

040102プログラミング解説

問01 エ

構造化定理に関する問題である。

構造化定理のプログラムの制御構造には、順次構造、選択構造、繰返し構造がある。順次構造は制御構造を持たない処理が単独または連続して現れる構造である。選択構造は制御条件式の判定結果によって、真か偽のいずれか一方が選択される制御構造である。繰返し構造は同じ処理を繰返し実行する制御構造で、前判定型と後判定型がある。

ア、ウは選択型、イは繰返し型のdo while型、エは繰返し型のwhile型である。求める答えはエとなる。

問02 エ

プログラミング工程の作業内容に関する問題である。

プログラミングでは、プログラム設計段階のモジュール分割の後を受けて、モジュール内の論理設計、コーディング、モジュールテスト計画、モジュールテストが行われる。

アは基本計画、イはプログラム設計、ウは内部設計、エはプログラミングで、求める答えはエとなる。

問03 イ

再帰的プログラムの特徴に関する問題である。

リカーシブ(再帰)は、プログラムの中から自分自身を呼び出すことを言う。自分自身を定義するのに自分自身よりも1次低い集合を用いる。その部分集合はより低次の部分定義を用いて定義することを繰り返して表現する。手続きの内部で再び自分自身を呼び出すことを再帰呼出という。

アはリューザブル、イはリカーシブ、ウはリロケータブル、エはリエントラントである。求める答えはイとなる。

問04 ア

プログラムの性質に関する問題である。問題の対象はリエントラントである。

アのリエントラントは、手続きだけを記憶したプログラム域と実行によって内容が変化していく部分を記憶するデータ域を分離し、複数の実行要求に対して同時に処理することのでき、それぞれの処理要求に対して正しい結果を返すことができるプログラムまたは手続きである。。求める答えはアとなる。

イのリカーシブは、プログラムの記述などで、関数や手続きを、その関数や手続きを使って定義することである。また、自分自身を直接または間接的に呼び出すことを再帰呼出という。

ウのリューザブルは、再ローディングしないで、逐次再利用しても正しい結果を返すことができる特徴を有するプログラムのことである。

エのリロケータブルは、記憶装置を効率的に使用するために、プログラムをメモリ内で移動させることである。プログラムの実行中に行うものを動的再配置、そうでないものを静的再配置と

いう。

問05 イ

プログラムの特性に関する問題である。

再帰プログラムはプログラムの実行中に自分自身を呼び出すことのできるプログラムである。実行中の状態は後入れ先出しで記憶するスタックを用いる。再入可能プログラムは複数のプロセスで同時に使用できるプログラムで、プログラムを可変な複数のデータ部分と不変な手続き部分に分離しておき、各プロセスごとにデータ部分を個別に割り当て、共通部分は同時処理する。逐次再使用可能プログラムは主記憶装置にロードすれば、その後は繰り返して実行できるプログラムで、実行終了前やプログラムの開始時に内容変数の初期化を行う。

アはFIFOは誤りで、LIFOである。

イの手続き部分とデータ部分に分割して、データ部分はプロセスごとに必要であるは正しい。求める答はイである。

ウの逐次使用可能なプログラムは必ずしも再入可能ではない。

エの複数のプロセスを同時実行できるプログラムは必ずしも再帰的ではない。

問06 エ

リエントラントに関する問題である。

リエントラントは、手続きだけを記憶したプログラム域と実行によって内容が変化していく部分を記憶するデータ域を分離し、複数の実行要求に対して同時に処理することができ、それぞれの処理要求に対して正しい結果を返すことができるプログラムまたは手続きである。

アはリューザブル、イはリロケータブル、ウはオーバレイ、エはリエントラントで、求める答えはエとなる。

問07 イ

値呼び出し、参照呼び出しに関する問題である。

値渡しは、プログラムにおける主プログラムと副プログラム間のデータの受渡し方式の一つで、プログラムにデータ(引数)を与えて処理を行わせることである。与えられたデータが変化しても元のデータに影響を与えない。参照渡しは、メイン・プログラムとサブルーチン間でデータを受け渡す方式の一つで、仮引数である変数が実行中に変化すると、自動的に実引数である変数も変化する。多くのプログラム言語では、サブルーチンに対して引数を、引数のアドレスで引き渡す。サブルーチン側でこの引数を操作する場合には、そのアドレスを参照することにより行う。

xは値呼び出しであり、yは参照呼び出しであるから、次のように計算される。

$x = x + y$ では、 $x = 3$ 、 $y = 2$ で、計算後、 $x = 5$ 、 $y = 2$ である。

$y = x + y$ では、 $x = 5$ 、 $y = 2$ で、計算後、 $x = 5$ 、 $y = 7$ で、サブプログラムからのリターン値はないため、メインプログラムのa、bの値は、aは変化しない、bは参照呼び出しのためyが求まった段階で修正されて、 $b = 7$ となる。

従って、 $a = 3$ 、 $b = 7$ となり、求める答えはイとなる。

問08 ア

関数呼び出し時の引数の性質に関する問題である。

引数はプログラムが関数を呼び出すとき、関数で引用する呼び出し側の情報である。呼び出し側と関数は引数を使用して情報の受け渡しを行う。呼び出し側で渡す情報を実引数、関数側で受け取る情報を仮引数という。

値渡しの場合、実引数の値は異なるメモリ領域に複写される。複写された変数は呼び出された関数の中だけで使用され、関数外には反映されない。参照渡しの場合、関数の引数にポインタを渡す。関数に渡されたポインタは元のメモリ領域を参照するので、呼び出された関数内から直接アクセスすることができる。従って、関数内で変数の値が変更されると元の値も直接変更されることになる。

