p>①データ管理者として、情報システム全体のデータ資源を管理する。</p> <p>②データベース管理者として、基幹データベースの構築と維持を行う。</p> <p>③個別システム開発の各工程（計画・分析・設計・運用・保守）において、データベース関連の技術支援を行う。</p>
期待する技術水準	<p>基幹業務の効率化と情報活用の高度化の両面において、データベースは重要な役割をもつ。高品質なデータベースを構築・維持するため、次の幅広い知識・経験・実践能力が要求される。</p> <p>①情報資源管理の目的と技法を理解し、データ部品の標準化、リポジトリシステムの設計・構築・運用・保守ができる。</p> <p>②データモデル化技法を理解し、利用者の要求に基づいてデータ分析を行い、正確な概念データモデルを作成できる。</p> <p>③データベース管理システムの特性を理解し、高品質な基幹データベース及びデータウェアハウスの設計・構築・運用・保守ができる。</p>

(7) テクニカルエンジニア（システム管理）試験：Systems Management Engineer Examination

対象者像	<p>情報システム基盤（業務システム共有のシステム資源）の構築・運用において中心的役割を果たすとともに、個別の情報システム開発プロジェクトにおいて、固有技術の</p>
------	---

	<p>専門家として開発・導入を支援する者。具体的には、システム管理を固有技術とする。</p>
役割と業務	<p>情報システム基盤（業務システム共有のシステム資源）を企画・構築・運用する業務に従事し、次の役割を果たす。</p> <p>①システム管理者として、情報システム基盤に関する構成管理・障害管理・性能管理・課金管理・セキュリティ管理を行う。</p> <p>②情報システムの安定的・効率的な運用のための改善活動を行う。</p> <p>③新規システム受入れに伴う運用テストとシステム移行を計画・実施する。</p> <p>④システム利用者に対して技術的な助言・援助・支援を行う。</p>
期待する技術水準	<p>複雑化・多様化している情報システムのシステム管理業務に関し、他の専門家と適切に協力しながら、自ら管理技術を適用して総合的にシステム管理を行うため、次の幅広い知識・経験・実践能力が要求される。</p> <p>①情報システム基盤を安全かつ安定的に運用し管理するための知識をもち、ハードウェア、ソフトウェア、ネットワーク、施設・設備などの構成を管理し、システム資源の維持と拡張に対応できる。</p> <p>②稼働システムのサービスレベル・業務処理量・資源・コストを勘案した性能管理を行うとともに、性能評価及び運用評価ができる。</p> <p>③障害の監視・究明・回復・防止に必要な技術をもち、システムの障害管理ができる。</p> <p>④実効性の高いセキュリティ対策を実施するための知識・技能をもち、セキュリティ管理ができる。</p> <p>⑤システムの利用状況を把握し、適切な課金管理ができる。</p> <p>⑥稼働中及び稼働予定のシステム運用品質の管理水準を設定し、必要な改善ができる。</p> <p>⑦新規システム受入れに伴う運用テストとシステム移行を計画し、実施できる。</p> <p>⑧システム利用者に対して、システム利用及び運用に関する教育計画を立案し、実施できる。</p>

(8) テクニカルエンジニア（エンベデッドシステム）試験：

Embedded Systems Development Engineer Examination

対象者像	<p>情報システム基盤（業務システム共有のシステム資源）の構築・運用において中心的役割を果たすとともに、個別の情報システム開発プロジェクトにおいて、固有技術の専門家として開発・導入を支援する者。具体的には、マイクロプロセッサやシステム LSI などを組み込んだエンベデッドシステムについて、情報システムを構成する専門性をもった技術要素として固有技術に含むものとする。</p>
役割と業務	<p>エンベデッドシステムに関する要求仕様のまとめ、システム開発工程の計画及びシステム開発工程に基づく設計作業を遂行する業務に従事し、次の役割を果たす。</p> <p>①要求された仕様を確保するために、製品に組み込むシステムに対するソフトウェア、マイクロプロセッサ、システム LSI などの技術的効果・経済的効果に関する評価を行う。</p> <p>②エンベデッドシステムを対象として、システム開発工程の立案、要求仕様書の作成、リアルタイム性の検討、及びハードウェアとソフトウェアのトレードオフを含む</p>

	<p>めてシステムの設計を行う。</p> <p>③マイクロプロセッサ，システム LSI，リアルタイム OS などの仕様の作成を行う。</p> <p>④開発を遂行するうえでの開発環境の整備を行う。</p>
期待する技術水準	<p>要求された性能，品質，信頼性などをソフトウェアとハードウェアの適切な組合せによってエンベデッドシステムとして実現するため，次の幅広い知識・経験・実践能力が要求される。</p> <p>①マイクロプロセッサやシステム LSI などを組み込む立場から，ソフトウェア開発の要求仕様をまとめることができる。</p> <p>②要求仕様書を分析し，ハードウェアとソフトウェアの適切な組合せによって，エンベデッドシステム開発工程の計画及び管理ができる。</p> <p>③機能設計，詳細設計，システムの実現，ハードウェアとソフトウェアを結合した総合テストなどの一連の作業を実施できる。</p> <p>④情報技術及び制御技術などに関する基礎から先端までの知識をもち，要求されたエンベデッドシステムを開発する当該分野の知識を速やかに獲得でき，当該分野の専門家と技術上の打合せができる。</p>

(9) テクニカルエンジニア（情報セキュリティ）試験：Information Security Engineer Examination

対象者像	<p>情報システム基盤（業務システム共有のシステム資源）の構築・運用において中心的役割を果たすとともに，個別の情報システム開発プロジェクトにおいて，固有技術の専門家として開発・導入を支援する者。具体的には，情報セキュリティを固有技術とする。</p>
役割と業務	<p>セキュリティ機能が求められる情報システムの計画・設計・構築・運用において，セキュリティ機能の計画・設計・構築を推進又は支援する業務，若しくはセキュアな開発プロジェクト環境を整備する業務に従事し，次の役割を果たす。</p> <p>①情報システムの脆弱性・脅威を分析・評価し，これらを適切に回避・防止するセキュリティ機能の計画・設計・構築を推進又は支援する。</p> <p>②情報システム又はセキュリティ機能の構築までのプロジェクトにおいて，情報システムへの脅威を分析し，プロジェクト管理を適切に支援する。</p> <p>③セキュリティ侵害への対処やセキュリティパッチの適用作業など情報システム運用時のセキュリティ管理作業を技術的な側面から支援する。</p>
期待する技術水準	<p>情報システムでは，多くのコンポーネントにおいてセキュリティ機能が求められる。情報セキュリティ技術の専門家として，他の専門家と協力しながら情報セキュリティ技術を適用して，セキュアな情報システムを開発・運用するため，次の幅広い知識・経験・実践能力が要求される。</p> <p>①情報セキュリティ対策のうち，技術的な対策について基本的な技術と複数の特定の領域における応用技術をもち，これらの技術を対象システムに適用するとともに，その効果を評価できる。</p> <p>②情報セキュリティ対策のうち，運用的な対策について基本的な知識と適用場面に関する技術をもつとともに，情報セキュリティマネジメントの基本的な考え方を理解し，これを適用するケースについて具体的な知識をもち，評価できる。</p> <p>③情報技術のうち，ネットワーク，データベース，システム開発環境について基本的な知識をもち，情報システムの機密性，責任追跡性などを確保するために必要な暗</p>

	号, フィルタリング, ログインなどの要素技術を理解している。 ④情報システム開発における工程管理について基本的な知識と具体的な適用事例の知識をもつ。
--	--

(10) 情報セキュリティアドミニストレータ試験 :

Information Systems Security Administrator Examination

対象者像	情報セキュリティに関する基本的な知識をもち, 情報セキュリティ管理の現場責任者として, 情報セキュリティを保つための施策を計画・実施し, その結果の評価を行う者
役割と業務	情報セキュリティ管理の現場責任者として, 情報セキュリティに関する企画・実施・運用・分析のすべての段階で, 物理的観点, 人的観点及び技術的観点から情報セキュリティを保つための施策を計画・実施し, その結果に関する評価を行う業務に従事し, 次の役割を果たす。 ①情報資源の洗い出し, 脅威分析, リスク分析を行い, 組織体におけるセキュリティ管理のターゲットを示す情報セキュリティポリシーを策定する。 ②情報セキュリティポリシーに則って, それを実現するための技術の選択と適用, 運用に関するガイドラインの策定, 一般利用者教育を行う。 ③管理対象から出力される各種情報に従って, セキュリティ侵犯がないか常に監視し, 侵犯発生時には対策を講じる。また, 情報セキュリティポリシー策定時のレベルを維持できるよう適切な措置を講じる。 ④セキュリティ侵犯事象の根本原因を解明し, 改善策を策定する。
期待する技術水準	情報セキュリティ確保は, 各組織における基本的な責任であるとともに, 社会的要請でもある。各組織において情報セキュリティを確保し, 必要な管理を遂行するために, 次の知識・技能が要求される。 ①情報セキュリティポリシーを策定できる。 ②リスク分析, リスク管理ができる。 ③情報セキュリティポリシーに基づき, 具体的な実施手順を作成できる。 ④OS, ネットワーク, インターネットに関する技術, 製品 (ハードウェア, ソフトウェア, サービス) 知識をもち, 利用・活用できる。 ⑤防御技術に関する知識をもち, 適用できる。 ⑥セキュリティ運用・管理に関する知識をもち, 策定できる。 ⑦脆弱性に関する知識をもち, 対処方法を検討・実施できる。 ⑧セキュリティ侵犯を発見し, 対処できる。 ⑨情報セキュリティ, プライバシー関連法規の知識をもっている。 ⑩情報セキュリティの監査, 評価に関する知識をもち, 対象を監査・評価できる。 ⑪一般利用者における情報セキュリティ対策を策定できる。 ⑫情報セキュリティに関する国内・国際標準の知識をもち, 適用できる。

(11) 上級システムアドミニストレータ試験 : Senior Systems Administrator Examination

対象者像	利用者側において, 業務の中でどのように情報技術を活用すべきかについて判断するために必要な知識・技能をもち, 情報化リーダーとして業務改革・改善を推進する者
役割と	企業・組織のビジネス活動, 業務活動の中心的役割を果たす立場にいて, 業務改善と

業務	<p>情報化に関する PDCA サイクルを主導する。すなわち、経営戦略及びビジネス戦略を理解したうえで、業務システム及び情報システムの面で、次のような役割を果たす。</p> <p>[業務システム]</p> <ul style="list-style-type: none"> ①業務モデル、業務プロセス変革の企画に参画するとともに実施計画を立案し、実施環境を整える。 ②新しい業務モデル、業務プロセスの具体的な実施内容を作成し、改革の活動を指導する。 ③新しい活動の効果を評価し、更なる改善にフィードバックさせる。 <p>[情報システム]</p> <ul style="list-style-type: none"> ①業務モデル変革に適合する情報システム化の RFP 作成、IT ベンダへの提示・評価作業に参画し、経営層を支援する。 ②システム開発の進捗状況と完成度を把握し、受入れ、運用準備を行うとともに、稼働開始に必要な各種事項を決定、指示する。 ③日常業務の一環として、システムの運用状況・利用状況を継続的に把握し、システム化の達成度を評価して改善要求につなげる。 ④経営層の行うビジネス戦略、情報戦略の立案・評価に参画し、支援する。
期待する技術水準	<p>情報化戦略が経営戦略を実現させる大きな要素となっているなか、上級システムアドミニストレータは、業務遂行側の立場で、情報技術を生かした業務革新の提案、必要となる情報システムの実現、新システムの活用・評価の担い手として、次の幅広い知識・経験・実践能力が要求される。</p> <ul style="list-style-type: none"> ①経営、マネジメント、情報技術に関する全般的な知識をもち、ビジネスの動向、情報技術の動向を正しく捉えられる。 ②電子商取引などの情報技術を活用した最新経営技術動向、情報技術動向を理解し、業務モデル変革の企画立案に参画できる。 ③業務モデルの策定、理解に当たって、幅広い視点、視野をもち、全体としての最適ソリューションを考えられる。 ④情報システムの提供者側との検討の場において、ビジネス、業務、システムを抽象化し、適切な機能モデルを作成できる。 ⑤自企業・組織の現状や業務を、他社状況、市場状況などを踏まえて正しく分析し、問題点を明確にできる。 ⑥問題点を整理し、優先順位を考慮した解決策を策定できる。 ⑦解決策に関する費用対効果分析、リスク分析を行い、その結果を経営層に説明できる。 ⑧解決策に基づく業務、組織、システムの設計を行い、実現させる。 ⑨解決策に対する効果目標及び評価基準を設定し、達成度を評価できる。 ⑩計画に対する管理、必要に応じての適切な対応策をとることができる。 ⑪システムの活用においては、イントラネットを活用した情報発信の促進、情報の質的向上を図るなど、情報技術を活用した業務の改革・改善を考えられる。

