

プログラミング演習問題

問1

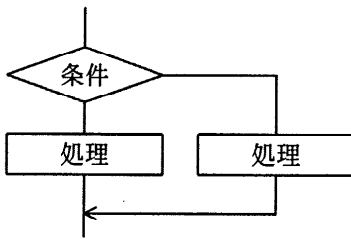
プログラムの制御構造に関する記述のうち、適切なものはどれか。

- ア “後判定繰返し” は、繰返し処理の先頭で終了条件の判定を行う。
- イ “双岐選択” は、前の処理に戻るか、次の処理に進むかを選択する。
- ウ “多岐選択” は、二つ以上の処理を並列に行う。
- エ “前判定繰返し” は、繰返し処理を1回も行わないことがある。

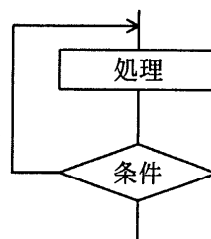
問2

プログラムの制御構造のうち、While型の繰返し構造はどれか。

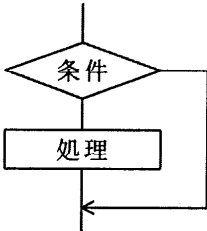
ア



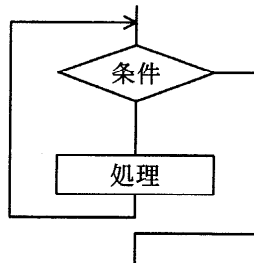
イ



ウ



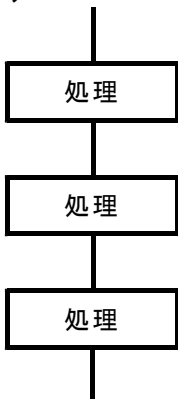
エ



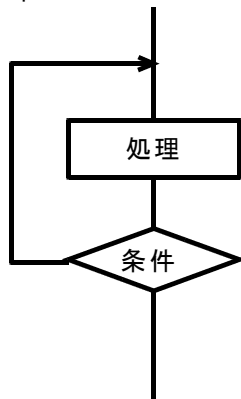
問3

次のプログラムの制御構造のうち、選択構造はどれか。

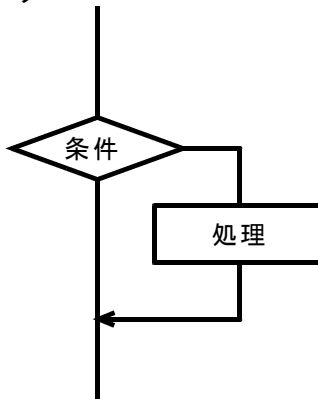
ア



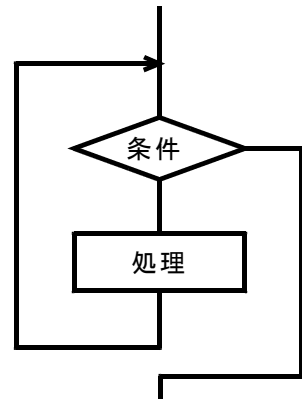
イ



ウ



エ



問4

プログラムの工程を説明しているのは、次のうちのどれか。

- ア システム化計画、プロジェクト実行計画、要求定義の3つに分けられる。
- イ プログラムの構造化設計を行う。プログラムの分割技法を使用して、プログラムを複数のモジュールに分割する。
- ウ プログラムを意識しながら、システムの機能を分割し、構造化を行う。
- エ モジュール設計、単体テスト計画、コーディング、単体テストがある。

問5

順次、選択、反復の制御構造を表すボックス図によってプログラム構造を記述する設計手法はどれか。

- ア スキーマ
- イ NSチャート
- ウ DFD
- エ 疑似コード

問6

プログラム図式の一つである構造化チャートを説明する記述として、最も適切なものはどれか。

- ア GOTOを表現する方法をもたず、論理の階層化を図的に表現できる。モジュール内の論理を構造化して表現するのに都合の良い図式である。
- イ 制御システムなどの内部設計に広く用いられる手法である。データの流れを示せないで、ほかの図式と併用するのが合理的である。
- ウ データの流れを表すのが容易な図式である。処理の手順を表しにくい。
- エ モジュール構造図を示す図式目次と、各モジュールの機能を入力・処理・出力の形式で記述した図からなる。構造化手法の特徴であるトップダウン設計向きの図式である。

問7

自分自身を直接的または間接的に呼びだして使用するサブルーチンをなんと呼ぶか。

- ア オーバレイサブルーチン
- イ 関数サブルーチン
- ウ 再帰的サブルーチン
- エ 再入可能サブルーチン

問8

再帰的プログラムの特徴として、最も適切なものはどれか。

- ア 一度実行した後、ロードし直さずに再び実行を繰り返しても、正しい結果が得られる。
- イ 実行中に自分自身を呼び出すことができる。
- ウ 主記憶上のどこのアドレスに配置しても、実行することができる。
- エ 同時に複数のタスクが共有して実行しても、正しい結果が得られる。

問9

再帰呼出しの説明はどれか。

- ア あらかじめ決められた順番ではなく、起きた事象に応じた処理を行うこと
- イ 関数の中で自分自身を用いた処理を行うこと
- ウ 処理が終了した関数をメモリから消去せず、必要になったとき再び用いること
- エ 処理に失敗したときに、その処理を呼び出す直前の状態に戻すこと

問10

複数のプロセスから同時に呼び出されたときに、互いに干渉することなく並行して処理することができるプログラムの性質を表すものはどれか。

- ア リエントラント
- イ リカーシブ
- ウ リューザブル
- エ リロケータブル

問11

プログラムの構造に関する記述のうち、正しいものはどれか。

- ア 再帰的処理のためには、実行途中の状態をF I F O方式で記録し、制御する必要がある。
- イ 再入可能プログラムを実現するためには、プログラムを手続部分とデータ部分に分割して、データ部分をプロセスごとにもつことが必要である。
- ウ 逐次再使用可能なプログラムは、再入可能でもある。
- エ 複数のプロセスで同時に実行できるようにしたプログラムは、再帰的であるという。

問12

プログラムの各種特性に関する記述のうち、適切なものはどれか。

- ア 再帰的処理のためには、実行途中の状態をF I F O方式で記録し、制御する必要がある。
- イ 再入可能プログラムを実現するためには、プログラムを手続部分とデータ部分に分割して、データ部分をプロセスごとにもつ必要がある。
- ウ 逐次再使用可能なプログラムは、再入可能でもある。
- エ 複数のプロセスで同時に実行できるようにしたプログラムは、再帰的である。

問13

再入可能(リエントラント)プログラムの説明として、最も適切なものはどれか。

- ア 一度実行した後、ロードし直さずに再び実行を繰り返しても、正しい結果が得られる。
- イ 実記憶上のどこのアドレスに配置しても実行することが可能である。
- ウ 複数のセグメントに分割されて、セグメント単位にロードして実行することが可能である。
- エ 複数のタスクで並行して実行しても、正しい結果が得られる。

問14

処理が終了していないプログラムが、別のプログラムから再度呼び出されることがある。このプログラムが正しく実行されるために備えるべき性質はどれか。

- ア 再帰的(リカーシブ)
- イ 再使用可能(リユーズابل)
- ウ 再入可能(リエントラント)
- エ 再配置可能(リロケータブル)