サブルーチンの実行後に変数の値が変更されないのは値渡しの場合である。求める答えはアとなる。

問09 ウ

オーバーレイに関する問題である。

オーバーレイ方式は、主記憶容量より大きなプログラムを実行できるようにする方式で、プログラムをセグメントという単位に分割して補助記憶装置に格納し、実行時には実行に必要なセグメントをその都度補助記憶装置から主記憶装置へローディングする方式である。セグメント大きさの決定、主記憶装置への割当、補助記憶装置から主記憶装置へのローディングの指示はプログラム内で決定する。主記憶装置と補助記憶装置との間の情報転送はプログラムが分割されたセグメント単位に行われる。オーバーレイでローディングされるセグメントは、同時に主記憶装置上に存在する必要のない排他的なプログラム部分である。共通して使用されるセグメントは主記憶装置内に常駐する。

アのBとC、イのCとD、エのFとGは排他的関係にあり、主記憶に同時に存在しないため参照することはできない。ウのFとAは同時に存在し、参照が可能である。求める答えはウとなる。

問10 エ

流れ図の問題である。

流れ図の内容は最大公約数を求める流れ図である。

ア～エの解答例で、①～⑥を実行する。

アの $a = 2$ b の場合、 $a : b = 2 : 1$ 、 $m = 2$ 、 $n = 1$ として検討する。③は $m > n$ となり、①→②→③→④を実行する。

イの $2 a = b$ の場合、 $a : b = 1 : 2$ 、 $m = 1$ 、 $n = 2$ として検討する。③は $m < n$ となり、⑤を実行、 $m = n$ で印字となる。①→②→③→⑤→②→⑥となる。

ウの $2 a = 3 b$ の場合、 $a : b = 3 : 2$ 、 $m = 3$ 、 $n = 2$ として検討する。③は $m > n$ となり、①→②→③→④を実行する。

エの $3 a = 2 b$ の場合、 $a : b = 2 : 3$ 、 $m = 2$ 、 $n = 3$ として検討する。③は $m < n$ となり、⑤を実行、 $n - m = 3 - 2 = 1 \rightarrow n$ 、③で $m > n$ となり、④を実行、 $m - n = 2 - 1 \rightarrow m$ 、②で $m = n$ となり、印刷となる。実行順は①→②→③→⑤→②→③→④→②→⑥となり、求める答えはエとなる。

問11 イ

Javaの仮想マシンに関する問題である。

アのJITコンパイラは、Javaプログラムのバイトコードを、実行環境の機種に最適化されたネイティブ・コードに変換するコンパイラである。

イの仮想マシンは、JavaVMで、Java言語で作成したソースプログラムは、Javaコンパイラにより、バイトコードと呼ぶ中間コードに変換されて配布され、JavaVMで実行される。JavaVMが組み込まれている環境であれば、コンピュータの機種やOSに関係なく動作可能である。求める答えはイとなる。

ウのクロスコンパイラは、他のコンピュータシステムで動作する機械語を生成するためのコンパイラである。開発環境のない組込用コンピュータのソフトウェア開発に使用される。

エのリバースエンジニアリングは、互換製品を作る目的で、オリジナル製品を解析して設計情報などを調査する手法である。

問12 ウ

コンパイル方式とインタプリタ方式の処理時間に関する問題である。

バイトコードの行数をXとすると、次の式が成り立つ。

$$0.2X / 100 \geq 0.003X / 100 + 0.1X / 100 + 0.15$$

両辺に100を掛けると

$$0.2X \geq 0.003X + 0.1X + 15$$

$$0.097X \geq 15$$

$$X \geq 154.6$$

求める答えはウとなる。

問13 ア

JavaBeansに関する問題である。

アのJavaBeansは、Javaのプログラムで構成されたソフトウェアをアプリケーション部品として取り扱うための規約、仕様である。求める答はアである。

イのJavaScriptは、オブジェクト指向のインターネット用スクリプト言語で、ユーザの操作に合わせてウェブページの表示内容を変えたり、アンケートなどでフォームに入力したデータをチェックするなどHTMLで実現できない機能をウェブページ上で実現する。

ウのJavaアプリケーションは、Javaで作成されたソフトのうち、ユーザのコンピュータに組み込んで、ウェブブラウザと独立して単体で動作させることが可能なアプリケーションである。

エのJavaアプレットは、Javaで作成されたソフトで、HTMLに組み込まれてウェブサーバからダウンロードし、ウェブブラウザ上で実行されるものである。

問14 ア

アプレットに関する問題である。

アのアプレットは、HTMLに組み込んでWebサーバからダウンロードして、webブラウザ上で実行するJavaで作成されたソフトウェアである。求める答えはアとなる。

イのサーバレットは、サーバ上で動作するJavaアプレットで、Webサーバ上でデータベース

の検索などクライアントに対してさまざまなサービスを提供できる。

ウのスク립トは、動きのあるWebページを作成できるインターネット用のスク립ト言語である。HTMLファイルに直接コードを貼り込んでブラウザで実行する。

エのスレッドは、インターネット上の掲示板やネットニュース、パソコン通信の会議室で用いる、ある話題に対する一連のメッセージ群を指す。

問15 イ

Javaアプリケーションに関する問題である。

アのJavaScriptは、オブジェクト指向のインターネット用スク립ト言語で、ユーザの操作に合わせてウェブページの表示内容を変えたり、アンケートなどでフォームに入力したデータをチェックするなどHTMLで実現できない機能をウェブページ上で実現する。

イのJavaアプリケーションは、Javaで作成されたソフトのうち、ユーザのコンピュータに組み込んで、ウェブブラウザと独立して単体で動作させることが可能なアプリケーションである。VMが稼働しておれば動作する。動作しているコンピュータのファイルの読み込み、書き換えは可能である。求める答えはイとなる。

エのJavaアプレットは、Javaで作成されたソフトで、HTMLに組み込まれてウェブサーバからダウンロードし、ウェブブラウザ上で実行されるものである。ユーザのコンピュータのファイルの読み書きや他のアプリケーションを起動できない。

ウのJavaサーバレットは、サーバ上で動作するJavaアプレットで、Webサーバ上でデータベースの検索などクライアントに対してさまざまなサービスを提供できる。CGIに比べてWebサーバのパフォーマンスが落ちないメリットがある。

