

プログラミング演習解説

問1 エ

構造化定理に関する問題である。

構造化定理は、順次型、選択型、繰り返し型の基本制御構造を用いる。繰り返し型には前判定と後判定がある。

アの後判定繰り返しは、繰り返し処理の後で終了条件の判定を行う。従って、処理は必ず1回は実行される。先頭で終了条件の判定を行うは誤りである。

イの双岐選択は、二つの処理のうちどちらを実行するかを選択することである。前に戻るか、次の処理に進むかの判定ではない。

ウの多岐選択は、二つ以上の処理のうちどれか一つの処理を実行する場合の選択である。並列に処理をするのではない。

エの前判定繰り返しは、最初の繰り返し判定で偽になると繰り返しに入らないことになる。繰り返し処理を1回も行わないことになる。求める答えはエとなる。

問2 エ

構造化定理に関する問題である。

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

問3 ウ

プログラムの制御構造に関する問題である。

基本となるプログラムの制御構造に、順次構造、選択構造、繰り返し構造がある。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。繰り返し構造は同じ処理を繰り返し実行する制御構造で、前判定型と後判定型がある。

アは順次構造、イは繰り返し構造の後判定型、ウは選択構造、エは繰り返し構造の前判定型である。求める答えはウとなる。

問4 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5 イ

プログラム構造を記述するNSチャートに関する問題である。

アのスキーマは、データベースの仕様を記述したもので、概念スキーマ、外部スキーマ、内部

スキーマの3種類がある。

イのNSチャートは、構造化プログラミングに使用される図式の1つで、全体を四角で表し、順次、選択、反復を表すボックス図の記号を組み合わせ、論理を記述し、構造化を図る方式である。求める答えはイである。

ウのデータフローダイアグラムは、データの流れを図式化したもので、データフロー、プロセス、データストア、外部の4種類の基本要素で構成される。

エの疑似コードは、アルゴリズムを記述するために用いる人の言葉に近い言語である。

問6 ア

構造化チャートに関する問題である。

構造化チャートは、機能間の従属関係を階層的に表すものである。パラメータの記述ができるため、機能間のインタフェースが理解しやすい、階層構造で表現するため、システムを構成しているプログラム構造が理解しやすい、ループ、判定などの手続きを表現することができるなどの特徴がある。構造図の記号としては、ボックス、連結線、パラメータ、パラメータ表などを用いる。

アが構造化チャート、イは状態遷移図、ウはDFD、エはHIPOで、求める答えはアとなる。

問7 ウ

再帰プログラムに関する問題である。

アのオーバーレイサブルーチンは、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、実行時に必要なセグメントを主記憶装置に格納する手続きである。

イの関数サブルーチンは、データを入力すると、一定の規則に従って計算結果を出力する仕組みである。主プログラムの中で繰り返し使用される処理を関数として宣言する。

ウの再帰的サブルーチンは、プログラムの中に記述された命令で、そのプログラム自身を呼び出すことができるプログラムである。求める答えはウとなる。

エの再入可能サブルーチンは、同じプログラムに対する複数の処理要求に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。

問8 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。

アはリユーザブル、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9 イ

再帰呼び出しに関する問題である。

再帰呼び出しは、関数の処理中に自分自身の関数を呼び出して処理を進めることである。再帰呼び出しでは、必ず、再帰呼び出しからの脱出条件が存在し、再帰からの脱出が行われると、逐

次、元の処理に戻っていく。

アはフィードバック制御、イは再帰呼び出し、ウは静的関数、エは復帰である。求める答えはイとなる。

問10 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11 イ

プログラムの構造に関する問題である。

アの再帰的処理の特徴は、スタックを利用するL I F O方式であり、F I F O方式ではない。

イの再入可能プログラムは、手続き部分とデータ部分に分割して、データ部分をプロセス毎に持つことが必要である。正しい記述である。求める答えはイとなる。

ウの逐次再使用可能なプログラムは、データ部分のプロセス毎の管理が可能になっていないため再入可能では使用できない。

エは、再入可能であって、再帰的ではない。

問12 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F I F Oは誤りで、L I F O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13 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、ウはオーバーレイ、エはリエントラントで、求める答えはエとなる。

問14 ウ

プログラムの性質に関する問題である。

アの再帰的(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。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。

イの再使用可能(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。

ウの再使用可能(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。求める答えはウとなる。

エの再配置可能(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15 エ

リエントラントに関する問題である。

aはプログラム、bはタスク、cは再入可能である。求める答えはエとなる。

問16 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17 イ

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。

アの値渡しの場合、仮引数の値が変わっても、実引数の値は変わらない。

イの実引数から仮引数に情報を渡す方法には値呼び出しと参照呼び出しの方法がある。求める答えはイとなる。

ウの実引数も変数も定数もある。

エの仮引数は値呼び出しの場合、関数外では有効でない。

問18 エ

参照渡し、値渡しに関する問題である。

参照渡しは、ある変数のメモリ上のアドレスを得て、その変数の値にアクセスすることである。結果の格納もアドレスの位置に格納される。値渡しは、値をコピーして渡すことである。処理した結果はコピーされた変数の修正になる。コピー前の領域の値は変わらない。

SubXの関数の引数の条件から、Aは参照渡しで引数Cに渡される。Bは値渡しで引数Dに渡される。Cの値はEに代入され、Dの値はCに代入され、Aに戻される。

D=B=2であるから、C=2、A=2となる。Bの値は値渡しであり、SubX関数の処理と関係しないため、B=2となる。従って、答えはA=2、B=2となり、求める答えはエとなる。

問19 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20 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21 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 $a = 2b$ 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 $2a = b$ 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 $2a = 3b$ 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 $3a = 2b$ 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22 イ

プログラム言語の特徴に関する問題である。

C言語は、OSなど制御系プログラムなどの開発に適したプログラム言語である。メモリを直接扱うなどアセンブラと同等レベルのハードウェアに近い制御を記述することができる。

C言語の特徴

- ① 移植性の高さとモジュール開発の容易さ
- ② パッケージプログラムの開発に利用する。
- ③ 開発済みモジュールの再利用を促進
- ④ 開発効率の向上
- ⑤ プログラム品質の改善

アのCOBOLは、事務処理向けのプログラム言語で、コンパイラ言語である。インタプリタは誤りである。

イのCに関する記述は適切である。求める答えはイとなる。

ウのJavaは、オブジェクト指向のプログラム言語で、OSやパソコンの機能に依存することなく動作する。Javaのソースコードをコンパイルすると、バイトコードの中間コードに変換され、このコードをJava仮想マシンが解釈して、プログラムを実行する。Webブラウザ上で動作するプログラムをJavaアプレット、Webブラウザに関係なく動作するプログラムをJavaアプリケーションと呼ぶ。言語仕様がプラットフォームに依存するは誤りである。