問15

次の記述中の□□□□に入れるべき適切な字句の組合せはどれか。

主記憶にロードされた一つの□ a □を複数の□ b □で共用して並行に実行するためには、その□ a □が□ c □であることが必要である。

	a	b	c
ア	サブルーチン	タスク	再帰的
イ	タスク	サブルーチン	再帰的
ウ	タスク	プログラム	再入可能
エ	プログラム	タスク	再入可能

問16

次のプログラムを実行したときの結果はどれか。ここで、仮引数 x は値呼出し(call by value), y は参照呼出し(call by reference)とする。

```

メインプログラム           サブプログラム sub(x, y)
a = 3;                      x = x + y;
b = 2;                      y = x + y;
sub(a, b);                  return;

```

- ア a = 3, b = 2
- イ a = 3, b = 7
- ウ a = 5, b = 2
- エ a = 5, b = 7

問17

プログラム言語における関数呼出し時の引数の性質のうち、適切なものはどれか。

- ア 値呼出しでは、仮引数の値を変えると実引数の値も変わる。
- イ 実引数から仮引数に情報を渡す方法として、値呼出し、参照呼出しなどがある。
- ウ 実引数は変数だけであるが、仮引数は変数でも定数でもよい。
- エ 実引数は呼び出される関数の中だけで有効であるが、仮引数は関数の呼出し側でも有効である。

問18

主プログラムMainと副プログラムSubXからなる図のプログラムを実行した後の、変数A、Bの値の組合せとして、正しいものはどれか。ここで、プログラム中の[]の部分は、コードの代わりにその内容を記述したものである。

	A	B
ア	1	1
イ	1	2
ウ	2	1
エ	2	2

```

Main
[変数 A の宣言]
[変数 B の宣言]
A = 1
B = 2
Call SubX ( A , B )
End
    
```

```

SubX ( 参照渡しの仮引数 C, 値渡しの仮引数 D )
[変数 E の宣言]
E = C
C = D
D = E
End Sub
    
```

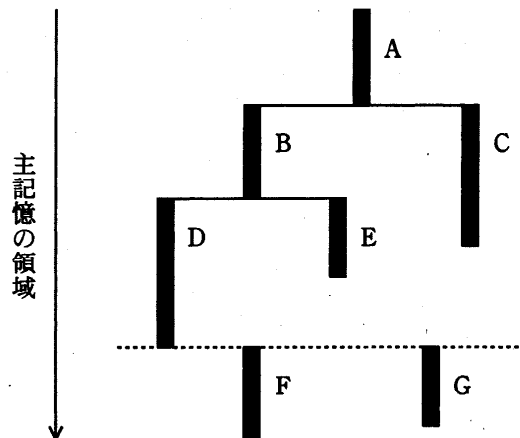
問19

サブルーチンへの引数の渡し方のうち、変数を引数として渡しても、サブルーチンの実行後に変数の値が変更されないことが保証されているものはどれか。

- ア 値呼出し イ 結果呼出し ウ 参照呼出し エ 名前呼出し

問20

モジュールA～E，共通ルーチンF，Gで構成されるプログラムを図のようなオーバレイ構造にしたとき、参照が許されるのはどれか。図は、例えばモジュールDの実行時には、主記憶にA、B、Dがロードされることを表している。また、“X→Y”はモジュールXからY内の領域をアクセスすることを表す。



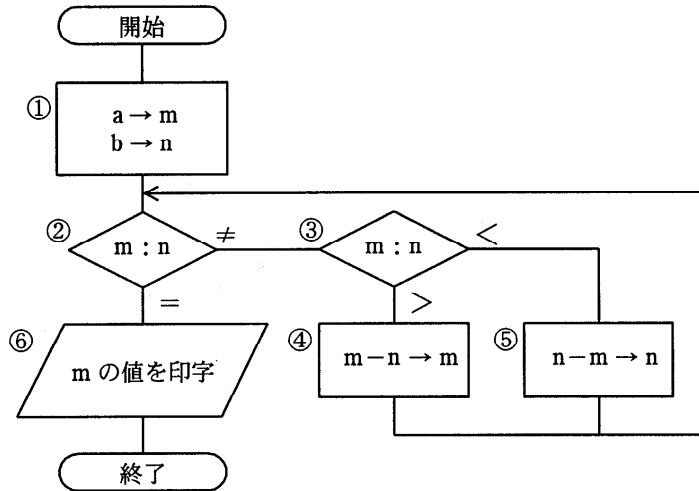
- ア B→C イ C→D ウ F→A エ G→F

問21

次の流れ図において、

①→②→③→⑤→②→③→④→②→⑥

の順に実行させるために、①においてmとnに与えるべき初期値aとbの関係はどれか。ここで、a、bはともに正の整数とする。



- ア a = 2 b
- ウ 2 a = 3 b

- イ 2 a = b
- エ 3 a = 2 b

問22

プログラム言語の特徴に関する記述のうち、適切なものはどれか。

- ア COBOLは、事務処理に適しており、インタプリタ方式で実行される。
- イ Cは、システム記述に適しており、そのプログラムの実行にはコンパイルが必要となる。
- ウ Javaは、言語仕様がプラットフォームに依存しており、インタプリタ方式で実行される。
- エ Perlは、クライアント上で動作するプログラムの記述に適しており、そのプログラムの実行にはコンパイルが必要となる。

問23

Java言語で作成されたプログラムは、異なるハードウェアや異なる基本ソフトウェア上で実行可能なことが特徴である。この実現に関連するものとして適切なものはどれか。

- ア JIT (Just in Time) コンパイラ
- イ 仮想マシン
- ウ クロスコンパイラ
- エ リバースエンジニアリング

問24

プログラム言語に関する記述のうち、Java言語の説明として適切なものはどれか。

- ア 1970年代に開発されたインタプリタ型のオブジェクト指向言語であり、エディタやデバッガなどの統合開発環境やOSの機能などを含む。
- イ C言語にクラスやインヘリタンスといったオブジェクト指向の概念を取り入れたものであり、C言語の上位互換性をもつ。
- ウ Webで用いられるマーク付き言語であり、タグによって文書の構造を識別する。テキストや動画などを関連づけたハイパテキストを作成する。
- エ ブラウザと連動して動作するアプレットなどを作成できる。このアプレットは、仮想マシンを実装する環境上であれば、どこでも実行できる。

問25

Javaなどのバイトコードプログラムをインタプリタで実行する方法と、コンパイルしてから実行する方法を、次の条件で比較するとき、およそ何行以上のバイトコードであれば、コンパイル方式の方がインタプリタ方式よりも処理時間が短くなるか。

〔条件〕

- (1) 実行時間はプログラムの行数に比例する。
- (2) 同じ100行のバイトコードのプログラムをインタプリタで実行すると0.2秒かかり、コンパイルしてから実行すると0.003秒かかる。
- (3) コンパイル時間は100行当たり0.1秒かかる。
- (4) コンパイル方式の場合は、プログラムの行数に関係なくファイル入出力、コンパイラ起動などのために常に0.15秒のオーバヘッドがかかる。
- (5) プログラムファイルのダウンロード時間など、その他の時間は無視して考える。