問16 エ

オープンソースソフトウェアに関する問題である。

アの特定の業界向けに作成されたものは、その業界に限定できるという条件は、自由な再頒布ができること、適用領域に基づいた差別をしないこと、特定製品に依存しないことなどのライセンス要件から考えて適切でない。

イの改変再配布時の同じライセンス適用の問題は、採用しているライセンスとそうでないライセンスがある。

ウの第三者が製品として再配布する場合も、開発者は第三者にライセンス費を請求することができない。

エの社内で改変して再配布しない場合には、改変部分のソースコードを公開しなくてもよい。求める答えはエとな。

問17 ア

オープンソースソフトウェアに関する問題である。

アのオープンソースソフトウェアはソースコードをインターネットなどを通じて無償で公開し、誰でもそのソフトウェアの改良、再配布が行なえるようにしたソフトウェアである。求める答えはアである。

イのコンポーネントウェアは標準仕様のソフトウェア部品を使用してアプリケーションを開発

する手法または開発ツールである。

ウのシェアウェアは継続使用に対して開発者がユーザに対価を要求するソフトウェアである。

エのミドルウェアは基本ソフトウェアとアプリケーションソフトウェアの間に位置して、複数のアプリケーションソフトウェア間の連携をとったり、オペレーティングシステムの違いを吸収したりする役割を行う。

問18 エ

Ajaxに関する問題である。

Ajaxは、JavaScriptとXMLという標準的な技術の組合せでユーザインタフェースを実現する技術である。バックグラウンドでXML形式のデータを非同期にサーバとやり取りを行うことで、ページを遷移せずに動的にページの一部を書き換えることができる。

アはパケットフィルタリング、イはCSS、ウはMIME、エはAjaxである。求める答えはエとなる。

問19 ウ

SGMLに関する問題である。

SGMLは文書の論理構造、意味構造を簡単なマークで記述する言語である。文書中に出てくるタイトル、人名など構造上あるいは意味上重要な部分に、あらかじめ決めてあるマークで印をつける。後で、その部分だけを抜き出して抄録にまとめたり、人名索引を作るのに利用する。

VRMLは3次元グラフィックスデータの記述言語で、VRML97はISO規格になっている。

PostScript言語はページプリンタやディスプレイに出力するテキスト、図形、イメージのページ表記を記述するための言語である。

アはVRML、イはHTML、ウはSGML、エはPostScript言語である。求める答えはウとなる。

問20 イ

XMLの特徴に関する問題である。

XMLは、Webページをつくるための記述言語で、決められたタグを使用するHTMLと異なり、ユーザが自由に独自のタグを定義して使えるのが特徴である。タグを手がかりにブラウザ側で加工がしやすくなるため電子商取引などで使用されている。同じく独自のタグを指定可能なSGMLのサブセットとして考案され、任意のデータをHTMLと同様の感覚で送受信できることを目標に作成されたものである。XMLはその性質上、他のマークアップ言語の骨組みとして使用されることが多い。

アのHTMLとの対応は、表示性能の向上ではなく、表現の自由度の向上である。

イの情報システム間のデータの交換は任意のデータをHTMLと同様に容易に送受信できる。正しい記述である。求める答えはイとなる。

ウのスタイル表現は、SGMLの文法を簡素化した文書であり、SGMLよりは習得しやすく、HTMLよりは柔軟性があり、SGMLとHTMLの間に位置する文書構造である。

エの開発の基はHTMLと同じSGMLである。

問21 ア

ソフトウェアの統合開発環境に関する問題である。

統合開発環境は、エディタ、コンパイラ、デバッガなど、プログラミングに必要なツールが一つのインターフェースで統合して扱えるような環境のことである。

アのEclipseは、IBMによって開発された統合開発環境(IDE)の一つである。高機能ながらオープンソースであり、Javaをはじめとするいくつかの言語に対応している。Eclipse自体はJavaで記述されている。求める答えはアとなる。

イのGCCは、GNUのコンパイラ群である。

ウのLinuxは、UNIXライクなOSカーネルのひとつである。一般には、カーネルにLinuxを採用したOS環境全体を指して、Linuxと呼ばれていることが多い。

エのTOMCATは、Jakartaプロジェクトのサブプロジェクトとして開発されているオープンソースのソフトウェアで、Javaサーブレット・JSPを処理するアプリケーションサーバである。

問22 ウ

言語プロセッサに関する問題である。

アセンブラは、アセンブリ言語で記述したソースプログラムを、CPUが直接実行できる機械語に翻訳するソフトウェアである。インタプリタは、ソースプログラムを1行ずつ解釈して機械語に翻訳しながら実行するソフトウェアで、プログラムの命令を逐次翻訳して実行していくため、実行速度は遅いが、文法エラーやプログラムの修正を対話的に行える特徴がある。トランスレータは、翻訳プログラムの総称として用いられる場合とある処理系用に書かれた原始プログラムを他の処理系用のプログラムに変換する意味に用いられる場合がある。シミュレータは、ある計算機で実行可能な目的プログラムの命令を1つずつ解釈し、他の計算機の命令に変換して実行する場合に利用する。ジェネレータは、条件をパラメータで記述し、問題の処理目的に応じたプログラムを自動的に生成することである。

アはトランスレータの内容であり、アセンブラではない。

イはシミュレータであり、インタプリタではない。

ウのジェネレータの記述のみが正しい内容である。求める答えはウとなる。

エはインタプリタに関する記述で、トランスレータではない。

問23 ア

言語プロセッサ、サービスプログラムに関する問題である。

アのコンパイラは、プログラミング言語で記述されたプログラムを、コンピュータが理解できる機械語に一括変換するソフトウェアである。

イのプリコンパイラは、ソースプログラムの中のマクロ機能などを前処理して、コンパイラに対する単純化したソースプログラムモジュールを作成する。

ウのリンケージエディタは、コンパイラやアセンブラで出力された複数本のオブジェクトモジュールを結合して1本のプログラムモジュールを作成する。

エのローダは、ロードモジュールを再配置処理を行った上で、主メモリ上でロードするプログラムである。

リンケージエディタは、コンパイルに別々に0番地の開始アドレスから生成されていたオブジ

ェクトモジュールにアドレスが連続するように変換する。外部関数を未解決アドレスとして次のステップに渡すのであるから、外部関数をコンパイラから次のステップであるリンケージエディタに渡すことになる。従って、求める答えはコンパイラになり、求める答えはアとなる。