(12) 初級システムアドミニストレータ試験：Systems Administrator Examination

対象者像	利用者側において、情報技術に関する一定の知識・技能をもち、部門内又はグループ内の情報化を利用者の立場から推進する者
------	---

役割と業務	<p>利用者側において情報技術に関する一定の知識・技能をもつ者であり、担当する業務の情報化を利用者の立場から推進するために、次の役割を果たす。</p> <p>①現状業務における問題点を把握し、情報技術を活用してその解決を図る。</p> <p>②必要とする情報システムの一部構築とその支援を行う。</p> <p>③情報システムの提供者側に対する利用者の意見や要望を提起する。</p> <p>④情報システムの運用環境とシステム利用環境を整備する。</p>
期待する技術水準	<p>利用者側において、担当する業務の情報化を利用者の立場から推進するため、次の知識・技能が要求される。</p> <p>①仕事の進め方を把握し改善策を考えるためのシステム思考能力、それを支えるDFD、ワークフローなどの手法やコンピュータの活用法に関する知識をもつ。</p> <p>②情報システムの開発・利用について、ヒューマンインタフェース設計、テスト及びシステム運用に関する知識・技能をもつ。</p> <p>③パソコンやネットワークに関する基礎知識をもつ。</p> <p>④業務において表計算ソフトやデータベースソフトなどのツールを操作・活用できる。</p> <p>⑤パソコン導入・運用・管理における実務的な知識・技能をもつ。</p> <p>⑥パソコンの様々な使い方やパソコン利用環境・オフィス環境に関する知識をもつ。</p> <p>⑦情報化推進のための話し方・文書の書き方・ビジュアル表現方法に関する知識をもつ。</p>

(13) システム監査技術者試験：Systems Auditor Examination

対象者像	<p>被監査部門から独立した立場で、トップマネジメントの視点で、情報システムが経営に貢献しているかどうかを、安全性、効率性、信頼性、可用性、機密性、保全性、有用性、戦略性など幅広い側面から総合的に調査し、あるべき姿を描くことによって自ら形成した判断基準に照らして評価し、問題点について説得力のある改善勧告を行う者</p>
役割と業務	<p>被監査対象から独立した立場で、情報システムを総合的に点検・評価し、監査結果をトップマネジメント及び関係者に説明し、改善点を勧告する業務に従事し、次の役割を果たす。</p> <p>①監査計画を立案し、監査を実施し、監査結果をトップマネジメント及び関係者に報告する。</p> <p>②情報システムに関する内部統制機能の改善を促進し、その実効性を担保することによって、企業経営はもとより、情報社会・ネットワーク社会の健全化に貢献する。</p>
期待する技術水準	<p>単に情報処理の視点からだけでなく、情報システムが企業及び社会に貢献できるように改善を促進するため、次の幅広い知識・経験・実践能力が要求される。</p> <p>①ビジネス要件や経営方針に合致した監査計画を立案できる。</p> <p>②情報システムの企画・開発・運用段階において、効率的な監査手続を実施するための監査技法を適時かつ的確に適用できる。</p> <p>③ビジネスアプリケーションが適用される業務プロセスの現状に関し、その問題点を洗い出し、問題点を分析・評価するための判断基準を自ら形成できる。</p> <p>④監査結果を論理的に矛盾のない報告書にまとめ、説得力のある改善勧告を行うことができる。</p>

<p>⑤監査の実施に当たって必要となる情報技術及びその技術動向を理解できる。</p> <p>⑥外部環境の変化を捉え、組織の将来像を描き出すことができる。</p>
--

(14) 基本情報技術者試験：Fundamental Information Technology Engineer Examination

対象者像	情報技術全般に関する基本的な知識・技能をもつ者（情報システム開発プロジェクトにおいて、プログラム設計書を作成し、プログラムの開発を行い、単体テストまでの一連のプロセスを担当しているか、将来、そのような業務を担当する者を含む）
役割と業務	<p>情報システム開発プロジェクトにおいて、内部仕様に基づいてプログラムを設計・開発する業務に従事し、次の役割を果たす。</p> <p>①情報技術全般に関する基礎的な知識を活用し、システム開発プロジェクトの一員として貢献する。</p> <p>②与えられた内部設計書に基づいて、上位技術者の指導のもとにプログラム設計書を作成する。</p> <p>③標準的なアルゴリズムやデータ構造に関する知識に基づいて、プログラムを作成する。</p> <p>④作成したプログラムの単体テストを実施する。</p>
期待する技術水準	<p>情報技術全般に関する基礎的な知識を活用し、情報システム開発においてプログラムの設計・開発を行うとともに、将来高度な技術者を目指す者として、次の知識・技能が要求される。</p> <p>①情報技術全般に関する基本的な用語・内容を理解している。</p> <p>②上位技術者の指導のもとにプログラム設計書を作成できる。</p> <p>③プログラミングに必要な論理的思考能力をもつ。</p> <p>④一つ以上のプログラム言語の仕様を知っており、その言語を使ってプログラムを作成できる。</p> <p>⑤プログラムのテスト手法を知っており、単体テストを実施できる。</p>

■ 試験の出題形式と試験時間

試験区分	午前	午後 I	午後 II
	9:30～11:10 (100 分)	12:10～13:40 (90 分)	14:10～16:10 (120 分)
システムアナリスト試験 (秋期実施)	多肢選択式 (四肢択一) 55 問出題して 55 問解答 (共通問題)	記述式 4 問出題して 3 問解答	論述式 (小論文) 3 問出題して 1 問解答
プロジェクトマネージャ試験 (秋期実施)			
アプリケーションエンジニア 試験 (秋期実施)			
テクニカル エンジニア 試験	ネットワーク (秋期実施)	多肢選択式 (四肢択一) 55 問出題して 55 問解答	論述式 (事例解析) 2 問出題して 1 問解答
	データベース (春期実施)		論述式 (小論文) 3 問出題して 1 問解答
	システム管理 (春期実施)		論述式 (事例解析) 2 問出題して 1 問解答
	エンベデッドシステム (春期実施)		論述式 (小論文) 3 問出題して 1 問解答
情報セキュリティ (春期実施)			
上級システムアドミニスト レータ試験 (秋期実施)			論述式 (小論文) 3 問出題して 1 問解答
システム監査技術者試験 (春期実施)			

試験区分 (秋期実施)	午前	午後 I	午後 II
	9:30～11:10 (100 分)	12:10～13:40 (90 分)	14:10～15:40 (90 分)
情報セキュリティアドミニ ストレータ試験	多肢選択式 (四肢択一) 55 問出題して 55 問解答	記述式 4 問出題して 3 問解答	記述式 2 問出題して 1 問解答

試験区分 (春期・秋期実施)	午前	午後 I	午後 II
	9:30～12:00 (150 分)	13:00～15:00 (120 分)	15:30～16:30 (60 分)
ソフトウェア開発技術者試験	多肢選択式 (四肢択一) 80 問出題して 80 問解答	記述式 6 問出題して 6 問解答	記述式 1 問出題して 1 問解答

試験区分 (春期・秋期実施)	午前	午後
	9:30～12:00 (150 分)	13:00～15:30 (150 分)
初級システムアドミニスト レータ試験	多肢選択式 (四肢択一) 80 問出題して 80 問解答	多肢選択式 7 問出題して 7 問解答
基本情報技術者試験		多肢選択式 13 問出題して 7 問解答

■出題範囲

午前の試験では、受験者の能力が当該試験区分における“期待する技術水準”に達しているかどうかを、知識を問うことによって評価します。

午後の試験では、受験者の能力が当該試験区分における“期待する技術水準”に達しているかどうかを、技術の応用能力及び実務能力を問うことによって評価します。

〔午前の試験〕

各試験区分における午前の出題範囲は、次のとおりです。

(試験区分別出題範囲一覧表)

試験区分 分野	システムアナリスト	アプリケーションエンジニア プロジェクトマネージャ	ソフトウェア開発技術者	テクニカルエンジニア					情報セキュリティアドミニストレータ	上級システムアドミニストレータ	初級システムアドミニストレータ	システム監査技術者	基本情報技術者
				ネットワーク	データベース	システム管理	エンベデッドシステム	情報セキュリティ					
コンピュータ科学基礎			○Ⅲ										○Ⅱ
コンピュータシステム	○Ⅱ	○Ⅱ	○Ⅱ	◎Ⅱ	○Ⅱ	◎Ⅱ	◎Ⅲ	○Ⅱ	○Ⅱ	○Ⅱ	○Ⅰ	○Ⅱ	○Ⅰ
システムの開発と運用	◎Ⅲ	○Ⅱ	○Ⅱ	○Ⅱ	○Ⅱ	◎Ⅲ	○Ⅱ	○Ⅱ	○Ⅰ	◎Ⅱ	○Ⅰ	○Ⅱ	○Ⅰ
ネットワーク技術		○Ⅱ	○Ⅱ	◎Ⅲ		○Ⅱ	○Ⅱ	◎Ⅲ	○Ⅱ				○Ⅰ
データベース技術		○Ⅱ	○Ⅱ		◎Ⅲ	○Ⅱ		○Ⅱ					○Ⅰ
セキュリティと標準化	○Ⅱ	○Ⅱ	○Ⅱ	○Ⅲ	○Ⅱ	○Ⅱ	○Ⅱ	◎Ⅲ	◎Ⅲ	○Ⅱ	○Ⅰ	○Ⅲ	○Ⅰ
情報化と経営	◎Ⅲ								○Ⅱ	◎Ⅲ	◎Ⅰ	◎Ⅱ	○Ⅰ
監査									○Ⅱ			◎Ⅲ	

・○は出題範囲であることを，◎は出題範囲のうちの重点分野であることを表します。

・Ⅰ，Ⅱ，Ⅲは技術レベルを表し，Ⅲが最も高度で，ⅢはⅡ及びⅠを，ⅡはⅠを包含します。

コンピュータ科学基礎

1 情報の基礎理論

- 1-1 数値表現・データ表現に関すること
基数変換, 数値表現, 文字表現, 数値計算 (演算方式と精度, 近似解法と方程式ほか), 確率と統計, 最適化 など
- 1-2 情報と理論に関すること
論理演算, 符号理論, 述語論理, 状態遷移, 計算量, 情報量, BNF, ポーランド表記法, 集合 など

2 データ構造とアルゴリズム

- 2-1 データ構造に関すること
2分木, リスト, スタック, キュー など
- 2-2 アルゴリズムに関すること
整列, 探索, 再帰, グラフ, 文字列処理, 流れ図 など

コンピュータシステム

1 ハードウェア

- 1-1 情報素子に関すること
半導体素子, 集積回路の種類・特徴 など
- 1-2 プロセッサアーキテクチャに関すること
命令とアドレッシング, プロセッサの性能・構造・方式・特徴 など
- 1-3 メモリアーキテクチャに関すること
メモリアクセス, メモリの容量, メモリの構成・特徴 など
- 1-4 補助記憶に関すること
記憶媒体, 補助記憶装置の種類・特徴 など
- 1-5 入出力アーキテクチャと装置に関すること
入出力インタフェース, 周辺装置・通信装置 (ルータ, モデム, DSU ほか) の種類・特徴 など
- 1-6 コンピュータの種類と特徴に関すること
コンピュータ (パソコン, ワークステーションほか) の構成・種類・特徴 など
- 1-7 エンベデッドシステムに関すること
構成部品と実装, 論理設計, 論理回路, 制御理論, 信号理論 など

2 基本ソフトウェア

- 2-1 オペレーティングシステムに関すること
仮想記憶, 多重プログラミング, 記憶管理, OS の機能・種類・特徴, エンベデッド OS, エンベデッドシステム設計, デバイスドライバ など
- 2-2 ファイル管理に関すること
ファイル編成, アクセス手法, 排他制御, リカバリ処理, 検索手法 など