ア 50 イ 75 ウ 155 エ 225

問26

プログラム言語Javaに関する記述として、適切なものはどれか。

- ア インターネットや分散システム環境で利用されている、オブジェクト指向のプログラム言語である。
- イ テキスト(文字)で記述が可能な、自然言語に近いプログラム言語である。
- ウ テキストやファイルの処理に適しており、連想配列とパッケージを結び付けることが可能なプログラム言語である。
- エ ハイパテキストを記述する言語であり、アンカーというタグを用いて別の文書とリンクが可能なプログラム言語である。

問27

Javaのプログラムにおいて、よく使われる機能などを部品化し、再利用できるようにコンポーネント化するための仕様はどれか。

- ア JavaBeans
- イ JavaScript
- ウ Javaアプリケーション
- エ Javaアプレット

問28

Web環境での動的処理を実現するプログラムであって、Webサーバだけで動作するものはどれか。

- ア JavaScript
- イ Javaアプレット
- ウ Javaサーブレット
- エ VBScript

問29

JavaVMが稼働している環境だけがあれば、WebブラウザやWebサーバがなくても動作するプログラムはどれか。

- ア JavaScript
- イ Javaアプリケーション
- ウ Javaアプレット
- エ Javaサーブレット

問30

小さいアプリケーションプログラムを意味し、コンパイル済みのオブジェクトコードがサーバに格納されていて、クライアントからの要求によってクライアントへ転送されて実行されるプログラムはどれか。

- ア アプレット
- イ サーブレット
- ウ スクリプト
- エ スレッド

問31

Java Servletの説明として、適切なものはどれか。

- ア Javaで開発されたプログラムであり、クライアントの要求に応じてWebアプリケーションサーバ上で実行される。
- イ Javaで開発されたプログラムであり、サーバからダウンロードして実行される。
- ウ Javaで開発されたプログラムをアプリケーションの部品として取り扱うための規約である。
- エ Javaで開発されたプログラムを実行するインタプリタであり、バイトコードと呼ばれる中間コードを実行する機能をもつ。

問32

XMLに関する記述のうち、適切なものはどれか。

- ア HTMLを基にしてその機能を拡張したものである。
- イ XML文書を入力するためには専用のエディタが必要である。
- ウ 文書の論理構造と表示スタイルを統合したものである。
- エ 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。

問33

XMLの特徴のうち、最も適切なものはどれか。

- ア XMLでは、HTMLにWebページの表示性能の向上を主な目的とした機能を追加している。
- イ XMLでは、ネットワークを介した情報システム間のデータ交換を容易にするために、任意のタグを定義することができる。
- ウ XMLで用いることができるスタイル言語は、HTMLと同じものである。
- エ XMLは、SGMLを基に開発されたHTMLとは異なり、独自の仕様として開発された。

問34

XML文書を構成する最小単位である要素の定義方法に関する記述のうち、適切なものはどれか。

- ア 開始タグと終了タグが対になって構成され、どちらのタグも省略できない。
- イ データを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。
- ウ 一つのXML文書には、階層構造を表すために複数のルート要素を定義できる。
- エ 要素の種別を表すために注釈情報を付加して、これを要素名として識別する。

問35

OSIによるオープンソースソフトウェアの定義に従うとき、適切なものはどれか。

- ア ある特定の業界向けに作成されたオープンソースソフトウェアは、ソースコードを公開する範囲をその業界に限定することができる。
- イ オープンソースソフトウェアを改変し再配布する場合、元のソフトウェアと同じ配布条件となるように、同じライセンスを適用して配布する必要がある。
- ウ オープンソースソフトウェアを第三者が製品として再配布する場合、そのオープンソースソフトウェアの開発者は第三者に対してライセンス費を請求することができる。
- エ 社内での利用などのようにオープンソースソフトウェアを改変しても再配布しない場合、改変部分のソースコードを公開しなくてもよい。

問36

GPLの下で公開されたOSSを使い、ソースコードを公開しなかった場合にライセンス違反となるものはどれか。

- ア OSSとアプリケーションソフトウェアとのインタフェースを開発し、販売している。
- イ OSSの改変を他社に委託し、自社内で使用している。
- ウ OSSの入手、改変、販売をすべて自社で行っている。
- エ OSSを利用して性能テストを行った自社開発ソフトウェアを販売している。

問37

オープンソースに関する記述として、適切なものはどれか。

- ア 一定の条件の下で、ソースコードの変更を許している。
- イ 使用分野とユーザを制限して再配布できる。
- ウ 著作権は、放棄されている。
- エ 無償で配布されなければならない。

問38

Linuxに代表されるソフトウェアであって、再配布の自由、再配布時のソースコード包含、派生ソフトウェア改変の許諾などが要求されるものを何というか。

- ア オープンソースソフトウェア
- イ コンポーネントウェア
- ウ シェアウェア
- エ ミドルウェア

問39

オープンソースライセンスにおいて、“著作権を保持したまま、プログラムの複製や改変、再配布を制限せず、そのプログラムから派生した二次著作物（派生物）には、オリジナルと同じ配布条件を適用する”とした考え方はどれか。

- ア BSDライセンス
- イ コピーライト
- ウ コピーレフト
- エ デュアルライセンス

問40

JavaScriptの非同期通信の機能を使うことによって、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技術はどれか。

- ア Ajax
- イ CSS
- ウ RSS
- エ SNS

問41

Ajaxの説明はどれか。

- ア HTTPプロトコルを用いて、XML形式やHTML形式のデータがファイアウォールを通過できるようにした通信方式
- イ スタイルシートを適用することによって、デザインを統一したり、保守性を高めたりする仕組み
- ウ ブラウザからの要求に対してWebサーバが応答し、テキスト・画像・音声・動画などをやり取りするための通信方式
- エ ブラウザとWebサーバとがXML形式のデータを用いて非同期の通信をし、動的に画面を再描画する仕組み

問42

リレーショナルデータベースをアクセスする言語で、当初は特定のメーカーのRDB専用言語であったが、今日では多くのRDBで適用可能になり、JISとして規格が制定されているのはどれか。

- ア COBOL イ SQL ウ LISP エ ALGOL

問43

タグを使って文書の論理構造や属性を記述する方法を定めた国際規格であって、電子的な文書の管理や交換を容易に行うための文書記述言語はどれか。

- ア DML イ HTML ウ SGML エ UML

問44

SGMLに関する記述のうち、適切なものはどれか。

- ア 3次元CADフォーマットの共通化を目的に規格化された。
- イ インターネットのホームページを作成することを目的に規格化された。
- ウ 文書を宣言、文書型定義及び文書実現値の3部の構成で記述する。
- エ 文字フォントの種類や画像などのレイアウトを保持したまま表示することができる。

問45

文書の論理構造や意味構造をタグを使用して記述する言語で、電子出版や文書データベースなどに利用され、国際規格になっている言語はどれか。

- ア DML イ HTML ウ PDL エ SGML

問46

SGMLの説明として、適切なものはどれか。

- ア 英文の文書の構造を表現するために用い、和文の場合には適さない。
- イ 標準化の対象として印刷用制御コードが含まれている。これによってプリンタのハードウェアが標準化される。
- ウ 文書を構造化するための言語の一つで、<title> といったタグを付けて文書の構造を表現する。
- エ 文字の大きさや段組などのレイアウトを標準化するための言語である。