問24 エ

コンパイラの処理順序に関する問題である。

コンパイラの処理順序

① 字句解析

プログラムの文字列を、字句またはトークンの生成規則に従って、文法上の意味のある語、構文を構成する最小単位の字句に切り出すことである。

② 構文解析

字句の連なりとして抽出したものに、文法に照らし合わせて合法的な文であるかどうかを判断することである。文の種類には、宣言文、代入文、構造文などがある。

③ 意味解析

変数名の未定義や二重定義、型の不一致など構文に含まれない制約を検査することである。文字型として宣言された変数が算術式のなかに現れれば、意味上の誤りとして意味解析の段階に検出される。

④ 最適化

実行時間や大きさが小さくなるように、目的プログラムを変更することである。

⑤ 目的コードの生成

手続き型言語のコンパイラの手順は、次の通りである。

字句解析→構文解析→意味解析→最適化→コード生成

の順に進められる。求める答えはエとなる。

問25 ア

四つ組形式に関する問題である。

($*$ 、 B 、 C 、 T_1)は $T_1 = B * C$ 、($/$ 、 T_1 、 D 、 T_2)は $T_2 = T_1 / D$ 、($+$ 、 A 、 T_2 、 T_3)は $T_3 = A + T_2$ となる。

$T_3 = A + T_1 / D = A + B * C / D$ となる。求める答えはアとなる。

問26 ア

コンパイラの処理における最適化の問題である。

最適化とは、実行速度を速くしたり、小さなオブジェクトプログラムを作成するためにコンパイル時に行われる手段の一つで、実行速度を速める場合には、定数の計算を事前に行ったり、ループ内で変化しない演算式をループの外に出したりして最適化を行う。

最適化の目的はプログラムの実行時間の短縮である。求める答えはアとなる。

問27 エ

プリコンパイラに関する問題である。

プリコンパイラはコンパイルに先立って、コンパイラ言語に付加された特殊な機能を通常のコ

ンパイラ言語に変換する。

アはトランスレータ、イはクロスコンパイラ、ウはコンパイラ、エはプリコンパイラで、求める答えはエとなる。

問28 ウ

動的リンクライブラリに関する問題である。

アのコンパイラによる組み込みは、埋め込み方式のSQL文をプリコンパイラに通して、親言語のCALL文に変換し、そのソースプログラムを親言語のコンパイラに通して、CALL文とモジュールを結合する場合などに利用される。

イのプリコンパイラによって生成されるのは、ソースプログラムの中のマクロ機能などの前処理に用いられる。

ウは動的リンク、エは静的リンクである。求める答えはウとなる。

問29 エ

動的リンクに関する問題である。

アのインタプリタは、ソースプログラムを1行ずつ解釈して機械語に翻訳しながら実行するソフトウェアで、プログラムの命令を逐次翻訳して実行していくため、実行速度は遅いが、文法エラーやプログラムの修正を対話的に行える特徴がある。

イのオーバレイは、主記憶装置に読み込めない大きなプログラムを実行するとき複数のセグメントに分割して実行する方法である。常駐部のルートセグメントが、必要に応じて複数個に分割された排他関係にあるセグメントを交互に読み込んで実行する。

ウの静的リンクは、コンパイラやアセンブラで出力された複数本のオブジェクトモジュールをリンケージエディタで結合して1本のプログラムモジュールを作成することである。

エの動的リンクは、プログラムの実行中に別のプログラム・モジュールの機能が必要になったとき、そのプログラムをその場で結合して利用することである。求める答えはエとなる。

問30 ア

プログラミングツールに関する問題である。

アのインスペクタは、プログラムを実行してエラー検出やデータ構造の内容を確認するためのデバッグツールで、プログラムを途中で中断し、トレース対象データの閲覧や更新を対話形式で処理する。求める答えはアとなる。

イのトレーサは、プログラムの実行時に制御の流れを追跡するときや実行過程を時系列的に1ステップずつモニタリングするとき使用する。

ウのシミュレータは、ある計算機で実行可能な目的プログラムの命令を1つずつ解釈し、他の計算機の命令に変換するプログラムである。

エのブラウザは、情報を閲覧するためのソフトウェアである。

問31 ア

シミュレータに関する問題である。

アのシミュレータは、ある計算機で実行可能な目的プログラムの命令を1つずつ解釈し、他

の計算機の命令に変換して実行する場合に利用するプログラムである。シミュレータの実行時は、解釈ルーチンが主記憶に常駐するので領域を多くとり処理効率が悪くなり、処理速度も遅い。求める答えはアとなる。

イのトランスレータは、ある機種用に開発された原始プログラムを他の機種用に変換するための言語で、機種切り換え時に利用する。

ウのジェネレータは、非手続き型言語であるRPGなどで作成されたプログラムを、機械語である目的プログラムに変換する翻訳処理プログラムである。

エのクロスコンパイラは、あるコンピュータ上で高水準言語で書かれたプログラムを、そのコンピュータとは別のコンピュータのオブジェクトプログラムに翻訳するプログラムである。

問32 イ

プログラムの静的解析ツールに関する問題である。

静的解析ツールは、プログラムを実行させずにプログラムテストを実行するソフトウェアである。ソースコード解析ツール、プログラム構造解析ツール、モジュールインタフェースチェックツールなどがある。ソースコード解析ツールはプログラムの読解性を高め、プログラムの構造解析ツールはプログラムの階層構造などを明らかにする。

静的解析ツールは、プログラムに対して、機械的にチェックを行うことにより、文法スタイルの誤りやパターン化されたバグを検出する。

動的解析ツールは、対象となるプログラムを実際に動かして解析を行うプログラムテスト支援ツールである。モニタリングツール、オンラインシミュレータ、システムロギング、自動テストデータ生成・実行ツールなどがある。