3 システムの構成と方式

- 3-1 システムの構成技術に関すること
クライアントサーバシステム, システムの構成方式・処理形態 (デュプレックス構成, デュアル構成, 分散コンピューティング, グリッドコンピューティング, ユビキタスコンピューティング, SAN, NAS ほか) など
- 3-2 システムの性能に関すること
システムの性能計算・性能設計・性能指標・性能評価, 待ち行列理論, キャパシティ管理 (コスト, 資源管理, 性能管理ほか) など

- 3-3 システムの信頼性・経済性に関すること
システムの信頼性計算・信頼性設計・信頼性指標・信頼性評価・経済性、稼働率、故障率、バスタブ曲線 など
- 4 システム応用
 - 4-1 ネットワーク応用に関すること
Web, CGI, インターネット, イントラネット, エクストラネット, 通信サービス, モバイル通信, 衛星通信システム, プロトコル (TCP/IP, UDP, IPv6 ほか), LAN の構成・種類・特徴 など
 - 4-2 データベース応用に関すること
データウェアハウス, データマイニング, データマート, SQL の利用 など
 - 4-3 データ資源管理に関すること
IRDS, メタデータ, リポジトリ など
 - 4-4 マルチメディアシステムに関すること
AI, パターン処理, AR/VR/CG, エージェント, メディア応用 など

システムの開発と運用

- 1 システムの開発
 - 1-1 言語に関すること
プログラム構造, データ型, 言語処理系, 構文解析, 言語 (C, COBOL, Java™, SQL, HTML ほか) の種類・特徴 など
 - 1-2 ソフトウェアパッケージに関すること
表計算ソフト, グループウェア, ミドルウェア, オープンソースソフトウェア など
 - 1-3 開発環境に関すること
開発ツール, EUC・EUD, デバッグ技法 など
 - 1-4 開発手法に関すること
プロセスモデル, ソフトウェア開発手法, 再利用手法, Web アプリケーション開発, ソフトウェアの品質保証 など
 - 1-5 要求分析・設計手法に関すること
DFD, E-R 図, UML, オブジェクト指向設計, プロセス中心設計, データ中心設計, モジュール設計, 入出力設計, ヒューマンインタフェース設計, ドキュメント設計, Web デザイン など
 - 1-6 プログラミング, テスト, レビューに関すること
プログラミング手法 (オブジェクト指向プログラミング, 並列処理プログラミングほか), テスト手法, レビュー手法, テスト設計・管理手法 など
 - 1-7 開発管理に関すること
プロジェクト計画, 見積手法, 品質管理, 工程管理, 日程管理, コスト管理 (アーンドバリュウほか), 構成管理, 要員計画・管理, ドキュメント管理, 開発メンバの役割と構成, プレゼンテーション技法, コミュニケーション技法, システムの可監査性 など
 - 1-8 外部環境の活用に関すること
アウトソーシング, システムインテグレーション など
- 2 システムの運用と保守
 - 2-1 システムの運用に関すること
システムの障害管理, 移行, オペレーション, 運用ツール, 資源管理, コスト管理, ユーザ管理, ITIL, 設備・施設とその設計・施工・維持運営 など
 - 2-2 システムの保守に関すること
保守の形態, 保守契約, ソフトウェア保守 など

※ Java は, 米国及びその他の国における米国 Sun Microsystems, Inc. の商標又は登録商標です。

ネットワーク技術

1 ネットワーク技術

- 1-1 プロトコル・伝送制御に関すること
ネットワークアーキテクチャ, 通信プロトコルとインタフェース, OSIの階層 など
- 1-2 符号化・伝送に関すること
誤り制御, 変調・符号化, 多重化方式, 交換方式, 伝送方式 など
- 1-3 ネットワーク (LAN・WAN) に関すること
LAN, WAN, インターネット技術, ネットワーク関連法規, 電気通信サービス など
- 1-4 通信装置に関すること
LAN間接続装置, 回線接続装置, 伝送・交換装置, 通信媒体 など
- 1-5 ネットワークソフトに関すること
ネットワーク管理, ネットワーク OS など

データベース技術

1 データベース技術

- 1-1 データベースのモデルに関すること
データモデル (関係モデル, E-Rモデルほか), 分析, 正規化, 操作 など
- 1-2 データベースの言語に関すること
SQL など
- 1-3 データベースの制御に関すること
データベースの排他制御・リカバリ, トランザクション管理, 分散データベース, DBMSの機能・特徴 など

セキュリティと標準化

1 セキュリティ

- 1-1 セキュリティに関すること
暗号化, 認証, アクセス管理, セキュリティ管理, 安全対策, 設備・施設管理, コンピュータウイルス, プライバシ保護, セキュリティポリシー, 個人情報保護, ネットワークセキュリティ, データベースセキュリティ, セキュリティ関連法規 など
- 1-2 リスク管理に関すること
リスクの分析・対策・種類, 内部統制 など
- 1-3 ガイドラインに関すること
情報システム安全対策基準, ソフトウェア管理ガイドライン, コンピュータウイルス対策基準, コンピュータ不正アクセス対策基準, 情報セキュリティ監査制度 など

2 標準化

- 2-1 開発と取引の標準化に関すること
ISO 9000, SLCP-JCF98 など
- 2-2 情報システム基盤の標準化に関すること
OSI, IEEE, EDIFACT, OMG, CORBA, RFC, ISMS, ISO/IEC 15408 など
- 2-3 データの標準化に関すること
各種コード (文字コード, バーコード, 2次元コードほか), データフォーマット, データの圧縮 など
- 2-4 標準化組織に関すること
内外の標準化組織の役割と活動内容 など

情報化と経営

1 情報戦略

1-1 経営管理に関すること

経営戦略, 組織, 経営管理, マーケティング, 行動科学, ヒューマンリソースマネジメント など

1-2 情報化戦略に関すること

情報化構想, システム化計画, 業務改善, 業務分析・設計, 企業情報システム など

2 企業会計

2-1 財務会計に関すること

会計基準, 財務諸表, 連結会計, 減価償却 など

2-2 管理会計に関すること

損益分岐点, 経済性計算, 経営・財務指標, 原価, リース・レンタル, 資金計画・管理, 資産管理 など

3 経営工学

3-1 IE・OR系に関すること

分析手法 (作業分析, 工程分析, 予測, 在庫管理, 線形計画法, 待ち行列, PERT ほか), 品質管理 (OC 曲線, 管理図, パレート図ほか), 最適化問題, 統計的手法 など

4 情報システムの活用

4-1 エンジニアリングシステムに関すること

MRP, 生産管理システム, 生産計画, 工程計画・管理 など

4-2 ビジネスシステムに関すること

経理・財務・人事システム, 営業支援システム, POS, 流通システム, 金融システム, 公共システム, 企業間システム, 電子決済システム など

5 関連法規

5-1 情報通信に関すること

電気通信事業法 など

5-2 知的財産権に関すること

著作権法, 産業財産権法, 不正競争防止法 など

5-3 労働に関すること

労働者派遣法, 男女雇用機会均等法, 労働基準法, 労働安全衛生法 など

5-4 取引に関すること

外注契約, ソフトウェア契約, 環境関連法規 など

5-5 安全に関すること

不正アクセス防止法, 個人情報保護法, PL法 など

5-6 その他の法律・倫理に関すること

刑法, 会社法, 商法, 電子帳簿保存法, 情報公開, 認定制度 など

監査

1 監査

1-1 システム監査に関すること

システム監査基準, 監査証跡, 監査調書, 監査技法, 監査計画, 監査の調査・実施・評価・報告・意義, システムの可監査性, システム監査人, セキュリティ監査 など

1-2 一般監査に関すること

業務監査, 会計監査 など

〔午後の試験〕

各試験区分における午後の出題範囲は、次のとおりです。

システムアナリスト試験

- 1 情報戦略の策定と情報システムの構築構想に関すること
経営戦略に基づく情報戦略の策定，情報技術によるビジネスモデルの開発提案，情報技術による業務革新の企画，システムソリューションの選択，アウトソーシング戦略の策定など
- 2 情報システムの全体計画の立案・推進に関すること
業務モデル定義，情報システム全体体系定義，情報システム体系モデル定義，情報システムの開発課題分析と優先順位付け，情報システム基盤構成方針策定，システムソリューション適用方針策定（ERP パッケージの適用ほか），中長期情報システム化計画の策定・推進 など
- 3 個別システムの開発計画の立案・推進に関すること
業務・組織・情報システム構造の現状分析，業務革新案策定，業務要件とシステム化要件の明確化，システムソリューションの適用方法検討，システム概要設計，システム構築計画策定，システム化の効果・コスト・リスクの分析・評価，開発計画に対する評価ポイントの作成 など

プロジェクトマネージャ試験

- 1 プロジェクトの計画立案に関すること
プロジェクトの目標・制約条件と運営方針の策定，工程計画，品質計画，予算の策定と実行計画，資源・工数・要員などの計画，調達計画，リスク分析とリスク対応計画，提案依頼（RFP），プロジェクトの意思決定手順，関連法規・標準 など
- 2 プロジェクトの運営・管理に関すること
プロジェクト管理技法と適用技術，要件と見積り，工程管理，品質管理，予算・費用管理，組織・要員管理，調達管理，リスク管理，機密・契約管理，変更管理，コミュニケーション管理，リーダーシップ，要員育成など人的側面 など
- 3 プロジェクトの評価に関すること
プロジェクト評価手法と適用技術，取得データの分析と評価，プロジェクト完了報告の取りまとめ，プロジェクト実績評価・分析，検収結果の評価，契約遵守状況評価，プロジェクト完成記録 など

アプリケーションエンジニア試験

- 1 システム分析・要求定義に関すること
システム分析技法の検討，対象業務の調査・分析，業務要件の定義，論理データフローの作成，データ分析，データの標準化・正規化，システム要求分析，業務機能・機能階層の定義，パッケージ導入方法の検討 など
- 2 システム設計・開発に関すること
 - 2-1 外部設計

- システム機能設計，コード設計，論理データの設計，画面・帳票の設計，外部システムとのインタフェースの設計 など
- 2-2 内部設計
分散方式の設計，システム機能の分割，処理方式の検討，ファイル・データベースの設計 など
- 2-3 システム方式設計
システム構成の設計，移行方式・運用設計，性能・信頼性の設計，セキュリティ検討 など
- 3 テストに関すること
テスト計画の作成，工程別テスト方式の検討，テスト内容の設計 など
- 4 開発管理・開発環境に関すること
工程管理，コスト管理，品質管理，仕様変更管理，デザインレビューと承認，標準化，プログラミング規約の作成，CASE ツールの利用 など

ソフトウェア開発技術者試験

- 1 ソフトウェア工学に関すること
ソフトウェア開発におけるモデル，ソフトウェアの要求定義，ソフトウェア設計技法，プログラミングパラダイム，ソフトウェアのテストと品質，ソフトウェアの開発環境，オブジェクト指向分析・設計 など
- 2 アルゴリズムに関すること
検索，照合，整列，データ圧縮，記憶域管理などのアルゴリズム，数値計算に関するアルゴリズム，計算量の多いアルゴリズム，自然言語処理におけるアルゴリズム，言語プロセッサにおけるアルゴリズム など
- 3 システム構成技術に関すること
集中処理，分散処理，クライアントサーバ，Web 技術，イントラネットなどのシステム構成，高信頼度システム構成，キャパシティプランニング，システムパラメタの種類と算定，システム統合 など
- 4 システム開発に関すること
内部設計（入出力設計，データ設計，部品化と再利用，デザインレビューほか），プログラム設計・開発（プログラム作成基準，モジュール分割技法，モジュール設計，プログラム作成技法，ソースコードレビューほか），テスト（テスト計画，テスト技法，テスト設計，テストの実施ほか） など
- 5 通信ネットワークに関すること
通信ネットワークを利用したシステム，ネットワークの種類，データ伝送技術，ネットワークプロトコル，ネットワークアーキテクチャ，インターネット技術，通信トラフィック，ネットワーク管理 など
- 6 データ構造及びデータベースに関すること
情報資源管理，データモデル，正規化，データベース管理システム，データベース言語，データベースの種類，データベース設計，データベースの作成と運用 など