問47

HTML文書の文字の大きさ、文字の色、行間などの視覚表現の情報を扱う標準仕様はどれか。

- ア CMS
- イ CSS
- ウ RSS
- エ Wiki

問48

ソフトウェアの統合開発環境として提供されているOSSはどれか。

- ア Eclipse
- イ GCC
- ウ Linux
- エ Tomcat

問49

高水準言語で原始プログラムを作成した後、そのプログラムをコンパイル方式によって実行するまでの手順として、正しいものはどれか。

- ア 原始プログラム作成→コンパイル→連携編集→ロード→実行
- イ 原始プログラム作成→コンパイル→ロード→連携編集→実行
- ウ 原始プログラム作成→連携編集→コンパイル→ロード→実行
- エ 原始プログラム作成→連携編集→ロード→コンパイル→実行

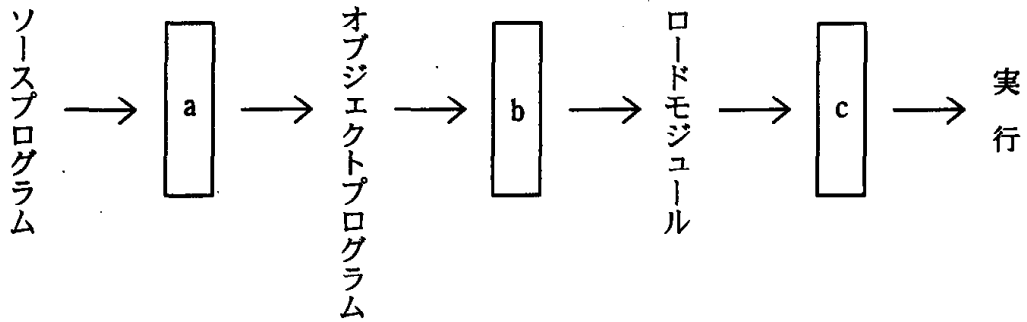
問50

言語プロセッサに関する記述のうち、正しいものはどれか。

- ア アセンブラは、ある処理系用にかかれた原始プログラムを、他の処理系用の原始プログラムに変換する。
- イ インタプリタは、他のコンピュータ用のプログラムを解読し、実行するマイクロプログラムである。
- ウ ジェネレータは、入力・処理・出力に関するいろいろな条件をパラメタで記述し、問題の処理目的に応じたプログラムを生成する。
- エ トランスレータは、高水準言語で書かれたプログラムを、解釈しながら実行する。

問51

COBOLやCでソースプログラムを作成した後、そのプログラムを実行するまでの手順として、a～cに入れるべきソフトウェアの適切な組合せはどれか。



	a	b	c
ア	コンパイラ	リンカ	ローダ
イ	コンパイラ	ローダ	リンカ
ウ	リンカ	コンパイラ	ローダ
エ	リンカ	ローダ	コンパイラ

問52

言語処理のコンパイラ、インタプリタに関して、正しい記述はどれか。

- ア インタプリタは、COBOLやFORTRANで利用されることが多い。
- イ プログラムの実行速度という点では、コンパイラを利用したときに比べてインタプリタを利用した方が速い。
- ウ インタプリタを使うと、プログラムが途中までしか完成していない場合でも、そこまでの部分を実行させることができる。
- エ コンパイラを使うと、ソースプログラムの命令文を一つずつ翻訳実行するので、1ステップずつデバッグするのに都合がよい。

問53

コンパイラについて記述したものはどれか。

- ア 原始プログラムを解釈し実行する。
- イ 構文解析、意味解析を行い、コードを生成する。
- ウ 実行するアドレスに合わせて、プログラムを再配置する。
- エ ロードモジュールを編集する。

問54

手続き形言語のコンパイラにおける処理を順に並べたものとして、正しいものはどれか。

- ア 意味解析→構文解析→字句解析→最適化→コード生成
- イ 意味解析→字句解析→構文解析→最適化→コード生成
- ウ 字句解析→意味解析→構文解析→最適化→コード生成
- エ 字句解析→構文解析→意味解析→最適化→コード生成

問55

次の文はある二つの言辞処理系について記述したものである。Bと比べたAの利点を記述しているものはどれか。

A：高水準言辞で作成されたプログラムを，中間言語，アセンブラ言辞又は機械語で記述されたプログラムに翻訳する。

B：原始プログラム中の命令文を一文ずつ解釈し，実行する。

- ア 処理の最適化が図れる。
- イ 対話的な実行環境が構築できる。
- ウ デバッグ機能を組み込みやすい。
- エ プログラム作成とテストを並行してできる。

問56

プログラムの中で使用している外部関数を見つけ，未解決アドレスとして，次のステップに渡すものはどれか。

- ア コンパイラ
- イ プリコンパイラ
- ウ リンケージエディタ
- エ ローダ

問57

コンパイラで構文解析した結果の表現方法の一つに四つ組形式がある。

(演算子，被演算子1，被演算子2，結果)

この形式は，被演算子1と被演算子2に演算子を作用させたものが結果であることを表す。次の一連の四つ組は，どの式を構文解析した結果か。ここで， T_1 ， T_2 ， T_3 は一時変数を表す。

- ($*$ ，B，C， T_1)
- ($/$ ， T_1 ，D， T_2)
- ($+$ ，A， T_2 ， T_3)

- ア $A + B * C / D$
- イ $A + B * C / T_2$
- ウ $B * C + A / D$
- エ $B * C + T_1 / D$

問58

コンパイラの最適化に関する記述として、正しいものはどれか。

- ア オブジェクトコードを生成する代わりに、インタプリタ用の中間コードを生成する。
- イ コンパイルするコンピュータとは異なる機種で動作するオブジェクトコードを生成する。
- ウ プログラムが実行されたときに、制御が渡ったルーチン名やある時点での変数の内容を表示するようなオブジェクトコードを生成する。
- エ プログラムコードを解析して、実行時の処理効率がより高くなるようにオブジェクトコードを生成する。

問59

コンパイラによる最適化の主な目的はどれか。

- ア プログラムの実行時間を短縮する。
- イ プログラムのデバッグを容易にする。
- ウ プログラムの保守性を改善する。
- エ 目的プログラムを生成する時間を短縮する。

問60

次の一連の3アドレス命令で得られる結果 x を表す式はどれか。ここで、3アドレス命令では、三つのオペランドを用いた命令 “ $c = a \text{ op } b$ ” を “ $\text{op}(a, b, c)$ ” として表記する。opは一つの演算子を表し、結果 x を表す式においては優先順位の高い順に $*$, $/$, $+$, $-$ とする。

- $/ (c, d, w1)$
- $+(b, w1, w2)$
- $/(e, f, w3)$
- $-(w3, g, w4)$
- $*(w2, w4, x)$

- ア $b + c / d * e / f - g$
- イ $b + c / d * (e / f - g)$
- ウ $(b + c / d) * e / f - g$
- エ $(b + c / d) * (e / f - g)$

問61

連携編集プログラムに関して、正しい記述はどれか。

- ア 作成したプログラムをライブラリに登録する。
- イ 実行に先立ってロードモジュールを主記憶に移す。
- ウ プログラムの構文や意味の誤りを検出する。
- エ 目的モジュールやロードモジュールを組み合わせて、ロードモジュールを作成する。