エのPerlは、テキストやファイル処理に適したインタプリタ型のプログラム言語で、WWWサーバでは、CGIプログラムの標準言語として活用されている。クライアントではなくサーバ上で動作し、コンパイラではなくインタプリタである。

問23 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24 エ

Java言語に関する問題である。

Java言語は、オブジェクト指向のプログラム言語で、パソコンやプリンタ、携帯電話、テレビなどの家電製品など様々な分野で利用が考えられている。Javaで記述したプログラムをコンパイルするとJavaVM用のバイトコードが生成される。JavaVMを搭載しておれば、どのプラットフォームでも動かすことができる。

Java言語の特徴

- ① ガベージコレクション機能を備えている。
- ② ポインタの概念がない。
- ③ ネットワーク上のプログラムをダウンロードし、Web上で動かすことができる。このプログラムをJavaアプレットと呼ぶ。
- ④ JavaVMが搭載されておれば、プラットフォームに依存しない。
- ⑤ 実行速度は遅い。
- ⑥ インターネットや分散システム環境で利用されている。
- ⑦ オブジェクト指向言語である。

アはsmalltalk、イはC++、ウはHTML、エはJavaである。求める答えはエとなる。

問25 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26 ア

プログラム言語に関する問題である。

アはJava言語、イは疑似言語、ウはPerl、エはHTMLである。求める答えはアとなる。

問27 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28 ウ

Javaサーブレットに関する問題である。

アのJavaScript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リプト言語である。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。

イのJava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。

ウのJavaサーブレットは、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。求める答えはウとなる。

エのVBScriptは、インターネット用のスクリプト言語で、Visual Basic が基になっている。Webページを記述するHTMLの中に直接記述し、クライアントのブラウザ上でインタプリタで実行する。

問29 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。

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サーブレットは、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30 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。

イのサーブレットは、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。

ウのスクリプトは、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リプト言語である。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31 ア

ジャバサーブレットに関する問題である。

サーブレットは、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。

アのJavaで開発されたプログラムで、サーバ上で動作する記述は正しい。求める答えはアとなる。

イはJavaアプレット、ウはJavaBeans、エはインタプリタのJava仮想マシンである。

問32 エ

XMLに関する問題である。

XMLは、WWWで使われる文書の構造を定義するための言語であり、HTMLと異なり、文書構造の指定に使うタグを独自に定義できる特徴がある。XMLはSGMLをもとに、インター

ネット向けに最適化した。

アのHTMLを基にして機能拡張したのではなく、SGMLを基にしている。

イの文書の入力、文書を通常使用するテキストで表現するため、テキスト文字の入力可能な汎用のエディタやテキスト形式の入力用ソフトウェアで可能である。

ウの文書の論理構造と表示スタイルは分離することが可能で、1つの文書から表現媒体の種類によって異なる表現が可能となる。

エの利用者独自のタグを使って文書の属性情報や論理構造を定義することができる記述は適切な内容である。求める答えはエとなる。

問33 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の中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34 イ

XMLに関する問題である。

アの開始タグ、終了タグの省略は、HTMLの場合には、終了タグの省略は可能であるが、XMLでは省略は認められない。ただし、空要素タグの場合は、<要素名/>で表現する。

イのデータは開始タグと終了タグで囲んで構成するが、データがないこともある適切な記述である。求める答えはイとなる。

ウの階層構造はルート要素と呼ばれる最上位層の要素は必ず1つでなければならない。

エのコメントは自由に使用することができるが、タグ中にコメントを記述することは許されていないため、要素名として識別に使用することはできない。

問35 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36 ウ

GPL下のOSSに関する問題である。

GPLは、ソフトウェアの自由な実行、複製、改変および再配布を、無保証の条件付きで許諾するライセンスである。ソフトウェアなどの著作権者が、自身の著作権の保持を表明したまま、複製や改変などの多くの著作権支分権の実施を複製物の受領者に許諾するとともに、その受領者が複製物またはそれらを改変した派生物を再配布することを妨げないこと、ソフトウェアの使用に伴う損害を保証しないことなどの条件に、ソフトウェアの利用を許諾する契約である。ソフトウェアのバイナリ形式のみでの配布を認めず、ソースコードの添付またはソースコードの入手方法の明示を義務づけている。

アはOSSとアプリケーションとのインタフェースの開発、販売なので問題ない。

イは自社内での使用であり、問題ない。

ウはOSSを利用した改変、販売であるからライセンス違反となる。求める答えはウとなる。

エは自社開発したソフトウェアのOSS環境でのテストであるから問題ない。

問37 ア

オープンソースに関する問題である。

オープンソースは、ソースプログラムが公開されていることである。オープンソース化することで、多くの人々が開発に参加でき、結果としてそのソフトウェアがより改善され、普及していくとされる。一定の条件下で、ソースコードの変更を許している。

アはオープンソース、イはシェアウェア、ウはパブリックドメインソフトウェア、エはフリーウェアである。求める答えはアとなる。

問38 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39 ウ

オープンソースライセンスに関する問題である。

アのBSDライセンスは、無保証・免責を宣言し、再配布する際に著作権表示を行うことのみを条件とする、極めて制限の緩いライセンスである。

イのコピーライトは、著作権のことである。ホームページやブログはもちろん、映画や音楽、本や漫画、絵や彫刻など、全ての作品に著作権があり、年齢、性別、プロ、アマ問わず誰が作った作品にでも、自動的に発生するものである。

ウのコピーレフトは、著作権に対する考え方で、著作権を保持したまま、二次的著作物も含めて、すべての者が著作物を利用・再配布・改変できなければならないという考え方である。求める答えはウとなる。

エのデュアルライセンスは、ソフトウェアなどを複数の異なるライセンスに基いて配布・販売などすることである。

問40 ア

Ajaxに関する問題である。

アのAjaxは、ウェブブラウザ内で非同期通信とインターフェースの構築などを行う技術であり、動的なユーザインタフェースを画面遷移を伴わずに実現する。

イのCSSは、ウェブページのレイアウトやデザインのスタイルを宣言する仕組みである。

ウのRSSは、ウェブサイトの見出しや要約などを記述したファイルのフォーマットである。

エのSNSは、ソーシャルネットワーキングサービス、または同サービスを提供しているウェブサイトのことである。

問41 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42 イ

データベース言語SQLに関する問題である。

アのCOBOLは、事務処理用に使用するプログラム言語で、CODASYLによって開発された。ファイルの構造の定義、効率的なファイルの入出力、整列機能、報告書作成機能、COPY機能が特徴的である。プログラムは見出し部、環境部、データ部、手続き部からなる。