- 7 情報セキュリティに関すること
セキュリティマネジメント、アクセス管理、暗号、認証、ファイアウォール、安全性対策、機密対策、インテグリティ対策、セキュリティ規程 など
- 8 システム評価に関すること
開発要件に関する評価、システム稼働状態に関する評価、システム全体の中での整合性評価、ハードウェア資源の使用率 など

テクニカルエンジニア（ネットワーク）試験

- 1 ネットワークシステムの設計・構築に関すること
ネットワークシステム（データ・音声・画像、LAN・WAN を含む）の要求分析、論理設計、物理設計、信頼性設計、性能設計、セキュリティ設計、アドレス設計、運用設計、インプリメンテーション、テスト、移行、評価（性能、信頼性、品質、経済性ほか）、改善提案 など
- 2 ネットワークシステムの運用・保守に関すること
ネットワークシステムの運用・保守管理、運用・保守体制、セキュリティ管理、セキュリティ管理体制 など
- 3 ネットワーク技術・関連法規・標準化に関すること
ネットワーク構成要素、待ち行列理論、トラフィック技術、ネットワーク構成技術、ネットワークセキュリティ技術、ネットワーク関連法規及び倫理規定、ネットワークの標準化 など
- 4 ネットワークサービス活用に関すること
市場で実現している又は実現しつつある各種ネットワークサービスの利用技術、評価技術 など

テクニカルエンジニア（データベース）試験

- 1 データベースシステムの設計・構築に関すること
データベースシステムの要求分析、概念データモデルの設計、コード設計、物理データベースの設計・作成、データ操作の設計、性能見積り など
- 2 データベースシステムの運用・保守に関すること
データベースの運用・保守・管理、運用管理体制（データベース管理者・データ管理者）、チューニング、再編成、再構成、バックアップ、リカバリ、データ移行、セキュリティ管理 など
- 3 データベース技術に関すること
リポジトリ、関係モデル、関係代数、正規化、データベース管理システム、SQL など

テクニカルエンジニア（システム管理）試験

- 1 情報システム基盤（業務システム共有のシステム資源）の企画・構築に関すること
 - 1-1 資源管理
ハードウェア・ソフトウェア・データ・ネットワーク資源管理，構成管理，施設・設備管理 など
 - 1-2 セキュリティ管理
セキュリティの基準，物理的セキュリティ対策，システムのセキュリティ対策，管理的セキュリティ対策 など
 - 1-3 性能管理
性能分析・評価，キャパシティ計画 など
- 2 情報システム基盤の運用に関すること
 - 2-1 運用管理
運用計画，運用体制・運用スケジュール，データ入出力管理，課金管理・会計管理，障害管理，運用管理システム，ネットワーク管理システム，利用者支援（ヘルプデスクほか）など
 - 2-2 システム評価
システム運用標準の設定，サービス水準の設定，システム評価，システム改善 など
 - 2-3 新規システムの受入れ・システム移行に関すること
運用テストの実施・評価，受入れ評価，移行計画，移行実施 など
- 3 システム管理技術に関すること
システム構成技術，システム運用管理技術 など

テクニカルエンジニア（エンベデッドシステム）試験

- 1 エンベデッドシステムの設計・構築に関すること
エンベデッドシステムの要求分析，エンベデッドシステム開発工程の設計，コスト設計，性能設計，機能安全，ユーザビリティ，ハードウェアとソフトウェアのトレードオフ，機能分割設計，高信頼性設計，全体性能の予測，省電力設計，テスト手法の検討，開発環境の設計 など
- 2 エンベデッドシステムのソフトウェア設計に関すること
リアルタイム OS の応用，カーネルの設計，タスク設計，共有資源設計，デバイスドライバの設計，ファイル管理の設計，通信プロトコルスタックの設計，マルチメディア処理ソフトの作成，実行環境の検討，ソフトウェアの実装 など
- 3 エンベデッドシステムのハードウェア設計に関すること
MPU アーキテクチャの設計，論理設計，データハザードの検討，メモリ階層の設計，周辺デバイスの検討，通信インタフェースの設計，高位ハードウェア設計言語の活用，高信頼化設計，故障解析，電気・機械まわりの検討 など

テクニカルエンジニア（情報セキュリティ）試験

- 1 情報セキュリティシステムの企画・設計・構築に関すること
情報システムの企画・設計・構築，物理的セキュリティ対策，アプリケーションセキュリティ対策，データベースセキュリティ対策，セキュアプログラミング，ネットワークセキ

セキュリティ対策, システムセキュリティ対策 など

- 2 情報セキュリティの運用・管理に関すること
情報セキュリティポリシー, リスク分析, 業務継続計画, セキュリティ運用・管理, 脆弱性分析, 誤使用分析, ユーザセキュリティ管理, 障害復旧計画, 情報セキュリティ教育, システム監査 (のセキュリティ側面) など
- 3 情報セキュリティ技術・関連法規に関すること
アクセス管理技術, ウイルス対策技術, 暗号技術, 認証技術, セキュリティ応用システム (署名, 侵入検知システム, ファイアウォール, セキュアな通信技術 (VPN など), 鍵管理技術, PKI など), 攻撃手法, 監査証跡のためのログ管理技術, 耐タンパ技術, 情報セキュリティ関連法規, 国内・国際標準, ガイドライン, 著作権法, 個人情報保護, 情報倫理 など
- 4 開発の管理に関すること
開発ライフサイクル管理, システム文書構成管理, 配布と操作, 人的管理手法 (チーム内の不正を起こさせないような仕組み), 開発環境の情報セキュリティ管理 など

情報セキュリティアドミニストレータ試験

- 1 情報セキュリティシステムの企画・設計・構築に関すること
情報戦略, 情報システム (ネットワークを含む) の企画・設計・構築, 開発管理, 物理的セキュリティ対策, アプリケーションセキュリティ対策, データベースセキュリティ対策, ネットワークセキュリティ対策, システムセキュリティ対策 など
- 2 情報セキュリティの運用・管理に関すること
セキュリティポリシーの策定・評価・見直し, リスク分析, 業務継続計画, セキュリティ運用・管理, 脆弱性分析, 不正アクセス検知・対策, ユーザセキュリティ管理, 障害復旧計画, セキュリティ教育, 契約管理, 要員管理, システム監査 (のセキュリティ側面) など
- 3 情報セキュリティの技術・関連法規に関すること
アクセス管理技術, ウイルス対策技術, 暗号技術, 認証技術, 署名技術, 情報漏えい防止技術, 情報セキュリティ関連法規, 国内・国際標準, ガイドライン, 著作権法, 個人情報保護, 情報倫理 など

上級システムアドミニストレータ試験

- 1 業務システム改善企画の立案に関すること
業務体系の把握, 業務内容の調査・分析, 情報技術を活用した業務改革・改善, 業務システム改善の企画, 業務システム改善案の費用対効果分析・優先順位付け・事後評価, 情報システムの企画・提案・実現・評価 など
- 2 情報システム構築のためのマネジメントに関すること
機能・性能要求の設定, 安全性・信頼性・障害対策要求の設定, 運用・保守要求の設定, ソフトウェアパッケージの選定, ヒューマンインタフェースの設計・開発, システム企画・システム設計・運用計画のレビュー, テスト基準・テスト手順の作成, テストの実施と評価, 検収 など

- 3 情報システム利用のためのマネジメントに関すること
システム利用の促進，情報化推進のための教育体制，教育メニューの立案，情報化推進のための組織・体制の立案，業務マニュアル・運用マニュアルの整備・オンライン化，データの活用，システム運用・情報システムの状況把握，セキュリティ対策，知的所有権 など

初級システムアドミニストレータ試験

- 1 仕事とコンピュータに関すること
システムアドミニストレータの役割，仕事の進め方の把握・改善，コンピュータの使い方，問題発見の手法，問題解決の手法，データ分析の手法 など
- 2 基幹業務システムとのかかわりに関すること
基幹業務システムの概要，システム運用の概要，ユーザ要求の定義，テストと検収 など
- 3 エンドユーザコンピューティング（EUC）に関すること
EUC の概要，パソコンのハードウェア・ソフトウェア，表計算ソフトの利用，データベースソフトの利用（SQL によるデータ操作を含む），ヒューマンインタフェース設計，ネットワークの種類と仕組み，クライアントサーバシステム，マルチメディア，グループウェア，インターネット，イントラネット，電子メール，Web 技術 など
- 4 システム環境整備と運用管理に関すること
ハードウェアとソフトウェアの選定，ハードウェアとソフトウェアの利用環境の整備，ネットワークの利用と運用，構成管理，ファイル管理，性能・障害管理の支援，セキュリティ管理の支援，ウイルス対策，暗号の利用，不正アクセス対策，権利の保護とエチケット など
- 5 情報化推進のための表現能力に関すること
発表技術，分かりやすい文章にするための工夫，用字・用語の使い方，文章の組み立て方と文書作成の手順，ビジュアル表現，情報伝達・情報発信におけるツールの効果的な利用 など

システム監査技術者試験

- 1 情報システム・通信ネットワーク・システム監査全般に関すること
経営一般，情報戦略，情報システム・通信ネットワークの評価，情報システム・通信ネットワークのリスク管理，情報システム・通信ネットワークの品質管理，リスク分析の手法，情報セキュリティ，セキュリティポリシー，事業継続管理，デジタル認証，内部統制（全般統制，アプリケーション統制），IT ガバナンス，他の監査（会計監査，業務監査）との連携・調整，情報技術動向の把握 など
- 2 システム監査の計画に関すること
監査目的の設定，リスクアプローチ，中長期計画書・基本計画書・個別計画書の作成 など
- 3 システム監査の実施に関すること

実施準備, 予備調査, 監査手続, 本調査 (企画・開発業務の監査, 運用・保守業務の監査), 評価・結論, 監査調書の作成・保管, 監査証拠の収集・評価, システム監査技法の適用, システム監査業務の管理 (監査業務の品質管理を含む) など

- 4 システム監査の報告に関すること
監査報告書の作成, 監査意見の表明 (保証意見, 助言意見), フォローアップの実施 など
- 5 システム監査関連法規に関すること
セキュリティ関連法規, 個人情報保護関連法規, 知的財産権関連法規, 労働関連法規, 法定監査関連法規, システム監査及びセキュリティ監査に関する基準・ガイドライン・施策, 内部監査に関する基準・ガイドライン・施策 など

基本情報技術者試験

- 1 ハードウェアに関すること
数値の表現, 文字の表現, 画像・音声の表現, 処理装置, 記憶装置, 入出力装置, 演算の実行, アドレス方式, 入出力の実行, システム構成 など
- 2 ソフトウェアに関すること
システムソフトウェア, アプリケーションソフトウェア, ソフトウェアパッケージ, OSの機能, プログラム言語, 言語プロセッサ, プログラムの実行 など
- 3 アルゴリズムに関すること
整列, 探索, 文字列処理, ファイル処理, 図形, グラフ, 数値計算 など
- 4 データ構造及びデータベースに関すること
基本データ構造, 記憶媒体の種類と特徴, ファイルの編成方法, データベースの種類と特徴, データベース言語, SQLによるデータ操作 など
- 5 通信ネットワークに関すること
データ伝送, 伝送制御, TCP/IP, LAN, WAN, インターネット, 電子メール, Web など
- 6 情報処理技術に関すること
システムの性能, システムの信頼性, リスク管理, セキュリティ, 標準化, オペレーションズリサーチ など
- 7 プログラム設計に関すること
システム開発工程, プログラム設計工程, 構造化設計, モジュール設計, プログラム設計書 など
- 8 プログラム開発に関すること
プログラム言語 (C, COBOL, Java, アセンブラ), コーディング, 開発環境, テスト手法 など

■試験で使用する情報技術に関する用語・プログラム言語など

(1) 情報技術に関する用語

試験で使用する情報技術に関する用語は、日本工業規格（JIS）に制定されているものについては、その規定に従うものとします。

(2) 記号・図など

試験で使用する代表的な記号・図などは、次の仕様に従うものとします。次以外については、問題文中で定義するものとします。

情報処理用流れ図など	: JIS X 0121
決定表	: JIS X 0125
計算機システム構成の図記号	: JIS X 0127
プログラム構成要素及びその表記法	: JIS X 0128

(3) プログラム言語

① 基本情報技術者試験において、プログラミング能力を問う試験問題に出題するプログラム言語は、C、COBOL、Java 及びアセンブラ言語（CASL II）の 4 言語とします。