問62

インタプリタの説明として、適切なものはどれか。

- ア アセンブラ言語で書かれた原始プログラムを、機械語のプログラムに翻訳するプログラムである。
- イ 原始プログラムを、1文ずつ解析して実行するプログラムである。
- ウ 高水準言語で書かれた原始プログラムを、機械語のプログラムに翻訳してロードモジュールを作るプログラムである。
- エ 指定されたパラメータから、処理の目的に応じたプログラムを自動的に生成するプログラムである。

問63

プリコンパイラの説明として、適切なものはどれか。

- ア ある言語でコーディングされたプログラムを、別の言語のプログラムに変換するプログラムである。
- イ あるコンピュータ上で実行されるオブジェクトプログラムを、それとはアーキテクチャが異なるコンピュータ上で生成するコンパイラである。
- ウ 高水準言語でコーディングされたソースプログラムを、オブジェクトプログラムに変換するプログラムである。
- エ 高水準言語に付加的に定義された機能と文法に従ってコーディングされたプログラムを、元の高水準言語だけを使用したプログラムに変換するプログラムである。

問64

動的リンクライブラリ（DLL）に関する正しい記述はどれか。

- ア コンパイル時に、コンパイラによって組み込まれる。
- イ コンパイルの前に、プリコンパイラによって生成される。
- ウ 実行時に、オペレーティングシステムによって連携される。
- エ ロードモジュール作成時に、連携編集プログラムによって連携され組み込まれる。

問65

動的リンクライブラリ（DLL）の特徴として、適切なものはどれか。

- ア アプリケーションがメモリにロードされるときに、同時にリンクによって組み込まれる。
- イ アプリケーションの実行中、必要になったときにOSによって連携される。
- ウ コンパイル時に、コンパイラによってアプリケーションに組み込まれる。
- エ コンパイルの前に、プリコンパイラによってアプリケーションに組み込まれる。

問66

動的リンクの機能はどれか。

- ア プログラム実行時に、共用ライブラリやシステムライブラリのモジュールをロードする。
- イ プログラム実行時に、適切なアドレスに目的プログラムをロードする。
- ウ プログラム実行時に、読み込まれたページの論理アドレスを物理アドレスに変換する。
- エ プログラムの実行に先立って、複数の目的プログラムを関係編集（リンケージエディット）する。

問67

プログラムを構成するモジュールの結合を、プログラムの実行時に行う方式はどれか。

- ア インタブリタ
- イ オーバレイ
- ウ 静的リンク
- エ 動的リンク

問68

言語プロセッサが動作するコンピュータとは別の機械語をもつコンピュータの目的プログラムを生成する言語プロセッサはどれか。

- ア クロスコンパイラ
- イ コンパイラコンパイラ
- ウ プリプロセッサ
- エ フロントエンドプロセッサ

問69

クロスコンパイラに関する記述として、適切なものはどれか。

- ア 開発対象のプログラムを実行するコンピュータとは異なる種類のコンピュータで動作する。
- イ 原始言語と目的言語の定義からコンパイラを自動的に生成する。
- ウ コンパイラによる翻訳に先んじて、拡張原始言語を原始言語に展開する。
- エ プログラムを1ステップずつ読み込んで実行し、実行過程を追跡する。

問70

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言語処理プログラムはどれか。

- ア エミュレータ
- イ クロスコンパイラ
- ウ ジェネレータ
- エ シミュレータ

問71

プログラミングツールに関する記述のうち、適切なものはどれか。

- ア デバッグ時にデータ構造の内容を確認するためのツールをインスペクタという。
- イ プログラム単位の機能説明や定義を容易に探索するためのツールをトレーサという。
- ウ プログラム内又はプログラム間の実行経路を確認するためのツールをシミュレータという。
- エ プログラムのソースコードを編集するために、文字の挿入、削除、置換えなどの機能をもつツールをブラウザという。

問72

あるコンピュータの目的プログラムの命令を一つずつ解釈し、他のコンピュータの命令に変換してから実行するプログラムはどれか。

- ア シミュレータ
- イ トランスレータ
- ウ ジェネレータ
- エ クロスコンパイラ

問73

入力データの様式や処理内容などのパラメータを取り入れ、機械語のプログラムを作り出すのはどれか。

- ア ジェネレータ
- イ アセンブラ
- ウ インタプリタ
- エ コンパイラ

問74

本来は小さいアプリケーションプログラムの意味であるが、現在では、コンパイル済みのオブジェクトコードがサーバに格納されていて、クライアントからの要求によってクライアントへ転送され実行されるものを指すようになっている。このようなプログラムを何というか。

- ア アプレット
- イ サーブレット
- ウ スクリプト
- エ スレッド

問75

プログラムの静的解析ツールで検出できるものはどれか。

- ア 関数ごとの実行処理時間
- イ 後に使用されない変数への代入
- ウ プログラム仕様に対応する処理の記述漏れ
- エ 用意したテストケースでは実行されなかった命令

問76

プログラムのテストの目的として、最も重要なものはどれか。

- ア バグがないことを示すこと
- イ バグの原因を究明すること
- ウ バグを修正すること
- エ バグを見つけること

問77

システムのテストを行う際の考え方として、適切なものはどれか。

- ア テストとは、正常に動作することを確認することであり、誤った操作に対するテストは省略してよい。
- イ テストとは、品質確認のために行うものであり、正常に動作する場合、その記録は不要である。
- ウ テストを行う場合、エラーが存在しないことを完全に証明するデータを作成しなければならない。
- エ テストを行う場合、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しなければならない。

問78

プログラムモジュールの単体テストに関して、正しい記述はどれか。

- ア トップダウンテストでは、テスト対象のプログラムモジュールが呼び出す下位モジュールの代わりにするスタブが必要である。
- イ 入力条件のテストでは、プログラム設計で規定された最大値・最小値のケースが重要であり、明らかに誤った条件の入力ケースを実施する必要がない。
- ウ プログラムモジュール1本ごとの論理上の正しさを証明するものであるから、コンパイルでエラーが発生しなければ単体テスト完了とする。
- エ プログラムモジュールのコーディングが全て完了していなくても、単体テストを開始することができる。

問79

入力データと出力結果の関係に注目してテストデータを作成し、プログラムの機能をテストする手法はどれか。

- ア トップダウンテスト
- イ ブラックボックステスト
- ウ ボトムアップテスト
- エ ホワイトボックステスト

問80

プログラムの内部構造や内部理論に関係なく、プログラムの外部仕様が正しく作成されているかどうかを検査するテストはどれか。

- ア トップダウンテスト
- イ ボトムアップテスト
- ウ ホワイトボックステスト
- エ ブラックボックステスト

問81

ブラックボックステストに関する記述として、適切なものはどれか。

- ア テストデータの作成基準として、テストケースの網羅率を使用する。
- イ 被テストプログラムに冗長なコードがあっても検出できない。
- ウ プログラムの内部構造に着目し、必要な部分が実行されたかどうかを検証する。
- エ 分岐命令やモジュールの数が増えると、テストデータが急増する。

問82

プログラムテストにおける限界値分析で設定するテストデータとして、適切なものはどれか。ここで、“Aの直前の値”とは“Aより小さくてAに近い値”を指し、“Aの直後の値”とは“Aより大きくてAに近い値”を指す。