イのSQLは、関係データベースを構築しアクセスするための言語で、コマンド形式で入力することでデータの検索、更新、表の定義などを行うことができる。求める答えはイとなる。

ウのLISPは、人工知能関係のソフトウェア開発、数式処理のためのプログラム言語である。数式などをリストのデータ構造で表現する。関数型言語で、標準関数とユーザ定義関数を組み合わせ、帰納的・再帰的定義によって記述する。

エのALGOLは、科学技術計算用言語で、ヨーロッパを中心に使用された。BNFで言語が定義されており、ブロックと複合文で構造的に記述できる。

問43 ウ

SGMLに関する問題である。

アのDMLはデータベースのデータ操作を行う命令で、データの格納、検索、変更などを行う。

イのHTMLはWWWで利用するハイパーテキストの構造を定義するための言語で、文書に見出し、テキスト、表、リスト、画像など多彩な情報を埋めることができる。他のウェブページやデータへのリンクを設定できるのが特徴である。

ウのSGMLは文書の構造や意味を定義するためのマークアップ言語の1つで、ISOで規格化されており、文書データベースの作成などに使われている。論文やマニュアルなどの文書にタグを加えることで、文書の論理的構造を指定する。それぞれのタグがどのような意味を持つかはDTDで定義される。電子的な文書の管理や交換を容易に行うための標準文書記述言語である。求める答えはウとなる。

エのUMLはオブジェクト指向分析・設計で利用されるモデリング言語である。

問44 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45 エ

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。

アのDMLはデータベースをアクセスするための言語である。

イのHTMLはインターネットのホームページを作成するための記述言語である。

ウのPDLはページ記述言語で、プリンタで印刷する文書の体裁をページ単位に処理するための言語である。

エのSGMLは文書の形式や構造を簡単なマークによって指定するための言語で、文章中にタグと呼ばれる文字列を入れることによって、単語や文章の形式や構造を指定する。電子出版や文書データベースに利用される。求める答えはエとなる。

問46 ウ

S G M Lに関する問題である。

S G M L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。

アの英文の文書構造に用いて、和文には適さないは誤りである。

イの印刷用の制御コードは含まれていない。

ウのタグを付けて文書の構造を表現する内容は適切な記述である。求める答えはウとなる。

エの文字の大きさや段組などのレイアウトを標準化する言語は誤りである。

問47 イ

C S Sに関する問題である。

アのC M Sは、W e bコンテンツを構成するテキストや画像、レイアウト情報などを一元的に保存・管理し、サイトを構築したり編集したりするソフトウェアまたはシステムのことである。

イのC S Sは、H T M Lで書かれたW e bページのレイアウトやデザインといったスタイルを宣言するための仕組みである。文書の文字の大きさ、文字の色、行間などの視覚表現を扱う標準仕様である。求める答えはイとなる。

ウのR S Sは、ニュースやブログなど各種のウェブサイトの更新情報を簡単にまとめ、配信するための幾つかの文書フォーマットの総称である。

エのW i k iは、W e bブラウザから簡単にW e bページの発行・編集などが行える、W e bコンテンツマネジメントシステムである。閲覧者が簡単にページを修正したり、新しいページを追加したりできるようになっている。

問48 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、I B Mによって開発された統合開発環境(I D E)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J a v aで記述されている。求める答えはアとなる。

イのG C Cは、G N Uのコンパイラ群である。

ウのLinuxは、UNIXライクなO S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O S環境全体を指して、Linuxと呼ばれていることが多い。

エのT O M C A Tは、J a k a r t a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、J a v aサーブレット・J S Pを処理するアプリケーションサーバである。

問49 ア

原始プログラム作成からロード、実行に至るプロセスに関する問題である。

正しい手順は、原始プログラムの作成→コンパイル→連携編集→ロード→実行の順に行われる。求める答えはアとなる。

問50 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51 ア

コンパイラの処理順序に関する問題である。

コンパイラは、ソースプログラムからオブジェクトプログラムを作成し、オブジェクトプログラムからリンカを使用してライブラリとリンクし、ロードモジュールを作成する。ローダは、データやプログラムをコンピュータに読み込むプログラムである。

aはコンパイラ、bはリンカ、cはローダで、求める答えはアとなる。

問52 ウ

コンパイラ、インタプリタに関する問題である。

コンパイラは、C言語などのプログラミング言語で記述されたソースプログラムを、CPUが解釈・実行できる機械語に翻訳するソフトウェアである。コンパイルを行うとソースプログラムからオブジェクトファイルが生成され、リンカを使ってライブラリをリンクして、実行可能プログラムファイルを作成する。インタプリタに比べてデバッグ作業に手間がかかるが、実行速度は非常に速い。

アのCOBOLやFORTRANはコンパイラである。

イのインタプリタの実行速度はコンパイラより遅い。

ウの記述は、インタプリタに関する内容である。求める答えはウとなる。

エの記述は、インタプリタに関するもので、コンパイラではない。

問53 イ

コンパイラに関する問題である。

アの原始プログラムを解釈し実行するのは、インタプリタに関する記述である。

イの構文解析、意味解析を行い、コードを生成するのは、コンパイラに関する記述であり、求める答えである。

ウの実行するアドレスに合わせてプログラムを再配置するは、仮の記憶領域に割り付けられているプログラムの格納位置を、実行に必要な記憶領域の格納位置に再設定することによって、コンパイラの基本機能ではなく、連携編集プログラムまたはロードモジュールプログラムの機能である。

エのロードモジュールの編集は、オブジェクトモジュールを幾つか組み合わせて実行可能プログラムにする連携編集プログラムである。

問54 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55 ア

言語プロセッサに関する問題である。

コンパイラの機能は字句解析→構文解析→意味解析→コード生成→目的プログラムの最適化の順に次のように進める。

① プログラムを構成する単語を分析する。(字句解析)

② プログラムの文法的な誤りをチェックし、さらに解析を進める。(構文解析)

③ プログラムの意味を解析する。(意味解析)

④ 機械語のコードをつくる。(コード生成)

⑤ 処理の最適化を図る。(最適化)