② テクニカルエンジニア（情報セキュリティ）試験において、セキュアプログラミングに関する試験問題に出題するプログラム言語は、C++、Java、Perl の 3 言語のいずれかとします。

③ 仕様などは、次によります。

C	: JIS X 3010
COBOL	: JIS X 3002
Java	: The Java Language Specification, Third Edition (JLS 3.0) ⁽¹⁾ (URL http://java.sun.com/docs/books/jls/index.html)
アセンブラ言語	: 26 ページの「アセンブラ言語の仕様」によります。
C++	: JIS X 3014
Perl	: 37 ページの「プログラム言語 Perl の用例・解説」によります。

注⁽¹⁾ なお、JLS 3.0 で追加された機能については、主に次の機能を問題中で使用します。ただし、これらの 3 機能に限定するものではありません。

- ・ Generics（総称）
- ・ 拡張された for 文
- ・ enum（列挙）型

また、メタデータは範囲外とします。

(4) データベース言語

試験で使用するデータベース言語は、次の仕様に従うものとします。

SQL	: JIS X 3005
-----	--------------

(5) 表計算ソフトなどのソフトウェアパッケージ

表計算ソフト : 34 ページの「表計算ソフトの機能・用語」によります。ここに規定されていない機能・用語などについては、問題文中で定義するものとします。

表計算ソフト以外のソフトウェアパッケージの機能・用語などは、問題文中で定義するものとします。

JIS の参照 (日本工業標準調査会ホームページ)

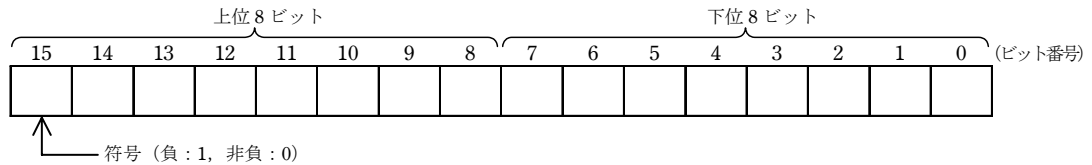
URL : <http://www.jisc.go.jp/>

■アセンブラ言語の仕様

1. システム COMET II の仕様

1.1 ハードウェアの仕様

- (1) 1語は16ビットで、そのビット構成は、次のとおりである。



- (2) 主記憶の容量は65536語で、そのアドレスは0～65535番地である。
 (3) 数値は、16ビットの2進数で表現する。負数は、2の補数で表現する。
 (4) 制御方式は逐次制御で、命令語は1語長又は2語長である。
 (5) レジスタとして、GR (16ビット)、SP (16ビット)、PR (16ビット)、FR (3ビット) の4種類がある。

GR (汎用レジスタ, General Register) は、GR0～GR7の8個があり、算術、論理、比較、シフトなどの演算に用いる。このうち、GR1～GR7のレジスタは、指標レジスタ (index register) としてアドレスの修飾にも用いる。

SP (スタックポインタ, Stack Pointer) は、スタックの最上段のアドレスを保持している。

PR (プログラムレジスタ, Program Register) は、次に実行すべき命令語の先頭アドレスを保持している。

FR (フラグレジスタ, Flag Register) は、OF (Overflow Flag)、SF (Sign Flag)、ZF (Zero Flag) と呼ぶ3個のビットからなり、演算命令などの実行によって次の値が設定される。これらの値は、条件付き分岐命令で参照される。

OF	算術演算命令の場合は、演算結果が-32768～32767に収まらなくなったとき1になり、それ以外るとき0になる。論理演算命令の場合は、演算結果が0～65535に収まらなくなったとき1になり、それ以外るとき0になる。
SF	演算結果の符号が負 (ビット番号15が1) のとき1、それ以外るとき0になる。
ZF	演算結果が零 (全部のビットが0) のとき1、それ以外るとき0になる。

- (6) 論理加算又は論理減算は、被演算データを符号のない数値とみなして、加算又は減算する。

1.2 命令

命令の形式及びその機能を示す。ここで、一つの命令コードに対し2種類のオペランドがある場合、上段はレジスタ間の命令、下段はレジスタと主記憶間の命令を表す。

命 令	書 き 方		命 令 の 説 明	FRの設定
	命 令 コード	オペランド		

(1) ロード、ストア、ロードアドレス命令

ロード LoaD	LD	r1, r2 r, adr [, x]	r1 ← (r2) r ← (実効アドレス)	○*1
ストア STore	ST	r, adr [, x]	実効アドレス ← (r)	
ロードアドレス Load Address	LAD	r, adr [, x]	r ← 実効アドレス	

(2) 算術, 論理演算命令

算術加算 ADD Arithmetic	ADDA	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) + (r2)$ $r \leftarrow (r) + (\text{実効アドレス})$	○
論理加算 ADD Logical	ADDL	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) +_L (r2)$ $r \leftarrow (r) +_L (\text{実効アドレス})$	
算術減算 SUBtract Arithmetic	SUBA	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) - (r2)$ $r \leftarrow (r) - (\text{実効アドレス})$	
論理減算 SUBtract Logical	SUBL	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) -_L (r2)$ $r \leftarrow (r) -_L (\text{実効アドレス})$	
論理積 AND	AND	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) \text{ AND } (r2)$ $r \leftarrow (r) \text{ AND } (\text{実効アドレス})$	○*1
論理和 OR	OR	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) \text{ OR } (r2)$ $r \leftarrow (r) \text{ OR } (\text{実効アドレス})$	
排他的論理和 eXclusive OR	XOR	$r1, r2$ $r, \text{adr} [, x]$	$r1 \leftarrow (r1) \text{ XOR } (r2)$ $r \leftarrow (r) \text{ XOR } (\text{実効アドレス})$	

(3) 比較演算命令

算術比較 ComPare Arithmetic	CPA	$r1, r2$ $r, \text{adr} [, x]$	($r1$) と ($r2$), 又は (r) と (実効アドレス) の算術比較又は論理比較を行い, 比較結果によって, FR に次の値を設定する。	○*1		
論理比較 ComPare Logical	CPL	$r1, r2$ $r, \text{adr} [, x]$	比較結果		FR の値	
					SF	ZF
			($r1$) > ($r2$)		0	0
			(r) > (実効アドレス)			
			($r1$) = ($r2$)		0	1
			(r) = (実効アドレス)			
			($r1$) < ($r2$)	1	0	
			(r) < (実効アドレス)			

(4) シフト演算命令

算術左シフト Shift Left Arithmetic	SLA	$r, \text{adr} [, x]$	符号を除き (r) を実効アドレスで指定したビット数だけ左又は右にシフトする。	○*2
算術右シフト Shift Right Arithmetic	SRA	$r, \text{adr} [, x]$	シフトの結果, 空いたビット位置には, 左シフトのときは 0, 右シフトのときは符号と同じものが入る。	
論理左シフト Shift Left Logical	SLL	$r, \text{adr} [, x]$	符号を含み (r) を実効アドレスで指定したビット数だけ左又は右にシフトする。	
論理右シフト Shift Right Logical	SRL	$r, \text{adr} [, x]$	シフトの結果, 空いたビット位置には 0 が入る。	

(5) 分岐命令

正分岐 Jump on Plus	JPL	$\text{adr} [, x]$	FR の値によって, 実効アドレスに分岐する。分岐しないときは, 次の命令に進む。	—																											
負分岐 Jump on Minus	JMI	$\text{adr} [, x]$	<table border="1"> <tr> <th rowspan="2">命令</th> <th colspan="3">分岐するときの FR の値</th> </tr> <tr> <th>OF</th> <th>SF</th> <th>ZF</th> </tr> <tr> <td>JPL</td> <td></td> <td>0</td> <td>0</td> </tr> <tr> <td>JMI</td> <td></td> <td>1</td> <td></td> </tr> <tr> <td>JNZ</td> <td></td> <td></td> <td>0</td> </tr> <tr> <td>JZE</td> <td></td> <td></td> <td>1</td> </tr> <tr> <td>JOV</td> <td>1</td> <td></td> <td></td> </tr> </table>		命令	分岐するときの FR の値			OF	SF	ZF	JPL		0	0	JMI		1		JNZ			0	JZE			1	JOV	1		
命令	分岐するときの FR の値																														
	OF	SF			ZF																										
JPL		0			0																										
JMI		1																													
JNZ					0																										
JZE					1																										
JOV	1																														
非零分岐 Jump on Non Zero	JNZ	$\text{adr} [, x]$																													
零分岐 Jump on Zero	JZE	$\text{adr} [, x]$																													
オーバーフロー分岐 Jump on Overflow	JOV	$\text{adr} [, x]$																													
無条件分岐 unconditional JUMP	JUMP	$\text{adr} [, x]$	無条件に実効アドレスに分岐する。																												

(6) スタック操作命令

プッシュ PUSH	PUSH adr [,x]	SP ← (SP) - _L 1, (SP) ← 実効アドレス	—
ポップ POP	POP r	r ← (SP), SP ← (SP) + _L 1	—

(7) コール, リターン命令

コール CALL subroutine	CALL adr [,x]	SP ← (SP) - _L 1, (SP) ← (PR), PR ← 実効アドレス	—
リターン RETurn from subroutine	RET	PR ← ((SP)), SP ← (SP) + _L 1	—

(8) その他

スーパーバイザコール SuperVisor Call	SVC adr [,x]	実効アドレスを引数として割出しを行 う。実行後の GR と FR は不定となる。	—
ノーオペレーション No OPeration	NOP	何もしない。	—

- (注) r, r1, r2 いずれも GR を示す。指定できる GR は GR0 ~ GR7
 adr アドレスを示す。指定できる値の範囲は 0 ~ 65535
 x 指標レジスタとして用いる GR を示す。指定できる GR は GR1 ~ GR7
 [] [] 内の指定は省略できることを示す。
 () () 内のレジスタ又はアドレスに格納されている内容を示す。
 実効アドレス adr と x の内容との論理加算値又はその値が示す番地
 ← 演算結果を, 左辺のレジスタ又はアドレスに格納することを示す。
 +L, -L 論理加算, 論理減算を示す。
 FR の設定 : 設定されることを示す。
 ○*1 : 設定されることを示す。ただし, OF には 0 が設定される。
 ○*2 : 設定されることを示す。ただし, OF にはレジスタから最後に送り
 出されたビットの値が設定される。
 — : 実行前の値が保持されることを示す。

1.3 文字の符号表

- (1) JIS X 0201 ラテン文字・片仮名用 8 ビット符
号で規定する文字の符号表を使用する。
 (2) 右に符号表の一部を示す。1 文字は 8 ビット
からなり, 上位 4 ビットを列で, 下位 4 ビット
を行で示す。例えば, 間隔, 4, H, ¥ のビット
構成は, 16 進表示で, それぞれ 20, 34, 48,
5C である。16 進表示で, ビット構成が 21 ~
7E (及び表では省略している A1 ~ DF) に対応
する文字を図形文字という。図形文字は, 表示
(印刷) 装置で, 文字として表示 (印字) できる。
 (3) この表にない文字とそのビット構成が必要な
場合は, 問題中で与える。

行 \ 列	02	03	04	05	06	07
0	間隔	0	@	P	`	p
1	!	1	A	Q	a	q
2	"	2	B	R	b	r
3	#	3	C	S	c	s
4	\$	4	D	T	d	t
5	%	5	E	U	e	u
6	&	6	F	V	f	v
7	'	7	G	W	g	w
8	(8	H	X	h	x
9)	9	I	Y	i	y
10	*	:	J	Z	j	z
11	+	;	K	[k	{
12	,	<	L	¥	l	
13	-	=	M]	m	}
14	.	>	N	^	n	~
15	/	?	O	_	o	

2. アセンブラ言語 CASL II の仕様

2.1 言語の仕様

- (1) CASL II は、COMET II のためのアセンブラ言語である。
- (2) プログラムは、命令行及び注釈行からなる。
- (3) 1 命令は 1 命令行で記述し、次の行へ継続できない。
- (4) 命令行及び注釈行は、次に示す記述の形式で、行の 1 文字目から記述する。

行の種類	記述の形式
命令行	オペランドあり [ラベル] {空白} {命令コード} {空白} {オペランド} [{空白} [コメント]]
	オペランドなし [ラベル] {空白} {命令コード} [{空白} [;] [コメント]]
注釈行	[空白] { ; } [コメント]