- ア 最小値、最小値の直後の値、最大値の直前の値、最大値
- イ 最小値、最大値
- ウ 最小値の直前の値、最小値、最大値、最大値の直後の値
- エ 最小値の直前の値、最小値の直後の値、最大値の直前の値、最大値の直後の値

問83

ソフトウェア開発におけるテスト技法のうち、ブラックボックステストに関する記述として、適切なものはどれか。

- ア 原始プログラムを解析し、プログラムの制御の流れと変数などのデータの流れをテストするものであり、主にプログラム開発者以外の第三者が実施する。
- イ プログラムが設計者の意図した機能を実現しているかどうかのテストであり、主にプログラム開発者以外の第三者が実施する。
- ウ プログラムのすべての命令が最低1回は実行されることを目的とするテストであり、主にプログラム開発者自身が実施する。
- エ プログラムの内部構造や論理が記述された内部仕様書に基づくテストであり、主にプログラム開発者自身が実施する。

問84

ブラックボックス法に属するテストケースの作成方法はどれか。

- ア 条件網羅
- イ 限界値分析
- ウ 命令網羅
- エ 判定条件網羅

問85

表は、あるプログラムの入力データを、有効同値クラスと無効同値クラスに分けたものである。同値分割法によってテストケースを設計する場合、最小限のテストデータの組合せとして、適切なものはどれか。

- ア -2, 0, 1, 5, 6, 8
- イ 0, 1, 5, 6
- ウ -1, 3, 6
- エ 1, 5

同値クラス	データ
無効同値クラス	-2, -1, 0
有効同値クラス	1, 2, 3, 4, 5
無効同値クラス	6, 7, 8

問86

ブラックボックステストにおけるテストケースの設計方法として、適切なものはどれか。

- ア プログラム仕様書の作成又はコーディングが終了した段階で、仕様書やソースリストを参照して、テストケースを設計する。
- イ プログラムの機能仕様やインタフェースの仕様に基づき、テストケースを設計する。
- ウ プログラムの処理手順、すなわちロジック経路に基づき、テストケースを設計する。
- エ プログラムのすべての条件判定で、真と偽をそれぞれ1回以上実行させることを基準に、テストケースを設計する。

問87

ホワイトボックステストのテストデータの作成方法に関する記述として、適切なものはどれか。

- ア 同値分割の技法を使用してテストデータを作成する。
- イ プログラムの外部仕様に基づいてテストデータを作成する。
- ウ プログラムの内部構造に基づいてテストデータを作成する。
- エ プログラムの入力と出力の関係からテストデータを作成する。

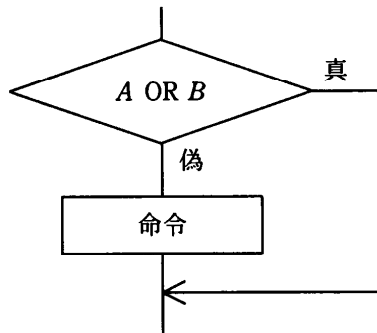
問88

ホワイトボックス法に属するテストケースの作成方法はどれか。

- ア 原因-結果グラフ
- イ 限界値分析
- ウ 条件網羅
- エ 同値分割

問89

図の論理を判定条件網羅(分岐網羅)でテストするときのテストケースとして、適切なものはどれか。



ア

A	B
偽	真

イ

A	B
偽	真
真	偽

ウ

A	B
偽	偽
真	真

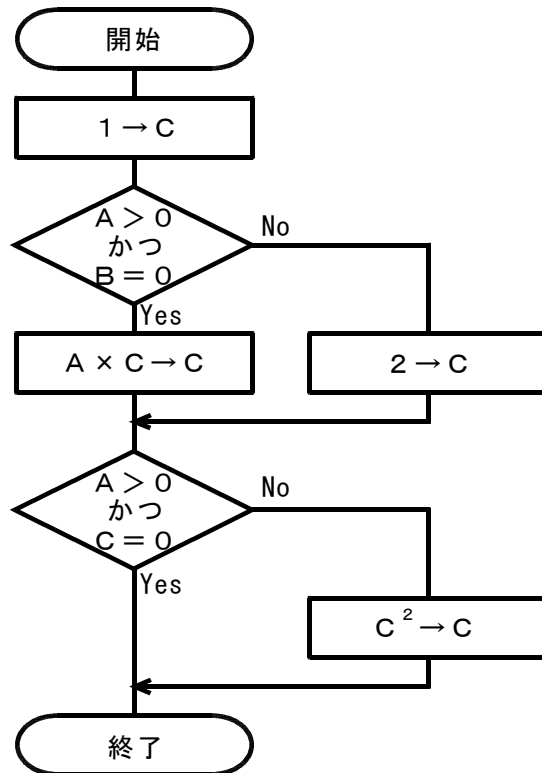
エ

A	B
偽	真
真	偽
真	真

問90

次の流れ図において、判定条件網羅(分岐網羅)を満たす最少のテストケースはどれか。

- ア (1) A = 0, B = 0
(2) A = 1, B = 1
- イ (1) A = 1, B = 0
(2) A = 1, B = 1
- ウ (1) A = 0, B = 0
(2) A = 1, B = 1
(3) A = 1, B = 0
- エ (1) A = 0, B = 0
(2) A = 0, B = 1
(3) A = 1, B = 0
(4) A = 1, B = 1

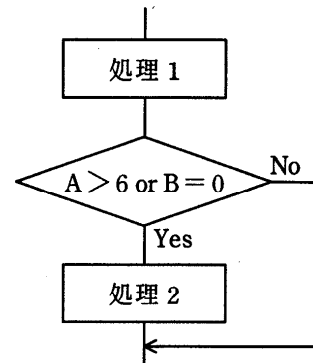


問91

プログラムの流れ図で示される部分に関するテストデータを、判定条件網羅(分岐網羅)によって設定した。このテストデータを複数条件網羅による設定に変更したとき、加えるべきテストデータのうち、適切なものはどれか。ここで、()で囲んだ部分は、一組のテストデータを表すものとする。

- ・判定条件網羅(分岐網羅)によるテストデータ
(A = 4, B = 1), (A = 5, B = 0)

- ア (A = 3, B = 0), (A = 7, B = 2)
- イ (A = 3, B = 2), (A = 8, B = 0)
- ウ (A = 4, B = 0), (A = 8, B = 0)
- エ (A = 7, B = 0), (A = 8, B = 2)



問92

プログラム中に次の複合判定がある。

条件1 OR (条件2 AND 条件3)

判定条件網羅(分岐網羅)に基づいてテストする場合、追加するテスト項目として、適切なものはどれか。

[終了したテスト項目]

- (1) 条件1が真, 条件2が偽, 条件3が偽
- (2) 条件1が偽, 条件2が真, 条件3が真

	条件1	条件2	条件3
ア	偽	偽	真
イ	真	偽	真
ウ	真	真	偽
エ	真	真	真

問93

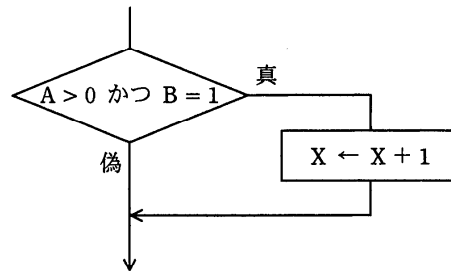
プログラムの構造や制御の流れに着目し、プログラム内のすべての経路を網羅するようなテストを行うのはどれか。