アの関数ごとの実行処理時間、エの実行されなかった命令の検出は動的解析ツールで可能であるが静的解析ツールではできない。

イの後で使用されない変数への代入はパターン化が可能であり、静的解析ツールで検出できる。求める答えはイとなる。

ウのプログラム仕様と対応する処理の記述漏れは静的ツールも動的ツールの不可能である。

問33 エ

テストの考え方に関する問題である。

テストは、完成したシステムやモジュールが設計仕様書どおり正しく動作することを確認するために、ソフトウェアの検証度を確認するための一連の作業である。開発者側と利用者側がそれぞれの観点からテストを実施し、システムやモジュールに含まれる欠陥を発見することがテストの目的になる。そのために、テストの方法を明確にし、テスト対象のプログラムの特性を吟味して、適切なテスト技法、テストツール、テストデータを選択・設計する。

テストを行う際の考え方は、可能な限りエラーを見つけることができるようにテストデータを作成することである。求める答えはエとなる。

問34 ア

単体テストに関する問題である。

アのトップダウンテストでは下位モジュールの代わりにするスタブが必要であり、アの記述は

正しい。求める答えはアとなる。

イの入力条件に関するテストはモジュールの機能仕様に基づいて、すべての入力条件、出力条件、エラー処理等、モジュールの持つ機能が満足しているかどうかを検証する。従って、誤った条件の入力ケースを実施する必要がないという記述は誤りである。

ウは、プログラム内部の論理の検証が必要である。コンパイルでエラーが発生しないだけでは論理上十分であるとはいえない。

エは、コーディングが完了していないとコンパイルできないため単体テストができない。

問35 エ

ブラックボックステストに関する問題である。

アのトップダウンテストは、上位のモジュールから下位のモジュールへと順次結合して行う結合テストである。

イのボトムアップテストは、下位のモジュールから上位のモジュールへと順次結合して行う結合テストである。

ウのホワイトボックステストは、モジュールの制御構造を詳細に検討するテストである。

エのブラックボックステストは、モジュールをブラックボックスと見なし、機能仕様書に基づき作成したテストデータでテストを行う手法である。求める答えはエである。

問36 ウ

限界値分析におけるテストデータに関する問題である。

限界値分析は、ブラックボックステスト法の一手法であり、入力データと出力データを同値クラスに分割し、それぞれのクラスの端がテストの対象になるように値を選ぶ方法である。通常は有効同値クラスの最大・最小とそれぞれを一つ超えた値を用いる。従って、最小値の直前の値、最小値、最大値、最大値の直後の値となる。求める答えはウとなる。

問37 イ

ブラックボックス法のテストケースに関する問題である。

ブラックボックス法のテストケースには同値分割や限界値分析などのテストケースがある。

アの条件網羅は、プログラムのすべての判定条件で、真および偽のすべての組合せを満たすようにテストケースを設計する。判定中の条件のすべての可能な結果が、少なくとも1回は実行させる基準である。

イの限界値分析は、ブラックボックステスト法の一手法であり、入力データと出力データを同値クラスに分割し、それぞれのクラスの端がテストの対象になるように値を選ぶ方法である。

ウの命令網羅は、プログラムの各モジュールの命令文を少なくとも1回以上実行するようにテストケースを設計する。

エの判定条件網羅は、プログラムのすべての判定条件で、真と偽を少なくとも1回以上実行するようにテストケースを設計する。

アの条件網羅、ウの命令網羅、エの判定条件網羅はホワイトボックス法のテストデータ作成法であり、イの限界値分析はブラックボックス法のものであり、求める答えはイとなる。

問38 ウ

同値分割法に関する問題である。

同値分割法は、テスト対象の入力データの取り得る値の範囲の中から、同じ意味を持つ範囲を1つのクラスとして、いくつかのクラスに分割する。分割したクラスの中から、各クラスを代表する値をテストデータとして選択する。

無効同値クラス -2 ~ 0、6 ~ 8の中からそれぞれ1つずつ選択し、有効同値クラス 1 ~ 5の中から1つ選択しテストデータとする。-1、3、6のテストデータの組み合わせが適切である。求める答えがウとなる。

問39 ウ

ホワイトボックステストに関する問題である。

判定条件網羅は、プログラムの全ての判定条件で、真と偽を少なくとも1回以上実行するようにテストケースを設計する。

アの場合は真の場合のテストのみである。

イの場合は、Aが偽Bが真の場合も、Bが偽Aが真の場合も共にORは真であるから、真の場合のテストのみである。

ウの場合は、Aが偽Bが偽の場合はORは偽、Aが真Bが真の場合はORは真であるから、真と偽を少なくとも1回実行していることになる。真、偽を少なくとも1回行うテストの判定条件網羅はウとなる。求める答えはウとなる。

エの場合は3回のテストは全て真の場合である。

問40 ウ

ホワイトボックステストのテストケースに関する問題である。

ホワイトボックステストは、プログラムのテストデータ選択方法の1つで、プログラムのモジュールの実行経路を詳細に確認するためにテストデータを選択する。すべての命令をテストすることは難しく、要求されている機能の確認も見出しにくい問題がある。テストデータの作成方法に、命令網羅、判定条件網羅、条件網羅、複数条件網羅等がある。

ア、イ、エはブラックボックス法のテストケース作成方法で、ウの条件網羅がホワイトボックス法のテストケース作成方法である。求める答えはウとなる。

問41 エ

テストの目的に関する問題である。

テストの目的は、作成されたソフトウェア製品の品質を評価することである。開発したシステムやモジュールが設計仕様書どおりに動作するかなど、ソフトウェアの完成度を検証する一連の作業である。開発者とユーザがそれぞれの観点からテストを実施し、システムやモジュールの潜在的な欠陥を発見することである。

テスト時の主要評価項目

- ① 信頼性は要求仕様を満足し、ソフトウェアにエラーがなく、規則違反の使用に対して堅固であることである。
- ② 操作性は入力しやすいこと、入出力情報やメッセージの内容が読みやすく、理解しやすい