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ためのプログラムで、次のような作業を行う。

① プログラムを1行ごとに解釈し、中間コードに変換する。

② 中間コードに対応する機械語の命令を呼び出す。

- ③ CPUが呼び出した機械語の命令を実行する。
イ、ウ、エはインタプリタの特徴である。
アはコンパイラの特徴である。求める答えはアとなる。

問56 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57 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58 エ

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。

アの中間コードはJavaマシン上で動作するバイトコードを生成する現象であり、最適化ではない。

イはクロスコンパイラ、ウはクロスリファレンス、エが最適化である。求める答えはエとなる。

問59 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60 エ

四つ組みに関する問題である。

$op(a, b, c)$ は、 $c = a \ op \ b$ を意味する表記であるから、与えられた四つ組みの表記を順次実行すると次のようになる。

$$\diagdown(c, d, w1) \quad w1 = c \diagdown d$$

$$+(b, w1, w2) \quad w2 = b + w1 = b + c \diagdown d$$

$$\diagdown(e, f, w3) \quad w3 = e \diagdown f$$

$$-(w3, g, w4) \quad w4 = w3 - g = e \diagdown f - g$$

$$*(w2, w4, x) \quad x = w2 * w4 = (b + c \diagdown d) * (e \diagdown f - g)$$

求める答えはエとなる。

問61 エ

連係編集プログラムに関する問題である。

連係編集プログラムは、オブジェクトプログラムとプログラムライブラリから呼び出したサブルーチンを結合させ、実行可能なプログラムを作るプログラムである。複数本のコンパイル単位のモジュールのオブジェクトコードを一つのプログラムに連結編集して結合する。

アはライブラリアン、イはローダ、ウはコンパイラ、エは連係編集プログラムで、求める答えはエとなる。

問62 イ

インタプリタに関する問題である。

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。インタプリタは、1行ごとに翻訳実行し、作成途中でもそれまでの部分を実行でき、変更が容易である。速度はコンパイラのほうが速い。

アはアセンブラ、イがインタプリタ、ウはコンパイラと連携編集プログラム、エはジェネレータである。求める答えはイとなる。

問63 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64 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65 イ

動的リンクライブラリに関する問題である。

アは静的リンク、イは動的リンクである。求める答えはイとなる。

ウ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。

エ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。

問66 ア

動的リンクに関する問題である。

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。

アは動的リンク、イは静的ローディング、ウは動的ローディング、エは連携編集プログラムによるロードモジュールの作成である。求める答えはアとなる。

問67 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に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カージェディタで結合して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68 ア

クロスコンパイラに関する問題である。

アのクロスコンパイラは、ほかのコンピュータで動くオブジェクトプログラムをつくるためのコンパイラである。求める答えはアとなる。

イのコンパイラコンパイラは、プログラミング言語の仕様を与えると、コンパイラプログラムを自動的につくるためのソフトウェアである。

ウのプリプロセッサは、プログラムの作成能力を上げるために、コンパイラで翻訳できない括

張した疑似命令を使用して作られた原始プログラムを、前もって解読可能な命令に置き換えるプログラムである。ソース・コード中に頻繁に出てくる数値を文字に置き換えておいたり、いつも必要なライブラリの初期化などの一連の手続きを別のファイルに用意しておいて、ソース・コードではそこを参照するようにしておくような機能に用いられる。C言語などのコンパイラで用いられ、コンパイル前にマクロの展開やヘッダファイルの挿入などの処理を行う。

エのフロントエンドプロセッサは、ホストコンピュータに先立って、通信制御などを行う処理装置やかな漢字変換を行うプログラムを示す場合に用いられる。

問69 ア

クロスコンパイラに関する問題である。

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。あるコンピュータで動作するプログラムを作るときに、他のコンピュータの開発環境を用いてプログラムを作成する場合に使用する。動作速度が遅いコンピュータや完成していないコンピュータ、開発環境のない組込用コンピュータなどのソフトウェア開発に使用される。

アがクロスコンパイラ、イはコンパイラコンパイラ、ウはプリプロセッサ、エはトレーサである。求める答えはアとなる。

問70 イ

クロスコンパイラに関する問題である。

アのエミュレータは、あるシステム上で、異なるシステム用に開発されたOSやアプリケーションを動作させるソフトウェアやハードウェアのことである。

イ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。

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ほぼ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。

異なる命令形式をもつコンピュータ用の目的プログラムを生成するのはクロスコンパイラであり、求める答えはイとなる。

問71 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に使用する。

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72 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73 ア

ジェネレータに関する問題である。

アのジェネレータは、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。求める答えはアとなる。

イのアセンブラは、アセンブラ言語で記述された原始プログラムを目的プログラムに変換する翻訳プログラムで、原始プログラムの命令と出力する機械語の目的コードが1対1に対応しているので、翻訳作業は単純である。

ウのインタプリタは、原始プログラムを目的プログラムに変換せずに、そのまま実行する方式である。インタプリタは、解釈・実行を繰り返して行うのでプログラムが完成していなくても試験的に実行して処理結果を判断し、エラーがあればただちに修正することができる。

エのコンパイラは、高水準・手続き型言語で記述された原始プログラムを機械語の目的プログラムに翻訳するプログラムである。

問74 ア

アプレットに関する問題である。

アのアプレットは、小さなアプリケーションソフトウェアのことであるが、Javaアプレットの意味に用いられる。JavaアプレットはHTMLファイルに組み込まれてWebサーバからダウンロードし、Webブラウザ上で実行されるものである。求める答えはアとなる。

イのサーブレットは、サーバ上で動作するJavaアプレットのことである。主にWebサーバ上で使用され、データベースの検索などクライアントに対して様々なサービスを提供する。

ウのスクリプトは、ソフトウェアに一連の処理手続きを実行させるためのプログラムで、C言語風の命令やマクロ言語で記述される。HTML内で直接記述されるJavaScriptやVBScript、CGIアプリケーションに利用されているPerlなどがある。

エのスレッドは、OSがCPUに割り当てる最小単位で、タスクを更に細かなスレッド単位に分割し、CPUに処理を割り当てることで、複数のプログラムを同時に効率的に実行できる。

問75 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76 エ

テストの目的に関する問題である。

単体テストは、最小単位の品質を確保するためのテストで、その目的はバグを見つけることである。見つかったバグの原因を追跡するためには小さな単位で実行することが望ましい。原因が解れば対策を講じることが可能になる。しかし、最も重要な目的はバグを見つけることである。

アのバグがないことを示すことは、テストの結果から諸条件を考えて論理的に展開することは可能であるが、現状ではバグが0であると断定することは難しい。テストを実施することの目的とは異なる。