- (注) [] [] 内の指定が省略できることを示す。
 { } { } 内の指定が必須であることを示す。
 ラベル その命令の (先頭の語の) アドレスを他の命令やプログラムから参照するための名前である。長さは 1～8 文字で、先頭の文字は英大文字でなければならない。以降の文字は、英大文字又は数字のいずれでもよい。なお、予約語である GR0～GR7 は、使用できない。
 空白 1 文字以上の間隔文字の列である。
 命令コード 命令ごとに記述の形式が定義されている。
 オペランド 命令ごとに記述の形式が定義されている。
 コメント 覚え書きなどの任意の情報であり、処理系で許す任意の文字を書くことができる。

2.2 命令の種類

命令は、4 種類のアセンブラ命令 (START, END, DS, DC), 4 種類のマクロ命令 (IN, OUT, RPUSH, RPOP) 及び機械語命令 (COMET II の命令) からなる。その仕様を次に示す。

命令の種類	ラベル	命令コード	オペランド	機能
アセンブラ命令	ラベル	START	[実行開始番地]	プログラムの先頭を定義 プログラムの実行開始番地を定義 他のプログラムで参照する入口名を定義
		END		プログラムの終わりを明示
	[ラベル]	DS	語数	領域を確保
	[ラベル]	DC	定数 [, 定数] …	定数を定義
マクロ命令	[ラベル]	IN	入力領域, 入力文字長領域	入力装置から文字データを入力
	[ラベル]	OUT	出力領域, 出力文字長領域	出力装置へ文字データを出力
	[ラベル]	RPUSH		GR の内容をスタックに格納
	[ラベル]	RPOP		スタックの内容を GR に格納
機械語命令	[ラベル]		(「1.2 命令」を参照)	

2.3 アセンブラ命令

アセンブラ命令は、アセンブラの制御などを行う。

- (1)

START	[実行開始番地]
-------	----------

START 命令は、プログラムの先頭を定義する。
 実行開始番地は、そのプログラム内で定義されたラベルで指定する。指定がある場合はその番地から、省略した場合は START 命令の次の命令から、実行を開始する。
 また、この命令につけられたラベルは、他のプログラムから入口名として参照できる。

(2)

END	
-----	--

END 命令は、プログラムの終わりを定義する。

(3)

DS	語数
----	----

DS 命令は、指定した語数の領域を確保する。

語数は、10 進定数 (≥ 0) で指定する。語数を 0 とした場合、領域は確保しないが、ラベルは有効である。

(4)

DC	定数 [, 定数] ...
----	---------------

DC 命令は、定数で指定したデータを (連続する) 語に格納する。

定数には、10 進定数、16 進定数、文字定数、アドレス定数の 4 種類がある。

定数の種類	書き方	命令の説明
10 進定数	n	n で指定した 10 進数値を、1 語の 2 進数データとして格納する。ただし、n が -32768 ~ 32767 の範囲にないときは、その下位 16 ビットを格納する。
16 進定数	#h	h は 4 けたの 16 進数 (16 進数字は 0 ~ 9, A ~ F) とする。h で指定した 16 進数値を 1 語の 2 進数データとして格納する ($0000 \leq h \leq FFFF$)。
文字定数	'文字列'	文字列の文字数 (> 0) 分の連続する領域を確保し、最初の文字は第 1 語の下位 8 ビットに、2 番目の文字は第 2 語の下位 8 ビットに、...と順次文字データとして格納する。各語の上位 8 ビットには 0 のビットが入る。文字列には、間隔及び任意の図形文字を書くことができる。ただし、アポストロフィ (') は 2 個続けて書く。
アドレス定数	ラベル	ラベルに対応するアドレスを 1 語の 2 進数データとして格納する。

2.4 マクロ命令

マクロ命令は、あらかじめ定義された命令群とオペランドの情報によって、目的の機能を果たす命令群を生成する (語数は不定)。

(1)

IN	入力領域, 入力文字長領域
----	---------------

IN 命令は、あらかじめ割り当てた入力装置から、1 レコードの文字データを読み込む。

入力領域は、256 語長の作業域のラベルであり、この領域の先頭から、1 文字を 1 語に対応させて順次入力される。レコードの区切り符号 (キーボード入力の復帰符号など) は、格納しない。格納の形式は、DC 命令の文字定数と同じである。入力データが 256 文字に満たない場合、入力領域の残りの部分は実行前のデータを保持する。入力データが 256 文字を超える場合、以降の文字は無視される。

入力文字長領域は、1 語長の領域のラベルであり、入力された文字の長さ (≥ 0) が 2 進数で格納される。ファイルの終わり (end of file) を検出した場合は、-1 が格納される。

IN 命令を実行すると、GR の内容は保存されるが、FR の内容は不定となる。

(2)

OUT	出力領域, 出力文字長領域
-----	---------------

OUT 命令は、あらかじめ割り当てた出力装置に、文字データを、1 レコードとして書き出す。

出力領域は、出力しようとするデータが 1 文字 1 語で格納されている領域のラベルである。格納の形式は、DC 命令の文字定数と同じであるが、上位 8 ビットは、OS が無視するので 0 でなくてもよい。

出力文字長領域は、1 語長の領域のラベルであり、出力しようとする文字の長さ (≥ 0) を 2 進数で格納しておく。

OUT 命令を実行すると、GR の内容は保存されるが、FR の内容は不定となる。

(3)

RPUSH	
-------	--

RPUSH 命令は、GR の内容を、GR1, GR2, …, GR7 の順序でスタックに格納する。

(4)

RPOP	
------	--

RPOP 命令は、スタックの内容を順次取り出し、GR7, GR6, …, GR1 の順序で GR に格納する。

2.5 機械語命令

機械語命令のオペランドは、次の形式で記述する。

r, r1, r2 GR は、記号 GR0 ~ GR7 で指定する。

x 指標レジスタとして用いる GR は、記号 GR1 ~ GR7 で指定する。

adr アドレスは、10 進定数、16 進定数、アドレス定数又はリテラルで指定する。

リテラルは、一つの 10 進定数、16 進定数又は文字定数の前に等号 (=) を付けて記述する。CASLII は、等号の後の定数をオペランドとする DC 命令を生成し、そのアドレスを adr の値とする。

2.6 その他

(1) アセンブラによって生成される命令語や領域の相対位置は、アセンブラ言語での記述順序とする。ただし、リテラルから生成される DC 命令は、END 命令の直前にまとめて配置される。

(2) 生成された命令語、領域は、主記憶上で連続した領域を占める。

3. プログラム実行の手引

3.1 OS

プログラムの実行に関して、次の取決めがある。

(1) アセンブラは、未定義ラベル（オペランド欄に記述されたラベルのうち、そのプログラム内で定義されていないラベル）を、他のプログラムの入口名（START 命令のラベル）と解釈する。この場合、アセンブラはアドレスの決定を保留し、その決定を OS に任せる。OS は、実行に先立って他のプログラムの入口名との関係処理を行いアドレスを決定する（プログラムの関係）。

(2) プログラムは、OS によって起動される。プログラムがロードされる主記憶の領域は不定とするが、プログラム中のラベルに対応するアドレス値は、OS によって実アドレスに補正されるものとする。

(3) プログラムの起動時に、OS はプログラム用に十分な容量のスタック領域を確保し、その最後のアドレスに 1 を加算した値を SP に設定する。

(4) OS は、CALL 命令でプログラムに制御を渡す。プログラムを終了し OS に制御を戻すときは、RET 命令を使用する。

(5) IN 命令に対応する入力装置、OUT 命令に対応する出力装置の割当ては、プログラムの実行に先立って利用者が行う。

(6) OS は、入出力装置や媒体による入出力手続の違いを吸収し、システムでの標準の形式及び手続（異常処理を含む）で入出力を行う。したがって、IN、OUT 命令では、入出力装置の違いを意識する必要はない。

3.2 未定義事項

プログラムの実行等に関し、この仕様で定義しない事項は、処理系によるものとする。

参考資料

参考資料は、COMET II の理解を助けるため又は COMET II の処理系作成者に対する便宜のための資料である。したがって、COMET II, CASL II の仕様に影響を与えるものではない。

1. 命令語の構成

命令語の構成は定義しないが、次のような構成を想定する。ここで、OP の数値は 16 進表示で示す。

15 11 7 3 0 15				0 ← ビット番号			
第 1 語		第 2 語		命令語長	命令語とアセンブラとの対応		
主 OP	副 OP	r/r1	x/r2		機械語命令	意味	
0	0	—	—	—	1	NOP	no operation
1	0				2	LD r,adr,x	load
	1				2	ST r,adr,x	store
	2				2	LAD r,adr,x	load address
	4			—	1	LD r1,r2	load
2	0				2	ADDA r,adr,x	add arithmetic
	1				2	SUBA r,adr,x	subtract arithmetic
	2				2	ADDL r,adr,x	add logical
	3				2	SUBL r,adr,x	subtract logical
	4			—	1	ADDA r1,r2	add arithmetic
	5			—	1	SUBA r1,r2	subtract arithmetic
	6			—	1	ADDL r1,r2	add logical
3	0				2	AND r,adr,x	and
	1				2	OR r,adr,x	or
	2				2	XOR r,adr,x	exclusive or
	4			—	1	AND r1,r2	and
	5			—	1	OR r1,r2	or
	6			—	1	XOR r1,r2	exclusive or
4	0				2	CPA r,adr,x	compare arithmetic
	1				2	CPL r,adr,x	compare logical
	4			—	1	CPA r1,r2	compare arithmetic
	5			—	1	CPL r1,r2	compare logical
5	0				2	SLA r,adr,x	shift left arithmetic
	1				2	SRA r,adr,x	shift right arithmetic
	2				2	SLL r,adr,x	shift left logical
	3				2	SRL r,adr,x	shift right logical
6	1	—			2	JMI adr,x	jump on minus
	2	—			2	JNZ adr,x	jump on non zero
	3	—			2	JZE adr,x	jump on zero
	4	—			2	JUMP adr,x	unconditional jump
	5	—			2	JPL adr,x	jump on plus
	6	—			2	JOV adr,x	jump on overflow
7	0	—			2	PUSH adr,x	push
	1		—	—	1	POP r	pop
8	0	—			2	CALL adr,x	call subroutine
	1	—	—	—	1	RET	return from subroutine
9 ~ E						その他の命令	
F	0	—			2	SVC adr,x	supervisor call

2. マクロ命令

マクロ命令が生成する命令群は定義しない（語数不定）が、次の例のような命令群を生成することを想定する。

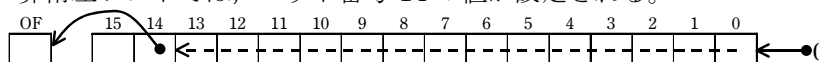
〔例〕 IN 命令

LABEL	IN	IBUF, LEN
	↓	マクロ生成
LABEL	PUSH	0, GR1
	PUSH	0, GR2
	LAD	GR1, IBUF
	LAD	GR2, LEN
	SVC	1
	POP	GR2
	POP	GR1

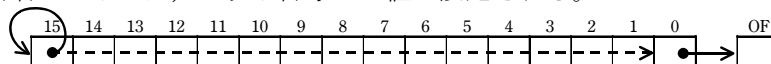
3. シフト演算命令におけるビットの動き

シフト演算命令において、例えば、1ビットのシフトをしたときの動き及び OF の変化は、次のとおりである。

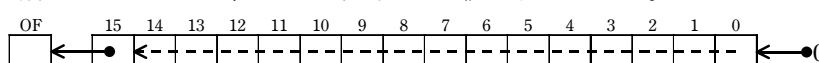
- (1) 算術左シフトでは、ビット番号 14 の値が設定される。



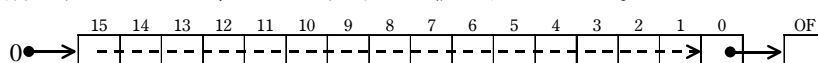
- (2) 算術右シフトでは、ビット番号 0 の値が設定される。



- (3) 論理左シフトでは、ビット番号 15 の値が設定される。



- (4) 論理右シフトでは、ビット番号 0 の値が設定される。



4. プログラムの例

```

COUNT1  START                ;
;      入力      GR1:検索する語
;      処理      GR1 中の'1'のビットの個数を求める
;      出力      GR0:GR1 中の'1'のビットの個数
            PUSH      0,GR1      ;
            PUSH      0,GR2      ;
            SUBA     GR2,GR2      ; Count = 0
            AND      GR1,GR1      ; 全部のビットが'0'?
            JZE     RETURN      ; 全部のビットが'0'なら終了
MORE      LAD      GR2,1,GR2      ; Count = Count + 1
            LAD      GR0,-1,GR1    ; 最下位の'1'のビット1個を
            AND      GR1,GR0      ; '0'に変える
            JNZ     MORE          ; '1'のビットが残っていれば繰り返し
RETURN    LD       GR0,GR2      ; GR0 = Count
            POP      GR2        ;
            POP      GR1        ;
            RET      ; 呼出しプログラムへ戻る
            END      ;