いことである。

③ 性能はソフトウェアの処理能力や所要記憶容量が目標値であることである。

④ 保守性は修正・改造が容易であり、拡張性があることである。

テストケースには、有効なテスト条件、システムに求められている機能、その機能の範囲外、エラーケースも含める。データの有効範囲と無効範囲の境界に関するエラーが多いので、境界に関するテストケースを必ず含める。

アは、内部構造のテストはホワイトボックス法、外部仕様のテストはブラックボックス法を用いる。

イは、見つかったエラーの個数が多いほど、プログラムに残っているエラーの個数も多い。関係がある。

ウのテストの目的はプログラムの完全性を証明することではない。仕様どおりに動作するか、エラー発生時の処理が十分か、操作性、性能、完成度などを検証する。

エの意図しなかった動きも調べる必要があるは正しい記述である。求める答えはエとなる。

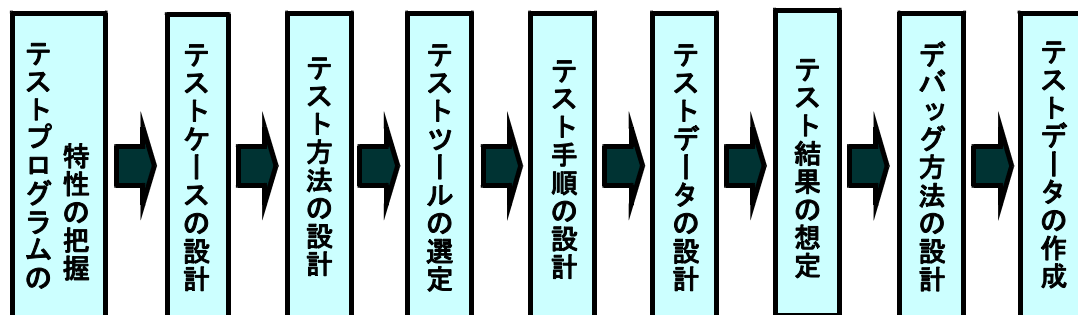
問42 エ

モジュールテストに関する問題である。

テストケースには、有効なテスト条件、システムに求められている機能、その機能の範囲外、エラーケースも含める。データの有効範囲と無効範囲の境界に関するエラーが多いので、境界に関するテストケースを必ず含める。

テストケースを効率よくテストするために必要なテストツールを選定する。テストドライバ、テストデータジェネレータ、スタブ、デバッグ等を選定する。

テストケースを吟味して、テスト実行の手順を決める。テストの手順によって、効率が大きく異なる。結合テスト、システムテスト、運用テストの場合には熟慮が必要である。



テストの目的を明瞭にし、最も効率よく実行できるテストデータを設計する。少ないデータで効率よくテストするためには、テストデータジェネレータで作成する方法、テストチームが作成する方法、ユーザが作成する方法のいずれかの方法を選択することが望ましい。テストの目的に合わせたテストデータの作り方を選択する。テスト設計はモジュール設計段階に図の手順で行う。

アのテストケースの作成は、モジュール設計段階に行うため、プログラマが行う。テストの内容によってユーザが参画することがある。テストデータはテストチームやユーザが作成する。

イのモジュール間インタフェースは、スタブまたはドライバを使用して、行うことができる。ただし、スタブまたはドライバを使用するので、代替モジュールの結果が適切とはいえない。

ウのモジュール設計書が正しくて、テスト結果に問題がある場合、テストケースやモジュールに誤りがなくても、テストデータやテストの方法、テストツールなどに問題があると、テスト結果に問題が発生する。

エの原則としてすべてのロジックパスを一度通るテストケースで検証を行う記述は適切である。求める答えはエとなる。

問43 ウ

トレーサに関する問題である。

トレーサは、プログラムの実行時に制御の流れを追跡するときや実行過程を時系列的に1ステップずつモニタリングするときに使用する。あるデータを処理するときのプログラムの動きを命令単位で調べたり、特定区間の命令を実行する毎に、その所在や命令自身、実行直後のレジスタなどの内容を書き出したりする場合に用いる。誤りの箇所が特定できない時に有効である。区間の指定を的確にしないと、出力量が膨大になり、処理時間がかかる。ワンステップずつ追跡調査できる。

デバッグは、プログラムのバグを検出し除去するために使用するテストツールである。プログラムを実行をさせながら流れを追跡するトレーサ機能やプログラムの任意の箇所で中断させるブレイクポイント機能、パフォーマンスの分析、実行の継続、コード部のスキップ、エラーの訂正、変数の表示及びセット、期待される入力の設定、出力の表示等の機能がある。

アはスナップショットダンプ、イはデバッグ、ウがトレーサ、エはシミュレーションである。求める答えはウとなる。

問44 エ

スナップショットダンプに関する問題である。

スナップショットダンプは、プログラムにデバッグ命令を組み込んでおき、デバッグ命令を実行する都度、主記憶装置の一部やレジスタの内容を書き出す。指定した条件のときだけ主記憶やレジスタの内容などの追跡データを出力するダンプである。動的ダンプのときに使われ、プログラムにエラーがあり原因不明で分からないとき、プログラムの特定の要所にダンプの条件を設定しておき、そのプログラムの記憶装置よりメモリの一部または全部の内容を出力して、エラーを調べることをいう。プログラムの誤りの箇所がほぼ見当がついていて、どのような状態で異常が発生しているかを知るため、誤りと思われる命令群の処理前のデータの状態と処理後のデータの状態を比較するときに使用するデバッグツールである。

アのメモリダンプは、ダンプ命令を組み込んで、メモリの特定の領域をリストするツールである。特定の命令が実行される前後にダンプ命令を組み込んでプログラムの実行状況を把握するときに使用する。