イのバグの原因追及は、バグの再発生を防止するために考える内容であり、テスト結果に基づいて次のステップで考える内容である。

ウのバグの修正は、テストの結果に基づいて、改善の対策を考える段階の問題であり、テストの直接の目的ではない。

エのバグを見つけることがテストの最も重要な目的である。求める答えはエとなる。

問77 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78 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79 イ

プログラムの機能をテストする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。プログラムの機能をテストするものである。求める答えはイである。

ウ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。

エ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。

問80 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81 イ

ブラックボックステストに関する問題である。

ブラックボックステストは、プログラムの外部仕様をもとにテストケースを設計するための技法で、プログラムの詳細なアルゴリズムの仕様は参照しないで、プログラムの機能仕様やインタフェース仕様だけを用いて、設計する。

アの網羅率を問題にするのはホワイトボックステストである。

イの冗長なコードの検出はブラックボックステストでは不可能である。求める答えはイとなる。

ウの内部構造に着目するテストはホワイトボックステストである。

エの分岐命令やモジュールの数が増えると、テストケースが急増するのはホワイトボックステストである。

問82 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83 イ

ブラックボックステストに関する問題である。

ブラックボックステストは、プログラムの外部仕様をもとにテストケースを設計するための技法で、プログラムの詳細なアルゴリズムの仕様は参照しないで、プログラムの機能仕様やインタフェース仕様だけを用いて設計する。プログラムの設計内容から、プログラムの機能とデータの関係性を考慮して、テストデータを作成し、プログラムのテストを行う。技法として、同値分割、限界値分析、因果グラフ、実験計画法などがある。

アは机上デバック、イはブラックボックステスト、ウ、エはホワイトボックステストの考え方である。求める答えはイとなる。

問84 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85 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86 イ

ブラックボックステストのテストケースの問題である。

ブラックボックステストは、プログラムの外部仕様をもとにテストケースを設計するための技法で、プログラムの詳細なアルゴリズムの仕様は参照しないで、プログラムの機能仕様やインタフェース仕様だけを用いて設計する。

アのテストケース設計のタイミングは、プログラムの仕様を作成した段階である。

イの内容は適切である。求める答えはイとなる。

ウ、エの内容はホワイトボックステストである。

問87 ウ

ホワイトボックステストに関する問題である。

ホワイトボックステストは、プログラムの制御の流れに着目し、プログラムのステップの重要な部分を通るようなテストデータを作成し、テストする方法である。プログラムの内部仕様をもとにして、テストケースを設計する技法で、プログラムの内部構造や論理を詳細に調べるため、プログラムの立場から見た詳細な機能テストは行えるが、仕様にはあるがプログラムに実現されていない機能のエラーを発見できない問題がある。規模の大きいプログラムでは、代表的な正常処理の経路と異常処理や例外処理の経路を中心にテストケースを設計する。すべてのステップを網羅するテストデータは膨大になるため、命令網羅、判定条件網羅、条件網羅、複数条件網羅などの簡略化した方法を利用してテストケースを設計する。

ア、イ、エはブラックボックステストに関する内容であり、ウがホワイトボックステストに関する内容である。求める答えはウとなる。

問88 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89 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90 イ

ホワイトボックステストの判定条件網羅に関する問題である。

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。

この流れ図では次の条件の内容に分けることができる。

- ① 最初の判定条件の真の場合は $A = 1$ かつ $B = 0$
- ② 最初の判定条件の偽の場合は $A = 1$ かつ $B = 1$ 、 $A = 0$ かつ $B = 1$ 、 $A = 0$ かつ $B = 0$ のいずれか。
- ③ 二つ目の判定条件の真の場合は $A = 1$ かつ $C = 0$
- ④ 二つ目の判定条件の偽の場合は $A = 1$ かつ $C = 1$ 、 $A = 0$ かつ $C = 1$ 、 $A = 0$ かつ $C = 0$ のいずれか。

この流れ図では $C = 1$ であるから、2番目の判定条件はすべて偽となる。従って、判定条件網羅では最初の判定条件の真の場合と偽の場合の内の1つを実行すればテスト条件は十分になる。

肯定の $A > 0$ かつ $B = 0$ は $A = 1$ かつ $B = 0$ で1ケース、否定は $A = 1$ かつ $B = 1$ で否定の条件判定を行う。求める答えはイとなる。

問91 エ

ホワイトボックステストの条件網羅に関する問題である。

判定条件網羅の場合、 $(A = 4, B = 1)$ はA、B共に偽で、流れ図のNo.の処理になる。

$(A = 5, B = 0)$ はAは偽、Bは真であり、ORであるから流れ図の処理はYesになる。

複数条件網羅にする場合、A、B共に真、Aは真、Bは偽の2つのテストケースを追加する必要がある。

アの場合、 $(A = 3, B = 0)$ はAは偽、Bは真であり、 $(A = 7, B = 2)$ はAは真、Bは偽となる。

イの場合、 $(A = 3, B = 2)$ はAは偽、Bは偽であり、 $(A = 8, B = 0)$ はAは真、Bは真となる。

ウの場合、 $(A = 4, B = 0)$ はAは偽、Bは真であり、 $(A = 8, B = 0)$ はAは真、Bは真となる。

エの場合、 $(A = 7, B = 0)$ はAは真、Bは真であり、 $(A = 8, B = 2)$ はAは真、Bは偽となる。求める答えはエとなる。

問92 ア

判定条件網羅に関する問題である。

判定条件網羅は、判定結果が真または偽となるすべての経路を網羅し、かつすべての命令を少なくとも1回は実行するようにテストケースを設計する。

このテスト内容で判定条件網羅の条件を満たすのは、真となるのは次の3つの内の1つが成り立てばよい。

1. 条件1または条件2と条件3が同時に真となる。
2. 条件1が真となる。
3. 条件2と条件3が同時に真となる。

偽となる条件は次のいずれかが成り立てばよいことになる。

4. 条件1が偽、条件2または条件3のどちらかが偽になる。
5. 条件1、条件2、条件3いずれもが偽になる。

終了したテスト項目では(1)が2の場合、(2)が3の場合で、偽となる条件のテストが行われていない。従って、条件1が偽、条件2が偽、条件3が真のテストが必要である。求める答えはアとなる。

問93 エ

ホワイト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。

ウ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。

エ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。

プログラムの構造や制御の流れに着目し、プログラム内のすべての経路を網羅するテストはホワイトボックステストで、求める答えはエとなる。

問94 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。

アの条件網羅は、処理の分岐条件で条件式に着目し、真偽について少なくとも1回は実行するようにテストケースを作成する。判定結果よりも条件式に着目する。最低2回テストする。