```

■表計算ソフトの機能・用語

表計算ソフトの機能、用語などは、原則として次による。

1. ワークシート

表計算ソフトの作業領域をワークシートという。ワークシートの大きさは 256 列（列 A から列 Z，列 AA から列 AZ，さらに列 BA から列 BZ と続き，列 IV まで続く），10,000 行（行 1 から行 10,000 まで）とする。

2. セル

- (1) ワークシートを縦・横に分割したときの一つのます目をセルという。列 A 行 1 のセルは A1 と表す。
- (2) 長方形の形をしたセルの集まりを範囲として指定することができる。範囲の指定は A1 ～ B3 のように表す。
- (3) 範囲に名前を付けることができる。範囲名は [] を用いて，“セル A1 ～ B3 に [金額] と名前を付ける”などと表す。
- (4) データが入力されていないセルを，空白セルという。

3. セルへの入力

- (1) セルに数値，文字列，計算式を入力できる。
- (2) セルを保護すると，そのセルへの入力を不可能にすることができる。セルの保護を解除すると，そのセルへの入力が再び可能になる。
- (3) セル A1 に数値 5 を入力するときは，“セル A1 に 5 を入力”と表す。
- (4) セル B2 に，文字列 ABC を入力するときは，“セル B2 に 'ABC' を入力”と表す。
- (5) セル C3 に，セル A1 とセル B2 の和を求める計算式を入力するときは，“セル C3 に計算式 A1+B2 を入力”などと表す。

4. セルの内容の表示

- (1) セルに数値を入力すると，右詰めで表示される。
- (2) セルに文字列を入力すると，左詰めで表示される。
- (3) セルに計算式を入力すると，計算結果が数値ならば右詰めで，文字列ならば左詰めで表示される。
- (4) セルの内容の表示については，左詰め，中央揃え^{そろ}，右詰めに変更できる。

5. 計算式

- (1) 計算式には，数学で用いられる数式が利用できる。
- (2) 計算式で使用する算術演算子は，“+”（加算），“-”（減算），“*”（乗算），“/”（除算）及び“^”（べき算）とする。
- (3) 算術演算子による計算の優先順位は，数学での優先順位と同じである。

6. 再計算

- (1) セルに計算式を入力すると，直ちに計算結果を表示する。
- (2) セルの数値が変化すると，そのセルを参照しているセルも自動的に再計算される。この再計算は A1, A2, A3, …, B1, B2, B3, … の順に 1 回だけ行われる。

7. 関数

- (1) 計算式には次の表で定義する関数を利用することができる。

関数名と使用例	解 説
合計 (A1 ~ A5)	セル A1 からセル A5 までの範囲のすべての数値の合計を求める。
平均 (B2 ~ F2)	セル B2 からセル F2 までの範囲のすべての数値の平均を求める。
平方根 (I6)	セル I6 の値 (正の数値でなければならない) の正の平方根を求める。
標準偏差 (D5 ~ D19)	セル D5 からセル D19 までの範囲のすべての数値の標準偏差を求める。
最大 (C3 ~ E7)	セル C3 からセル E7 までの範囲のすべての数値のうちの最大値を求める。
最小 ([得点])	[得点] と名前を付けた範囲のすべての数値のうちの最小値を求める。
IF (B3 > A4, '北海道', '九州')	第 1 引数に指定された論理式が真 (成立する) ならば第 2 引数が, 偽 (成立しない) ならば第 3 引数が求める値となる。左の例では, セル B3 が A4 より大きければ文字列 '北海道' が, それ以外の場合には文字列 '九州' が求める値となる。論理式中では, 比較演算子として, =, ≠, >, <, ≤, ≥ を利用することができる。第 2 引数, 第 3 引数に, 更に IF 関数を利用して, IF 関数を入れ子にすることができる。
個数 (G1 ~ G5)	セル G1 から G5 までの範囲のうち, 空白セルでないセルの個数を求める。
条件付個数 (H5 ~ H9, '>25')	第 1 引数に指定された範囲のうち, 第 2 引数に指定された条件を満たすセルの個数を求める。左の例では, セル H5 から H9 までの範囲のうち, 値として 25 より大きな数値を格納しているセルの個数を求める。
整数部 (A3)	セル A3 の値 (数値でなければならない) を超えない最大の整数を求める。 例えば, 整数部 (3.9) = 3 整数部 (-3.9) = -4 となる。
剰余 (C4, D4)	セル C4 の値を被除数, D4 の値を除数とし, 被除数を除数で割ったときの剰余を求める。剰余の値は常に除数と同じ符号をもつ。“剰余”関数と“整数部”関数は, 次の関係を満たしている。 剰余 (x, y) = x - y * 整数部 (x/y)
論理積 (論理式 1, 論理式 2, ...)	引数として指定された論理式がすべて真であれば, 真を返す。引数のうち一つでも偽のものがあれば, 偽を返す。引数として指定できる論理式の数は任意である。
論理和 (論理式 1, 論理式 2, ...)	引数として指定された論理式がすべて偽であれば, 偽を返す。引数のうち一つでも真のものがあれば, 真を返す。引数として指定できる論理式の数は任意である。
否定 (論理式)	引数として指定された論理式が真であれば偽を, 偽であれば真を返す。
注	“合計”, “平均”, “標準偏差”, “最大”, “最小” は, 引数で指定された範囲のセルのうち, 値として数値以外を格納しているものは無視する。

(2) 関数の引数には, セルを用いた計算式, 範囲, 範囲名, 論理式を指定することができる。

8. セルの複写

- (1) セルに入力された数値, 文字列, 計算式を他のセルに複写することができる。
- (2) セルに入力された計算式が他のセルを参照している場合は, 複写先のセルでは相対的にセルが自動的に変更される。例えば, セル A6 に合計 (A1 ~ A5) を入力した場合, セル A6 をセル B7 に複写すると, セル B7 の計算式は合計 (B2 ~ B6) となる。

9. 絶対参照

- (1) 計算式を複写しても参照したセルが変わらない参照を絶対参照といい, 記号 \$ を用いて \$A \$1 などと表す。例えば, セル B1 に計算式 \$A \$1+5 を入力した場合, セル B1 をセル C4 に複写してもセル C4 の計算式は \$A \$1+5 のままである。
- (2) 絶対参照は行と列の一方だけについても指定可能であり, \$A 1, A \$1 などと表す。例えば, セル D2 に計算式 \$ C1-3 を入力した場合, セル D2 をセル E3 に複写すると, セル E3 の計算式は \$ C2-3 となる。

る。また、セル G3 に計算式 F\$2-3 を入力した場合、セル G3 を H4 に複写すると、セル H4 の計算式は G\$2-3 となる。

10. マクロ

- (1) ワークシートには幾つかのマクロを保存できる。マクロはマクロ P, マクロ Q などと表す。
- (2) マクロについては“マクロ P を実行するとワークシートを保存する。”, “セル A1 からセル A10 までを昇順に並べ替える手続をマクロ Q に登録する。”, “マクロ R : 数値を入力。”, “C 列のデータがその数値以下のものを抽出する。”などと記述する。

11. その他

ワークシートの“保存”, “読出し”, “印刷”や、罫線機能, グラフ化機能など市販されている多くの表計算ソフトに備わっている機能は使用できるものとする。

■ プログラム言語 Perl の用例・解説

Perl を使用した問題では、各問題文中に注記がない限り、次に示す用例に従って記述する。
 なお、用例は、解答で使用する演算子、関数、予約語などを制限するものではない。

種類	用例 ----- 解説
----	-------------------

1. 注釈

#	#ここにコメントを書く ----- 行末までが注釈となる。
---	-------------------------------------

2. リテラル

スカラ	123	----- 10 進数 123 である。
	12.3	----- 10 進数 12.3 である。
	4E-5	----- 10 進数 4×10^{-5} である。
	0x9f	----- 16 進数 9F である。
	0147	----- 8 進数 147 である。
	0b010111	----- 2 進数 010111 である。
	<code>\$var = "hello"; print '\$var ', "\$var ", `echo world`;</code>	----- 変数 <code>var</code> に文字列 "hello" を代入する。文字列のスカラ <code>'\$var '</code> , <code>"\$var "</code> , <code>`echo world`</code> を出力する。 <code>"\$var "</code> は変数を展開し、 <code>`echo world`</code> はコマンドの出力を展開するので、出力は <code>"\$var hello world"</code> となる。
	\n	----- 制御文字 (改行) である。
	\r	----- 制御文字 (復帰) である。
	\t	----- 制御文字 (水平タブ) である。
リストリテラル	('a', 'b', 'c')	----- リスト ('a', 'b', 'c') である。
	('a', 'b', 'c')[0]	----- リスト ('a', 'b', 'c') の 1 番目の要素 'a' である。
	()	----- 空リストである。
	('a' => 'alpha', 'b' => 'bravo', 'c' => 'charlie')	----- キー a, b, c に、それぞれ値 alpha, bravo, charlie を結び付けたハッシュである。

ファイルハンドル	STDIN
	標準入力である。
	STDOUT
	標準出力である。
	STDERR
	標準エラー出力である。
	ARGV
	コマンドラインから指定されたファイル名のリストを順に読み込むためのファイルハンドルである。

3. 変数

スカラー変数	\$var スカラー変数 var である。
配列変数	@ary 配列変数 ary である。
配列要素	\$ary[6] 配列変数 ary の 7 番目の要素である。
ハッシュ変数	%hash ハッシュ変数 hash である。
ハッシュ要素	\$hash{'a'} ハッシュ変数 hash の要素のうち、キー a に結び付けられた値である。
局所的な変数	{my \$var;} { } 内を有効範囲とする変数 var の宣言である。
\$_	\$_ = "abc"; if (/b/) print "match"; パターンマッチの演算子が省略されたとき、 \$_ の文字列 “abc” が // 内のパターン b と一致するかどうかを判定し、“match” が出力される。
@ARGV	@ARGV コマンドライン引数のリストを格納する配列変数である。
@_	@_ サブルーチンに渡す引数のリストを格納する配列変数である。

4. 演算子

->	\$object->method1 オブジェクト object のメソッド method1 を呼び出す。
	Class->method2 クラス Class のメソッド method2 を呼び出す。
++, --	\$a++ 変数 a を評価した後に 1 を加算する。
	--\$b 変数 b から 1 を減算した後に評価する。
!, + (単項), - (単項)	!\$a 変数 a の論理否定である。
	+123 正の数 123 である。
	-123 負の数 123 である。