イのエディタは、プログラムやデータ、文書などの文字列を編集するプログラムである。図形や絵を作成するエディタに図形エディタがある。

ウのトレーサは、プログラムの実行過程を時系列的に1ステップずつモニタリングする動的デバッグツールである。

エのスナップショットダンプは、プログラムにデバッグ命令を組み込んでおき、デバッグ命

令を実行する都度、指定した特定の命令が実行されるたびに主記憶装置の一部やレジスタの内容を書き出すデバッグツールである。

問45 ア

ポートスキャナーの利用目的に関する問題である。

ポートスキャンは、ネットワークを通じてサーバに連続してアクセスし、保安上の弱点(セキュリティホール)を探す行為で、すべてのポートに信号を送り通信に利用可能なポートを探す。通常、ポートスキャナーの利用目的は、使用したいポートが通信可能であること、あるいは、使用していないポートが通信不能であることをコンピュータやルータの管理者が確認することであり、ネットワーク管理者が、自分の管理するシステムに弱点がないかどうか調べるためにポートスキャンを行う。

アはポートスキャナーによる検査、イは利用者IDの管理状況の確認、ウはアクセスログの解析、エはWebアプリケーション脆弱性診断サービスである。求める答えはアとなる。

問46 ウ

結合テストの目的に関する問題である。

結合テストは、モジュール間のインタフェースとプログラムの機能を検査する。

ア、イ、エはシステムテストである。システムテストでは、インタフェーステスト、機能テスト、性能テスト、障害テストや負荷テストが行われる。

ウは結合テストで、求める答えはウとなる。

問47 イ

トップダウンテストに関する問題である。

トップダウンテストは、最上位のモジュールからモジュールテストを行う方式で、順次、下位モジュールを結合しながらモジュールテストを繰り返す。上位モジュールからテストを行うため、下位モジュールが未完の場合、スタブが必要である。スタブは上位モジュールから制御をもらい、引数を設定してから制御を戻す簡単な機能を持つ。トップダウンテストは、モジュール間インタフェースを仮定する必要が無く、引数の受渡をテストすることができる。最上位モジュールは最初からテストが繰り返されており、バグの発見も効率よく行え、プログラム全体の制御を司る重要な部分がきちんとテストできる。

トップダウンテストの特徴

- ① 最初は並行作業が困難であり、テストするまで時間が空いてしまう可能性がある。
- ② 上位のインタフェースが早い時期にテストできる。
- ③ 重要度の高い上位モジュールが繰返し実行されるので、信頼性が高い。
- ④ モジュール単体の機能や、個々の論理を十分にテストできないことがある。
- ⑤ 一般に、新規開発システムに適用すると効果がある。
- ⑥ トップダウンでモジュールの集積を進めるため、並行した開発がし難い。
- ⑦ 全体としては工数が多くなる。

アの下位モジュールを代行するのはドライバではなく、スタブである。

イの内容はトップダウンテストの特徴を表している。求める答えはイとなる。

ウの最終段階でモジュール間のインタフェースに問題が生じやすいのはボトムアップテストである。

エの開発の初期から並行作業が可能なのはボトムアップテストである。

問48 ア

ボトムアップテストに関する問題である。

ボトムアップテストは、プログラム構造の最下位のレベルのモジュールを最初にテストし、次に、それらの上位レベルのモジュールを結合してテストする。これを繰り返しながら最後に最上位のモジュールを結合してテストを行う。上位モジュールが未完の場合、テストモジュールの上位モジュールの機能をシミュレートするドライバが必要である。ドライバは下位モジュールに制御と引数を渡すとともに、下位モジュールから戻ってきた引数をもとに、対応する操作を実行する。上位モジュールの役割を代行する機能を持つ。インタフェーステストを何回も行う場合、繰り返し下位モジュールを呼び出せる制御構造を持つ必要があり、ドライバの作成が煩雑になる。

ボトムアップテストの特徴

- ① 開発の初期の段階から並行作業が可能である。
- ② モジュール単体の機能や論理が十分にテストできる。
- ③ テストの最終段階でインタフェース上の問題が発生しやすい。
- ④ 全体の機能を把握するのに時間がかかる。
- ⑤ 既に稼働しているシステムを修正して、システムを開発する場合に有効である。
- ⑥ 問題は、モジュール間のインタフェースについてすべてを結合しない限り、全体の整合性をテストできない点にある。
- ⑦ 最上位モジュールが最後に結合されるため、十分にテストできない。

アはボトムアップテスト、イはビックバンテスト、ウはトップダウンテスト、エは通常の開発工程で順次行われる一連のテスト手順である。求める答えはアとなる。

問49 エ

ボトムアップテストにおけるテスト支援ツールであるドライバに関する問題である。

アのインタプリタは、高水準言語で書かれた実行プログラムを1行ずつ解釈しながら実行していくプログラムである。

イのコンパイラは、プログラムをコンピュータに理解できる機械語に変換するプログラムの一種である。

ウのスタブは、トップダウンテストに用いられるツールである。

エのドライバは、ボトムアップテストに用いられるツールである。求める答えはエである。

問50 イ

スタブに関する問題である。

アのエミュレータは、シミュレータの機能のプログラムをファームウェア化したものである。

イのスタブは、トップダウン結合テストにおいて、下位モジュールの代替となるテスト用のモジュールである。求める答えはイである。

ウのデバッガは、プログラムの不具合(バグ)の発見や修正を支援するソフトウェアである。
エのドライバは、ボトムアップテストに用いられるツールである。

問51 ウ

システム開発におけるテスト順序に関する問題である。

単体テストは、最小単位であるモジュールの検証および品質をテストする。モジュールの機能仕様に基づいて、入力条件、出力条件、エラー処理などのモジュールのもつ機能の満足度をテストする機能テストとモジュールの詳細仕様やソースプログラムをもとにモジュールの論理の正しさを検証する構造テストがある。結合テストは、モジュールを結合して行うテストで、モジュール間のインタフェースの検証、プログラムの入出力を含むテストで、モジュールの結合の仕方により、ボトムアップテスト、トップダウンテスト、ビッグバンテストなどがある。システムテストは、ソフトウェアやシステムが、仕様に合致しているかどうかを検証するテストである。プログラム間の結合テスト、サブシステム間の結合テスト、サブシステム、全体システムの入出力を含むテスト、例外事項テスト、障害テスト、処理速度テストなどを行う。