イの判定条件網羅は、分岐条件の判定結果である真偽の両方の判定を少なくとも1回実行するようにテストケースを設計する。最低2回テストする。

ウの複数条件網羅は、分岐条件の判定結果の真偽を通過するあらゆる組合せを網羅するテストケースを作成する。最低4回テストする。

エの命令網羅は、プログラム中のすべての命令を1回は実行するようにテストケースを作成する。最低1回はテストする。

テストケースが最大になるのは複数条件網羅である。求める答えはウとなる。

問95 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。

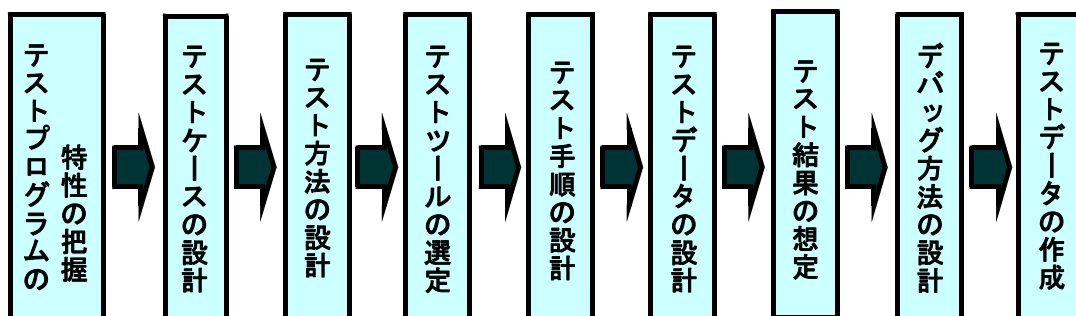
問96 エ

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97 エ

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。

デバッガは、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ークポイント機能、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。

アはダンプツール、イはデバッガ、ウはスナップショットダンプ、エがトレーサである。求める答えはエとなる。

問98 ウ

デバッ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、エラーを検出する動的デバッグツールで、実行を中断しデータ内容の閲覧や更新を対話形式で処理する。

イのクロスリファレンスは、プログラムで用いられている関数、変数や定数に関する名前を相互に参照する静的デバッグツールである。

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、プログラムをステップごとにエラーを検証する。

エのプリティプリンタは、字下げや予約語のフォント変更など、プログラムの整形機能をもつ静的デバッグツールである。

プログラムの動作過程を実行順にモニタリングするのはトレーサである。求める答えはウとなる。

問99 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ークポイント機能、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100 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101 ウ

トレーサに関する問題である。

アのエディタは、ディスプレイを見ながら、キーボードやマウスなどでファイルの内容を編集することができるプログラムである。

イのスナップショットダンプは、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デバッグツールである。

ウのトレーサは、命令が実行された順番やその命令が実行されたときの変数の内容などを出力するデバッグツールである。誤り発生の時点などを把握するため、命令の実行順序を調べる場合に用いる。求める答えはウである。

エのメモリダンプは、主記憶装置の指定した部分の内容を16進数や文字などで出力するデバッグツールである。

問102 ア

静的テストツールに関する問題である。

静的テストは、プログラムを実行せずに実施するプログラムテストである。プログラムソースコードを見ながら、机上で論理を追ってテストを行う。テストツールとしては、ソースコード生成ツール、プログラム構造解析ツールなどが実用化されている。これに対して、プログラムを実行して結果を確認するテストを動的テストという。

アのソースコードを解析するのは、静的テストツールである。求める答えはアとなる。

イはテストベッドツール、ウのガバレージモニタ、エのテスト自動生成ツールである。

問103 エ

ホワイトボックステストの支援ツールに関する問題である。

アのアサーションチェッカは、エラーのないプログラムを作るためのツールで、ある条件が成立していなければならない部分にチェック用のコードを入れ、その条件に違反している場合はエラーを出力することで、プログラムをチェックする仕組みである。

イのシミュレータは、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。

ウの静的コード解析はソフトウェアの解析手法の一種で、実行ファイルを実行することなく解析を行う。ソースコードに対して行われることが多く、オブジェクトコードに対して行う場合もある。

エのテストカバレッジ分析は、ソフトウェア開発において、出来上がったプログラムのテストをする際に、どの程度をテスト対象とするかまたはテストを実行したかの割合を分析することである。求める答えはエとなる。

問104 ア

プログラムの正当性を検証する手法に関する問題である。

プログラムの正当性とは、「入力条件を満たしたプログラムを実行すると、そのプログラムは確実に停止するとともに出力条件を満たした結果が得られる」ことをである。

アのアサーションチェックは、プログラミング言語などの仕様・機能の一つで、プログラムの前提として満たされるべき条件を記述し、実行時にそれが満たされていない場合にエラーや例外を発生させたり、メッセージを表示して処理を中断したりする機能である。プログラム中のバグや不具合、論理の矛盾を発見するのに使われる。プログラムの正当性検証に使用する。求める答えはアとなる。

このコード追跡は、トレーサを使用して行う。トレーサ上でプログラムを実行すると、その過程を追跡・記録し、実行された命令を順番に、レジスタや変数の値などの状態と共に表示してくれる。これを見ることで、プログラムが意図したとおりに動作しているかを調べたり、どこに誤りがあるかを発見するのが容易になる。

ウ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、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。

エのテストカバレッジ分析は、テスト対象プログラムのうち、既テスト分と未テスト分を識別するツールである。あるテストデータがプログラムのどの経路を通ったかを調べながら、プログラム全体の経路のうち約何%をカバーしたかを求めることができる。

問105 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106 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107 エ

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。

アの並行作業は可能である。

イはスタブではなく、ドライバが必要である。

ウはドライバがあれば、テスト済みの上位モジュールは必要ない。

エのドライバが必要は正しい。求める答はエとなる。

問108 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109 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110 イ

結合テストに関する問題である。

アの運用テストは、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ことである。

イの結合テストは、サブシステムやモジュールを結合したテストで、サブシステム間や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。求める答えはイとなる。

ウのシステムテストは、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。

エの単体テストは、プログラムの最小単位であるモジュールの品質をテストすることで、モジュール内のエラーを発見することである。

問111 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112 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113 イ