<code>=~, !~</code>	<code>\$html_contents =~ //</code> ----- 変数 <code>html_contents</code> の値に、文字列 “” が含まれているときに真を返す。 <code>\$html_contents !~ /
/</code> ----- 変数 <code>html_contents</code> の値に、文字列 “ ” が含まれていないときに真を返す。
<code>*, /, %</code>	<code>314 * 34</code> ----- 314 と 34 の乗算である。 <code>6 / 469</code> ----- 6 を 469 で割る除算である。 <code>34 % 6</code> ----- 34 を 6 で割る剰余演算である。
<code>+, -, .</code>	<code>3.14 + 2.72</code> ----- 3.14 と 2.72 の加算である。 <code>220 - 8125</code> ----- 220 から 8125 を引く減算である。 <code>"IPA"."JITEC"</code> ----- 文字列 “IPA” と “JITEC” の連結である。
<code><, >, <=, >=, lt, gt, le, ge</code>	<code>1 < 2</code> ----- 数値 1 と 2 を比較し、演算子の左側が右側より小さいので真を返す。数値の関係演算子には、ほかに <code>></code> , <code><=</code> , <code>>=</code> がある。 <code>"b" lt "a"</code> ----- 文字列 “b” と “a” を比較し、演算子の左側が右側より小さくないので偽を返す。文字列の関係演算子には、ほかに <code>gt</code> , <code>le</code> , <code>ge</code> がある。
<code>==, !=, <=>, eq, ne, cmp</code>	<code>1 <=> 2</code> ----- 数値 1 と 2 を比較し、演算子の左側が右側より大きければ 1, 等しければ 0, 小さければ -1 を返すので、この場合は -1 を返す。数値の比較演算子には、ほかに <code>==</code> , <code>!=</code> がある。 <code>"b" cmp "a"</code> ----- 文字列 “b” と “a” を比較し、演算子の左側が右側より大きければ 1, 等しければ 0, 小さければ -1 を返すので、この場合は 1 を返す。文字列の比較演算子には、ほかに <code>eq</code> , <code>ne</code> がある。
<code>&&</code>	<code>\$x >= 0 && \$x < 10</code> ----- 変数 <code>x</code> の値が 0 以上かつ 10 未満なら真を返す。
<code> </code>	<code>\$x < 0 \$x >= 10</code> ----- 変数 <code>x</code> の値が 0 未満又は 10 以上なら真を返す。
<code>..</code>	<code>@card = (1 .. 52)</code> ----- 1 から 52 までの連続する整数を配列変数 <code>card</code> に代入する。
<code>=, +=, -=, *=, /=, %=</code>	<code>\$a = 1</code> ----- 変数 <code>a</code> に 1 を代入する。 <code>\$a += 10</code> ----- 変数 <code>a</code> の値に 10 を加算して <code>a</code> に代入する。 代入演算子には、ほかに <code>-=</code> , <code>*=</code> , <code>/=</code> , <code>%=</code> がある。
<code>=>, ,</code>	<code>%hash = ('a' => 'alpha', 'b' => 'bravo', 'c' => 'charlie')</code> ----- <code>a</code> に <code>alpha</code> , <code>b</code> に <code>bravo</code> , <code>c</code> に <code>charlie</code> を結び付けたハッシュをハッシュ変数 <code>hash</code> に代入する。
<code>not</code>	<code>not \$a</code> ----- 変数 <code>a</code> の論理否定である。

and	<code>\$a < 0 and \$b == 0</code> ----- 変数 a が 0 より小さいか、変数 b が 0 と等しいかという二つの関係式の論理積である。
or, xor	<code>\$a < 0 or \$b == 0</code> ----- 変数 a が 0 より小さいか、変数 b が 0 と等しいかという二つの関係式の論理和である。 <code>\$a < 0 xor \$b == 0</code> ----- 変数 a が 0 より小さいか、変数 b が 0 と等しいかという二つの関係式の排他的論理和である。

注 演算の優先順位は、上表の枠の順である。

5. 文

if	<pre>if (\$var == 1) { print "a"; } elseif (\$var == 2) { print "b"; } else { print "c"; }</pre> ----- 変数 var の値が 1 なら “a” を、2 なら “b” を、それ以外なら “c” を出力する。
while	<pre>\$i = 1; while(\$i <= 10) { print \$i++, "\n"; }</pre> ----- 変数 i の値を 1 から 1 ずつ増やし、10 回出力する。
for	<pre>for(\$i = 1; \$i <= 10; \$i++){ print "\$i\n"; }</pre> ----- 変数 i の値を 1 から 1 ずつ増やし、10 回出力する。
foreach	<pre>foreach \$i (1, 3, 5) { print "\$i\n"; }</pre> ----- 変数 i にリストの各要素 1, 3, 5 を順に代入し、3 回出力する。
next	<pre>for (\$i = 1; \$i <= 10; \$i++) { next if \$i % 2; print "\$i\n"; }</pre> ----- 変数 i が 2 で割り切れないとき、ループ本体の next 行より後を実行しないので、偶数を出力する。

6. 正規表現

\	<code>/\.\^\\$\[\]\ +*?{\}(\)\ \/\ /</code> ----- 次の 1 文字そのものを表す。“ <code>.\^\$[+*?{}()\ \/</code> ” と一致する。
.	<code>/www.ipa.go.jp/</code> ----- 改行文字以外の任意の 1 文字と一致する。“ <code>wwwdipa,go@jp</code> ” と一致する。
^	<code>/^ab/</code> ----- 先頭が “ab” である文字列と一致する。“abc” と一致するが、“cab” とは一致しない。

\$	/yz\$/ ----- 末尾が“yz”である文字列と一致する。“xyz”と一致するが、“yza”とは一致しない。
+	/go+d/ ----- 直前の 1 文字 o の 1 回以上の繰返しと一致する。“god”や“goood”と一致するが、“gd”とは一致しない。
*	/go*d/ ----- 直前の 1 文字 o の 0 回以上の繰返しと一致する。“gd”、“god”や“goood”と一致する。
?	/colou?r/ ----- 直前の 1 文字 u の 0 回又は 1 回の出現と一致する。“color”又は“colour”と一致する。
{m}, {m,n}	/co{2}l/ ----- 直前の 1 文字 o の 2 回の繰返しと一致する。“cool”と一致するが、“col”や“coool”とは一致しない。 ----- /go{1,3}d/ ----- 直前の 1 文字 o の 1 ~ 3 回の繰返しと一致する。“god”や“good”と一致するが、“gd”や“goood”とは一致しない。
(…)	/ <(h.)>/ ----- () 内の文字列と一致するパターンを部分パターンとしてまとめる。“<h1>”と一致した場合は“h1”が、“<hr>”と一致した場合は“hr”が、まとめられる。
\1, \2, …	/ <(.)><([bp])>JITEC<\/\2><\/\1>/ ----- 左から順に () 内のパターンと一致した文字列が \1, \2, … に割り当てられる。“<h1>JITEC</h1>”と一致するが、“<td>JITEC</p></td>”とは一致しない。
[…]	/ <h[12r]>/ ----- [] 内で指定した文字 1, 2 又は r のどれか一つと一致する。“<h1>”, “<hr>”と一致するが、“<h3>”や“<HR>”とは一致しない。 ----- / [^0-9]/ ----- [] 内で指定した 0 ~ 9 以外の 1 文字と一致する。“a”と一致するが、“3”とは一致しない。
… …	/ <(a href img src)=/ ----- で区切られた“a href”又は“img src”のどちらか一方と一致する。“<a href=”や“<img src=”と一致するが、“<A HREF=”や“<img height=”とは一致しない。

7. サブルーチン

定義	<pre>sub greeting { print "hello Perl\n"; }</pre> ----- “hello Perl”を出力するサブルーチン <code>greeting</code> を定義する。
呼出し	<pre>subroutine (\$arg1, \$arg2);</pre> ----- サブルーチン <code>subroutine</code> を引数 <code>arg1</code> と <code>arg2</code> で呼び出す。() を省略して“ <code>subroutine \$arg1, \$arg2;</code> ”とする表記もある。
戻り	<pre>return -1;</pre> ----- サブルーチンから抜け出し、値 <code>-1</code> を返す。

8. モジュール

use	<pre>use CGI;</pre> <hr/> モジュール CGI を 1 度だけ読み込み, 利用可能にする。
-----	---

9. メソッド呼出し

->	<pre>\$object->method1(arg1);</pre> <hr/> 演算子 -> を使って, オブジェクト object のメソッド method1 を引数 arg1 で実行する。
	<pre>Class->method2(arg1, arg2);</pre> <hr/> 演算子 -> を使って, クラス Class のメソッド method2 を引数 arg1 及び arg2 で実行する。

10. 文字列操作関数

chomp	<pre>chomp @lines;</pre> <hr/> 配列変数 lines の各要素の末尾にある改行文字を削除する。
eval	<pre>eval \$exp_str;</pre> <hr/> 変数 exp_str の内容を Perl プログラムとして解釈し実行する。
length	<pre>length \$long_str;</pre> <hr/> 変数 long_str に格納される文字列の文字数を返す。

11. 配列・ハッシュ操作関数

keys	<pre>%hash = ('a' => 'alpha', 'b' => 'bravo', 'c' => 'charlie'); foreach \$key (keys %hash) { print "\$key\n"; }</pre> <hr/> ハッシュ変数 hash のキーのリストを取り出し, 各キーを出力する。この場合は, “a”, “b”, “c” を順不同に出力する。
shift	<pre>\$next = shift @queue;</pre> <hr/> 配列変数 queue の先頭要素を取り除いて詰め, 取り除いた値を変数 next に代入する。
sort	<pre>@pile = sort @jumble;</pre> <hr/> 配列変数 jumble の値を文字列の大小比較によって昇順に整列し, 配列変数 pile に代入する。
	<pre>@pile = sort {\$b <=> \$a} @jumble;</pre> <hr/> 配列変数 jumble の値を数値の大小比較に従って降順に整列し, 配列変数 pile に代入する。
split	<pre>@fields = split ',', \$csv;</pre> <hr/> 変数 csv の値をコンマで区切って分割したリストを配列変数 fields に代入する。

12. 検索・置換関数

m/…/ 又は /…/	<pre>\$html_contents =~ //i;</pre> <hr/> 変数 html_contents の値が, 文字列 “” 又は “” を含んでいるかどうかを判定する。i は, 大文字, 小文字の区別をしないオプションである。
s/…/…/	<pre>\$html_contents =~ s/
/\n/gi;</pre> <hr/> 変数 html_contents の中の文字列 “ ”, “ ”, “ ” 又は “ ” を改行文字に置換する。g は, 一致したすべての文字列を置換するオプションである。

\$`, \$&, \$', \$1, \$2, ...	<pre>'The date is 1970-01-23.' =~ /([0-9]{4})-([0-9]{2})-([0-9]{2})/;</pre> <pre>print "String before the date: \$\n";</pre> <pre>print "Date: \$&\n";</pre> <pre>print "String after the date: \$\n";</pre> <pre>print "Year: \$1\n", "Month: \$2\n", "Day: \$3\n";</pre> <p>文字列 “The date is 1970-01-23.” に対して、一致した部分の前の文字列、一致した文字列、一致した部分の後ろの文字列をそれぞれ変数 ` , & , ' に代入する。また、 () で囲まれた部分パターンと一致した文字列を、1 番目から順に変数 1, 2, 3 に代入する。これらを利用し、“String before the date: The date is ”, “Date: 1970-01-23”, “String after the date: .”, “Year: 1970”, “Month: 01”, “Day: 23” の 6 行を出力する。</p>
---------------------------------	---

13. 入出力操作関数

open	<pre>open LOG, '>>cgi.log';</pre> <p>ファイル cgi.log を追記モードで開き、ファイルハンドル LOG に対応付ける。</p>
<filehandle>	<pre>\$line = <USER_FILE>;</pre> <p>ファイルハンドル USER_FILE から 1 行を読み込んで変数 line に代入する。</p>
<>	<pre>@records = <>;</pre> <p>標準入力（コマンドライン引数があるときは、コマンドライン引数で指定されたファイル）から順にデータを読み込み、すべての行を配列変数 records に代入する。</p>
print	<pre>print LOG "sync.\n";</pre> <p>ファイルハンドル LOG に対応するファイルに文字列を出力する。</p>
close	<pre>close LOG;</pre> <p>ファイルハンドル LOG に対応するファイルを閉じる。</p>

14. システムインタフェース

die	<pre>open(FILE, 'a_file') or die 'cannot open a_file';</pre> <p>ファイル a_file を開く。開くのに失敗したとき、“cannot open a_file” というメッセージを出力して実行を終了する。</p>
system	<pre>system 'a.out';</pre> <p>コマンド a.out を実行し、コマンドが終了するまで待機する。</p>

<改訂履歴>

初版：平成 12 年 6 月
改訂：平成 13 年 5 月
改訂：平成 14 年 11 月
改訂：平成 16 年 4 月
改訂：平成 16 年 11 月
改訂：平成 17 年 9 月
改訂：平成 17 年 10 月
改訂：平成 18 年 4 月
改訂：平成 18 年 11 月
改訂：平成 20 年 1 月

情報処理技術者試験 出題範囲

発行者 独立行政法人 情報処理推進機構 情報処理技術者試験センター

〒113-8663 東京都文京区本駒込 2-28-8

文京グリーンコート センターオフィス 15 階

電話 03 (5978) 7600 (代表)

F A X 03 (5978) 7610

音声&F A X 情報サービス 03 (5978) 7609

ホームページ <http://www.jitec.jp/>

i モード専用ホームページ <http://i.jitec.jp/>