- ア トップダウンテスト
- イ ブラックボックステスト
- ウ ボトムアップテスト
- エ ホワイトボックステスト

問94

図の構造をもつプログラムに対して、ホワイトボックステストのテストケースを設計するとき、少なくとも実施しなければならないテストケース数が最大になるテスト技法はどれか。

- ア 条件網羅
- イ 判定条件網羅
- ウ 複数条件網羅
- エ 命令網羅



問95

プログラムのテストに関する記述として、適切なものはどれか。

- ア 内部構造のテストとしてブラックボックス法を用い、外部仕様のテストとしてホワイトボックス法を用いる。
- イ プログラムに残っているエラーの個数は、既に見つかったエラーの個数とは無関係である。
- ウ プログラムの完全性を証明することがテストの目的であり、エラーはすべて検出できるという仮説の下にテスト計画を立てるべきである。
- エ プログラムのテストでは、それが意図したように動くかどうかだけではなく、意図しなかった動きがあるかどうか調べる必要がある。

問96

モジュール単体テストに関する記述として、最も適切なものはどれか。

- ア 通常はコーディングを行ったプログラマではなく、専任のテスト要員がテストケースを作成し、実行する。
- イ モジュール間インタフェースは、モジュール単体ではテストできないので、単体テストの対象外となる。
- ウ モジュール設計書は、正しいことが検証済みであるので、テスト結果に問題があるときは、テストケース又はモジュールに誤りがある。
- エ モジュール設計書を見ながら、原則としてすべてのロジックパスを一度は通るようなテストケースによって、検証を行う。

問97

デバッグツールとして用いるトレーサの説明として、適切なものはどれか。

- ア 磁気テープファイルや磁気ディスクファイルなどの内容出力する。
- イ プログラムの実行中にエラーが発生したとき、メモリの内容出力する。
- ウ プログラムの特定の命令を実行するごとに、指定されたメモリの内容出力する。
- エ プログラムの命令の実行順序、実行結果などの履歴情報出力する。

問98

プログラムの動作過程を実行順にモニタリングするデバッグツールはどれか。

- ア インспекタ
- イ クロスリファレンス
- ウ トレーサ
- エ プリティプリンタ

問99

動的デバッグツールの一つであるトレーサの説明として、適切なものはどれか。

- ア プログラム中に埋め込んだデバッグ用のコードを実行するたびに、メモリやレジスタの内容を書き出す。
- イ プログラムの異常終了時に、メモリやレジスタの内容を書き出す。
- ウ プログラムの実行過程を時系列的にモニタリングするために、メモリやレジスタの内容を書き出す。
- エ プログラムの動作に必要となる環境を擬似的に作り出す。

問100

プログラムの内部処理についてテストを行う場合、指定した特定の命令が実行されるたびにレジスタや主記憶の一部の内容を出力することによって、正しく処理が行われていることを確認する支援ツールはどれか。

- ア メモリダンプ
- イ エディタ
- ウ トレーサ
- エ スナップショット

問101

論理エラーが発生し、プログラムの実行順序を調べるのに有効なテスト支援ツールはどれか。

- ア エディタ
- イ スナップショットダンプ
- ウ トレーサ
- エ メモリダンプ

問102

静的テストツールの機能に分類されるものはどれか。

- ア ソースコードを解析して、プログラムの誤りを検出する。
- イ テスト対象モジュールに必要なドライバ又はスタブを生成する。
- ウ テストによって実行した経路から網羅度を算出する。
- エ プログラムの特定の経路をテストするためのデータを生成する。

問103

ホワイトボックステストにおいて、コード中のどれだけの割合の部分を実行できたかを評価するのに使うものはどれか。

- ア アサーションチェッカ
- イ シミュレータ
- ウ 静的コード解析
- エ テストカバレッジ分析

問104

プログラム実行中の特定の時点で成立する変数間の関係や条件を記述した論理式を埋め込んで、そのプログラムの正当性を検証する手法はどれか。

- ア アサーションチェック
- イ コード追跡
- ウ スナップショットダンプ
- エ テストカバレッジ分析

問105

Webサーバの検査におけるポートスキャナの利用目的はどれか。

- ア Webサーバで稼働しているサービスを列挙して、不要なサービスが稼働していないことを確認する。
- イ Webサーバの利用者IDの管理状況を運用者に確認して、情報セキュリティポリシーとの相違を調べる。
- ウ Webサーバへのアクセス履歴を解析して、不正利用を検出する。
- エ 正規の利用者IDでログインし、Webサーバのコンテンツを直接確認して、コンテンツの脆弱性を検出する。

問106

結合テストの目的に関する記述として、適切なものはどれか。

- ア 機能が外部設計書に記されているとおりに実現されているかどうかを検証する。
- イ 処理時間や応答時間が目標を満たしているかどうかを検証する。
- ウ プログラムの部品であるモジュール間のインタフェースを検証する。
- エ 目標どおりにジョブの多重度や端末の同時接続が実現できるかどうかを検証する。

問107

ボトムアップテストの特徴として、適切なものはどれか。

- ア 開発の初期の段階では、並行作業が困難である。
- イ スタブが必要である。
- ウ テスト済みの上位モジュールが必要である。
- エ ドライバが必要である。

問108

トップダウンテストに関する記述として、適切なものはどれか。

- ア 下位のモジュールを代行するドライバの作成が必要になる。
- イ 重要度の高い上位のモジュールがテストで繰り返し使用されるので、上位モジュールの信頼性が高くなる。
- ウ テストの最終段階でモジュール間のインタフェース上の問題が生じやすい。
- エ モジュール数の少ない上位部分から開発していくので、開発の初期段階からプログラミングとテストの並行作業が可能である。

問109

テスト手法の一つであるボトムアップテストの説明として、適切なものはどれか。

- ア 下位のモジュールから上位のモジュールへと順に結合しながらテストする方法であり、未完成の上位モジュールの代わりにドライバが必要である。
- イ 個々のモジュールを独立にテストし、各々のテストが終了した時点ですべてを結合してテストする方法である。
- ウ 上位のモジュールから下位のモジュールへと順に結合しながらテストする方法であり、未完成の下位モジュールの代わりにスタブが必要である。
- エ 単体テスト、結合テスト、システムテスト、運用テストの順にテストする方法である。

問110

モジュール間やサブシステム間のインタフェースを検証するために行うテストはどれか。

- ア 運用テスト
- イ 結合テスト
- ウ システムテスト
- エ 単体テスト

問111

ボトムアップテストにおいて、被テストモジュールの上位のモジュール機能を代行するのはどれか。

- ア インタプリタ
- イ コンパイラ
- ウ スタブ
- エ ドライバ

問112

トップダウン方式で結合テストを行うとき、特に必要となるものはどれか。

- ア エミュレータ
- イ スタブ
- ウ デバッガ
- エ ドライバ

問113

結合テストで用いられるスタブの役割の記述として、適切なものはどれか。

- ア テスト完了のモジュールの代わりに結合される。
- イ テスト対象のモジュールからの呼出し命令の条件に合わせて、値を返す。
- ウ テスト対象のモジュールからの呼出し命令の条件に合わせて、テストデータを自動生成する。
- エ テスト対象のモジュールを呼出し命令で呼び出す。