テスト支援ツールスタブの役割に関する問題である。

スタブは、テストモジュールの下位モジュールの機能をシミュレートするものである。下位モジュールの開発が未完の場合に下位モジュールの機能を代行する。

アは、テスト完了モジュールの代行ではなく、未完の下位モジュールの機能の代行である。

イのテスト対象モジュールからの呼び出し命令の条件に合わせて、値を返すは適切な記述である。求める答えはイとなる。

ウのテストデータの自動生成は行わない。

エのテスト対象モジュールから呼び出されるのであって、呼び出さない。

問114 イ

テスト支援ツールのスタブに関する問題である。

アはスナップショット、イはスタブの利用方法、ウはデバグガ、エはドライバである。求める答えはイとなる。

問115 エ

テスト支援ツールのスタブ、ドライバに関する問題である。

ドライバは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。テストモジュールから見て主プログラムの役割を果たす。結合テストのボトムアップテストに利用される。スタブはテストモジュールの下位モジュールの機能をシミュレートするものである。テストモジュールから見て副プログラムの役割を果たす。結合テストのトップダウンテストに利用される。

ア、イはドライバの説明、ウはスタブの説明、エはドライバの説明で適切である。求める答えはエとなる。

問116 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117 ウ

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。

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

ウのスタブは、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。求める答えはウである。

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118 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119 ウ

システムテストに関する問題である。

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テストを行う。

単体テストはモジュール設計工程、結合テストはプログラム設計工程、システムテストは外部設計工程、運用テストは基本計画工程の各段階に計画される。求める答えはウとなる。

問120 エ

運用テストの実施体制や手順に関する問題である。

運用テストは、運用と同じ環境で実データを使って行うテストで、利用者が新システムへ業務を移行しても問題がないかどうかをテストして確認する。従って、ユーザ部門が主体となり、実際に運用するときと同じ条件でテストする。

アの運用テストの順序はシステムテストの後工程であって、前工程ではない。

イのテストケースの設定は、ユーザ部門が行うもので、開発部門が設定するものではない。

ウのテストの主導はユーザ部門が行うもので、開発部門主導のテストではない。

エのユーザ部門が主体であり、運用の条件でのテストであるという内容は運用テストの記述である。求める答えはエとなる。

問121 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122 エ

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。

アはブラックボックステスト、イはホワイトボックステスト、ウは結合テスト、エは運用テストである。求める答えはエとなる。

問123 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124 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125 エ

レグレッションテスト(退行テスト)に関する問題である。

アの運用テストは、運用と同じ環境で実データを使って行うテストで、利用者が新システムへ業務を移行しても問題がないかどうかをテストして確認する。

イの結合テストは、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。

ウのシステムテストは、プログラム間の結合とシステムの機能を検査する。

エのレグレッションテストは、既存のソフトウェアを修正する場合、その与える影響についてテストすることで、修正内容と関係ない部分に影響を与え、異常が発生していないかどうかを確認するテストである。求める答えはエとなる。

問126 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127 エ

ビッグバンテストに関する問題である。

ビッグバンテストは、すべてのモジュールの単体テストを厳密に行い、単体テスト終了後に同時に結合してテストを行う。結合テストが短期間で行えるが、バグがあると発見が難しい。単体テスト時のドライバやスタブの作成量が多い。モジュールテストが十分に行われるため、信頼性は高い。

アはトップダウンテスト、イはホワイトボックステスト、ウはブラックボックステスト、エのビッグバンテストである。求める答えはエとなる。

問128 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、要求仕様と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。

問129 ア

ストレステストに関する問題である。

ストレステストは、システムに通常以上の負荷をかけて正常に動作するか、つまり隠れた欠陥がないか調べるリスク管理手法のひとつで、負荷テストとも言われる。

アはストレステスト、イは運用テスト、ウはベンチマークテスト、エはレグレッションテストである。求める答えはアとなる。

問130 エ

リンケージエディタに関する問題である。

リンケージエディタは、複数のモジュールのオブジェクトコードを一つのプログラムに連結編集して結合するプログラムである。入力されたオブジェクトモジュールに連続したアドレスを割り当てて、各オブジェクトモジュール間の参照を結合し、ローダが読み込める形式の統合した1本のプログラムモジュールを作り上げる。

アはグループまたはブロック内の問題で、コンパイル時の問題である。

イはプログラムの局所的な問題で、コンパイル時の問題である。

ウはプログラム内の論理的な問題であり、処理結果に現れる問題である。

エのサブルーチンや関数間の問題がリンケージエディタによって検出される問題である。求める答えはエとなる。

問131 ア

システム適格性確認テストに関する問題である。

システム適格性確認テストは、ソフトウェア構成目、ハードウェア構成目、手作業、他システムを結合した後に行われるテストである。システムに関して指定された適格性確認要求事項に従って、テストを行う。評価項目としては、システム要件の網羅性、期待した結果への適合性、運用及び保守の実現可能性、使用されたテスト手法及び標準の適切さなどがある。業務で使用するデータや業務上例外処理とされるデータを使用して実施する。

アはシステム適格性確認テスト、イはソフトウェアの結合テスト、ウ、エはソフトウェアのユニットテストに使用されるテストデータである。求める答えはアとなる。

問132 ア

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。

開発工程を含めたすべてのテスト順序

- ① モジュールの機能を検査する単体テスト
- ② 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結合テスト
- ③ プログラム間の結合とシステムの機能を検査するシステムテスト
- ④ 運用と同一の条件で行う運用テスト

アの障害発生が業務に与える影響は運用テストの結果確認できる内容であり、システムテストの段階までに発見することは困難である。求める答えはアとなる。

イのスループットの確認、ウのシステムの安全性に対する確認はシステムテストの段階で実施される。

エは単体テストの段階で検査できる。

問133 ウ

連係編集プログラムに関する問題である。

連係編集プログラムは、オブジェクトプログラムとプログラムライブラリから呼び出したサブルーチンを結合させ、実行可能なプログラムを作るプログラムである。複数本のコンパイル単位のもジュールのオブジェクトコードを一つのプログラムに連結編集して結合する。