システム開発におけるテスト順序は、単体テスト→結合テスト→システムテストである。求める答えはウである。

問52 ア

システムテストに関する問題である。

システムテストで行う主要テスト内容

- ① プログラム間やサブシステム間の結合テスト
- ② サブシステム・システムの入出力を含むテスト
- ③ 例外事項のテスト
- ④ 障害テスト
- ⑤ 性能テスト(スループット、レスポンスタイム、ターンアラウンドタイム)、負荷テスト
- ⑥ 機能テスト

アはシステムテストで確認する内容、イは結合テストで行う内容、ウはブラックボックステストで確認する内容、エのレグレッションテストはソフトウェアの保守時に行うテストである。求める答えはアとなる。

問53 ウ

運用テストに関する問題である。

運用テストは、完成プログラムをユーザ部門の運用グループが実際の運用と同一の条件を作り出し、機能面および操作面の双方のテストを行う。承認テスト、導入テスト、フィールドテストなどがある。フィールドテストは特定の利用者に実際にシステムを使ってもらった実地テストである。ユーザ部門の責任で行う。求める答えはウとなる。

問54 イ

運用テストに関する問題である。

運用テストはユーザ部門の運用グループが実際の運用と同一の条件を作り出し、機能面および

操作面の双方のテストを行う。承認テスト、導入テスト、フィールドテストなどがある。運用テストでは、本番移行基準の確認、移行テスト、保守性テスト、運用効率や業務効率の測定、ユーザビリティテストなどが行われる。移行テストでは、安全性・効率性の観点で、既存システムから新システムへの切り替え手順、切り替えに伴う問題点を確認する

日常の業務内容を熟知している利用者が優先して確認すべき事項は、決められた業務手順通りにシステムが稼働することである。求める答えはイとなる。

ア、ウ、エの内容は、情報システム部門の運用者が確認すべき事項である。

問55 ウ

退行テスト(レグレッションテスト)に関する問題である。

アの機能テストは、ユーザ部門の要求仕様を満足しているかどうかをテストする。ユーザ部門の要求仕様の機能が詳細に記述されていない場合には、設計書の中から要求仕様を取り出す作業が必要になる。

イの結合テストは、モジュールを結合したテストで、モジュール間のインタフェースの検証とプログラムの入出力を含むテストである。プログラムが外部仕様で定められた機能どおりに、実現されているかを検証する。信頼性や操作性、保守性などの検証も行う。

ウの退行テストは、システムの保守段階に行うテストで、プログラムの変更により、既存の正しい範囲に新しい誤りが発生しないかどうかを検証する。求める答えはウとなる。

エの例外テストは、例外データを与えたときに正しく例外処理を行うことを確認するテストである。

問56 ア

テストデータに関する問題である。

プログラム検査は、作成されたソフトウェア製品の品質を評価するために行うものであり、信頼性、操作性、性能、保守性が主要評価項目になる。

テスト計画の立案は、単体テストはモジュール設計時、結合テストはプログラム設計時、システムテストは外部設計時にそれぞれ行われる。

限られた時間内に効率的にテストを実施するためには、事前にテスト毎にテストの目的、テストの方法を明確にし、テスト技法、テストツールを選択し、テストデータを作成する必要がある。

テストデータの設計・作成には、テストの目的に合わせて、必要なテストケースを洗い出し、有効なテスト条件、システムに求められている機能、その機能の範囲外、エラーケースをも含めたテストの計画が必要である。特に、データの有効範囲と無効範囲の境界に関するエラーの確認は不可欠である。

アの事前にテストケースを設定し、それに沿ったテストデータを準備するは適切な記述である。求める答えはアである。

イの正しく処理するデータのための準備は誤りである。

ウの2割程度のデータ量は状況によって変化するので一律の考え方としては正しくない。

エのエラー処理用のデータも必要に応じて準備しなければならない。

問57 ア

移行テストに関する問題である。

システムの移行は、新システムを稼働させるために、現行システムから新システムへ、ハードウェアやソフトウェア、各種ファイルを円滑に移し変えることである。そのために、移行方法や移行手順、移行体制、移行日程計画、移行タイムチャートなどを作成する。移行作業の中で大きな比重を占めるものがデータの移行である。利用部門と協同して、現行の業務フロー、データベース仕様を元に、現行システムと新システムのデータ項目を比べ、移行対象データ項目を決める。ハードウェア、ソフトウェア、データの移行方法、移行のタイミングも重要な検討項目である。新たなシステム開発に伴って、ハードウェアの入れ替えや増強、OSやミドルウェアの入れ替えや変更、通信回線の新設や増設が行われる。

システムテストはソフトウェアやシステムが要求された仕様に合致しているかどうかを検証するために、複数の機能グループを組み合わせて行うテストである。システムテストの内容は、プログラム間やサブシステム間の結合テスト、サブシステム・システムの入出力を含むテスト、例外事項のテスト、障害テスト、性能テスト(スループット、レスポンスタイム、ターンアラウンドタイム)、負荷テスト、機能テストなどがある。

運用テストとはユーザ部門の運用グループが実際の運用と同一の条件を作り出し、機能面および操作面の双方のテストを行う。承認テスト、導入テスト、フィールドテストなどがある。承認テストは開発部門がユーザ部門の承認を得るテストである。ユーザ部門がデータを用意し、要求仕様どおりの結果が得られるかどうかをユーザ部門に承認してもらうために行う。ユーザ部門が管理、実行する。更に、運用テストの内容として、本番移行基準の確認、移行テスト、保守性テスト、運用効率や業務効率の測定、ユーザビリティテストなどが行われる。

アの安全性・効率性の観点で、既存システムから新システムへの切り替え手順、切り替えに伴う問題点を確認するのは移行テストの内容に含まれる。求める答はアとなる。

イのデータベースの確認は運用システムの運用効率・業務効率の測定で行う作業である。

ウのテストは結合テスト、システムテストで実施する。

エの機能確認テストはシステムテストで実施する。