アはライブラリアン、イはローダ、ウは連係編集プログラム、エはデバッガである。求める答えはウとなる。

問134 ア

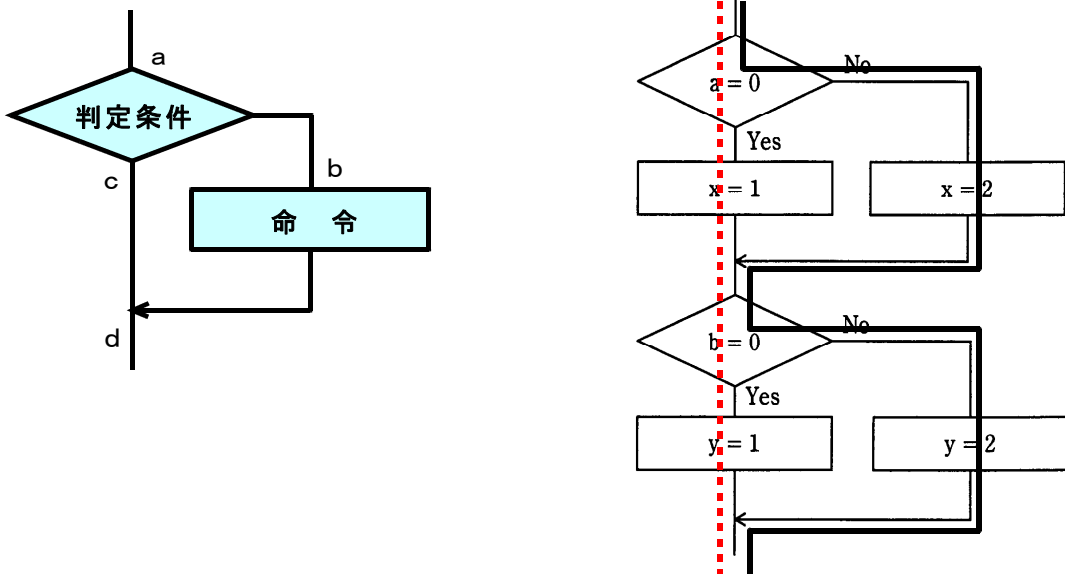
命令網羅に関する問題である。

ホワイトボックステストはプログラムの制御の流れに着目し、プログラムのステップの重要な部分を通るようなテストデータを作成し、テストする方法である。プログラムの内部仕様をもとにして、テストケースを設計する技法で、プログラムの内部構造や論理を詳細に調べるため、プログラマの立場から見た詳細な機能テストは行えるが、仕様にはあるがプログラムに実現されていない機能のエラーを発見できない問題がある。

規模の大きいプログラムでは、代表的な正常処理の経路と異常処理や例外処理の経路を中心にテストケースを設計する。すべてのステップを網羅するテストデータは膨大になるため、命令網羅、判定条件網羅、条件網羅、複数条件網羅などの簡略化した方法を利用してテストケースを設計する。

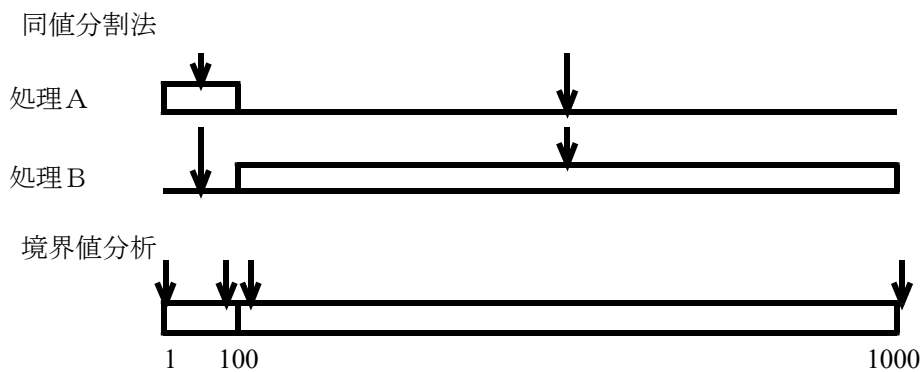
命令網羅は条件式の真偽に関係なく、すべての命令を少なくとも1回は実行するようにテストケースを設計する。繰り返しの対象のブロック内の命令も最低1回は実行させる。流れ図で a → b → d の経路のテストを実行すれば、命令を網羅したことになる。

右の図に示す黒の実線に従ったテストと赤の点線に従ったテストを行うと全ての命令を最低1回実行したことになる、命令網羅の条件を満たしたことになる。従って、テストケースは2通りあればよい。求める答えはアとなる。



問135 エ

ブラックボックステスト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。

境界値分析は、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越えた値を用いる。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。

同値分割法は、処理Aで有効同値クラスで1個、無効同値クラスで1個、計2個、処理Bで有効同値クラスで1個、無効同値クラスで1個、計2個、合計4個となる。

境界値分析は、1、100、101、1000の4個の境界値でテストする。

テストデータの最小個数は4 + 4 = 8となる。求める答えはエとなる。

問136 ウ

状態遷移テストに関する問題である。

状態遷移テストは、組込システムのソフトウェア開発における設計手法のひとつである状態遷移表に基づいて行うソフトウェアのシミュレーションテストのことである。状態遷移表を使うことでソフトウェアの状態モデル全体を網羅的にテストできるのが特長で、設計段階でのテストにより動的検証を行い、事象と状態の組合せの漏れや抜けを防止し、ソフトウェアの品質を高めることができる。組込システムの大規模化に合わせて、状態遷移表から試験経路を自動抽出したり、テストを自動実行したりするツールもある。

アは内部状態が変化するシステムのテストで用いられる。

イは内部状態に着目するホワイトボックステストである。

ウのイベントと内部状態の組合せ通りにシステムが動作することを確認するは適切な記述である。求める答えはウとなる。

エは状態遷移図、状態遷移表を使用して内部状態を解析する。

問137 イ

判定条件網羅のテストケースに関する問題である。

最初の判定条件ではXの値に対応して真と偽の2通りのケースが発生するが、次の判定条件ではA=2またはB=0の条件に対する判定であるからこの場合の結果はすべて偽になり1通りとなる。従って、2つの条件をあわせて行われるテストケースは最初の条件の2通りに対して考えればよい。テストケースの数は2となる。求める答えはイとなる。

問138 エ

Perlの実行に関する問題である。

Perlの特徴は次の通りである。

- ① インタープリタ方式であり、作成したソースコードをすぐに実行することが可能である。
- ② 正規表現をサポートしており、柔軟な文字列処理が可能である。
- ③ フリーソフトのため導入コストが低い
- ④ Webサーバー、WebアプリケーションからGUI開発、システム管理、ネットワーク管理、テキスト処理など幅広く用いられる。
- ⑤ UNIX系やWindowsなどのプラットフォーム上で動作する。
- ⑥ 付属のPerlデバッガでデバッグが可能である。

アのWindows系の言語処理系もある。

イのCGIの開発に利用されるが、Webアプリの開発にも用いられる。

ウのデバッグは付属のデバッガが利用できる。

エはインタープリタであり、事前のコンパイルを必要としない。求める答えはエとなる。