問114

テスト工程におけるスタブの利用方法に関する記述として、適切なものはどれか。

- ア 指定した命令が実行されるたびに、レジスタや主記憶の一部の内容を出力することによって、正しく処理が行われていることを確認する。
- イ トップダウン的にプログラムのテストを行うとき、作成したモジュールをテストするために、仮の下位モジュールを用意して動作を確認する。
- ウ プログラムの実行中、必要に応じて変数やレジスタなどの内容を検査し、必要であればその内容を修正した後、後続の処理のテストを行う。
- エ プログラムを構成するモジュールの単体テストを行うとき、そのモジュールを呼び出す仮の上位モジュールを用意して、動作を確認する。

問115

モジュールテストで使用されるドライバ又はスタブの機能に関する記述のうち、適切なものはどれか。

- ア スタブは、テスト対象モジュールからの戻り値を表示・印刷する。
- イ スタブは、テスト対象モジュールを呼び出すモジュールである。
- ウ ドライバは、テスト対象モジュールから呼び出されるモジュールである。
- エ ドライバは、テスト対象モジュールに引数を渡して呼び出す。

問116

システム開発におけるテストでは、小さな単位から大きな単位へ、テストを積み上げて行く方法がとられることが多い。このとき、テストの適切な実施順序はどれか。

- ア システムテスト→結合テスト→単体テスト
- イ システムテスト→単体テスト→結合テスト
- ウ 単体テスト→結合テスト→システムテスト
- エ 単体テスト→システムテスト→結合テスト

問117

階層構造のモジュール群からなるソフトウェアの結合テストを、上位のモジュールから行う。
この場合に使用する、下位モジュールの代替となるテスト用のモジュールはどれか。

- ア エミュレータ
- イ シミュレータ
- ウ スタブ
- エ ドライバ

問118

システムテスト工程で実施するテストはどれか。

- ア 負荷テスト
- イ モジュール間のインタフェーステスト
- ウ モジュール仕様書に基づいた動作確認テスト
- エ レグレッションテスト

問119

システムの開発工程の段階を要求定義、外部設計、内部設計、プログラム設計及びプログラミングに分けた場合、外部設計の検証を主目的として行うテストはどれか。

- ア 運用テスト
- イ 結合テスト
- ウ システムテスト
- エ 単体テスト

問120

運用テストの実施体制や手順に関する記述として、適切なものはどれか。

- ア 運用テストは、システムテストの前工程として実施する。
- イ 開発部門がテストケースを設定し、ユーザ部門がこれに従いテストをする。
- ウ 開発部門の最後の責任として開発部門主導でテストをする。
- エ 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をする。

問121

運用テストの説明のうち、適切なものはどれか。

- ア 運用中に発生した問題に対する修正後のテストであり、原則としてユーザ部門の責任で行う。
- イ 開発したプログラムが正しく動作するかどうかを確認するテストであり、開発を担当した部門の責任で行う。
- ウ 完成プログラムを本稼働環境下で試行するテストであり、原則としてユーザ部門の責任で行う。
- エ 本稼働環境への移行を確認するテストであり、開発を担当した部門の責任で行う。

問122

運用テストにおける検査内容として、適切なものはどれか。

- ア 個々のソフトウェアユニットについて、仕様を満足していることを確認する。
- イ ソフトウェア品目の中で使用しているアルゴリズムの妥当性を確認する。
- ウ ソフトウェアユニット間のインターフェースが整合していることを確認する。
- エ 利用者に提供するという視点で、システムが要求を満足していることを確認する。

問123

システムが正常に稼働するかどうかを確認するために、システムの利用部門の利用者と情報システム部門の運用者が合同で、本稼働前に運用テストを実施することになった。利用者が優先して確認すべき事項はどれか。

- ア オンライン処理、バッチ処理などが、運用手順どおりに稼働すること
- イ 決められた業務手順どおりに、システムが稼働すること
- ウ 全てのアプリケーションプログラムが仕様書どおりに機能すること
- エ 目標とする性能要件を満たしていること

問124

システムの一部に修正を加えたときに、修正部分がほかに悪影響を及ぼさずに正しい結果が得られることを検証するテストはどれか。

- ア 機能テスト
- イ 結合テスト
- ウ 退行テスト
- エ 例外テスト

問125

ソフトウェアのテスト方法のうち、ソフトウェア保守のために変更した箇所が他の部分に影響しないかどうかを確認する目的で行うものはどれか。

- ア 運用テスト
- イ 結合テスト
- ウ システムテスト
- エ レグレッションテスト

問126

プログラムの検査に用いるテストデータに関して、適切な記述はどれか。

- ア 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。
- イ 正しく処理されるテストデータだけを、テストの進行に応じて準備する。
- ウ テスト用のデータとして、運用時に処理されるデータ量の2割程度を用意する。
- エ 入力段階のチェックでエラーになるテストデータは、準備する必要はない。

問127

ビッグバンテストの手法の説明として、適切なものはどれか。

- ア 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テストする。
- イ モジュールの内部構造に着目してテストする。
- ウ モジュールの内部構造を考慮することなく、仕様書どおりに機能が作動するかどうかをテストする。
- エ モジュールを一度に結合してテストする。

問128

システムの移行テストを実施する主要な目的はどれか。

- ア 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替え手順や切替えに伴う問題点を確認する。
- イ 既存システムのデータベースのコピーを利用して、新システムでも十分な性能が発揮できることを確認する。
- ウ 既存のプログラムと新たに開発したプログラムとのインタフェースの整合性を確認する。
- エ 新システムが要求されたすべての機能を満たしていることを確認する。

問129

ストレステストの目的はどれか。

- ア システムに要求されている処理能力の限界状態における動作を確認する。
- イ 実際に利用者に使ってもらうことによって、システムの使いやすさを評価する。
- ウ 標準的なプログラムの実行時間を計測することによって、他のコンピュータと性能を比較する。
- エ プログラムの修正又は変更によって他の機能が意図しない影響を受けていないことを確認する。

問130

次のエラーの事例のうち、リンケージエディタによって検出される事例はどれか。

- ア グループ又はブロックを示すカッコの対応が正しくなかった。
- イ プログラムで局所的に使用する変数の名前と属性の宣言を忘れていた。
- ウ プログラムの論理を間違えていた。
- エ 呼び出すサブルーチンの名前を間違えていた。

問131

システム適格性確認テストを実施するとき、用意しておくべきテストデータはどれか。

- ア 実際に業務で使うデータや、業務上例外として処理されるデータ
- イ ソフトウェアユニット間のインタフェースに関するエラーを検出するデータ
- ウ ソフトウェアユニット内の全分岐を1回以上通るデータ
- エ ソフトウェアユニット内の全命令が1回以上実行されるデータ

問132

利用部門での運用テストに入る前の段階で、開発部門がシステムテストを実施した。この段階までのテストで、見つけることが困難な項目はどれか。

- ア 障害発生が業務に与える影響
- イ スループットの不足
- ウ データ破壊や改ざんに対する安全性
- エ 論理エラー

問133

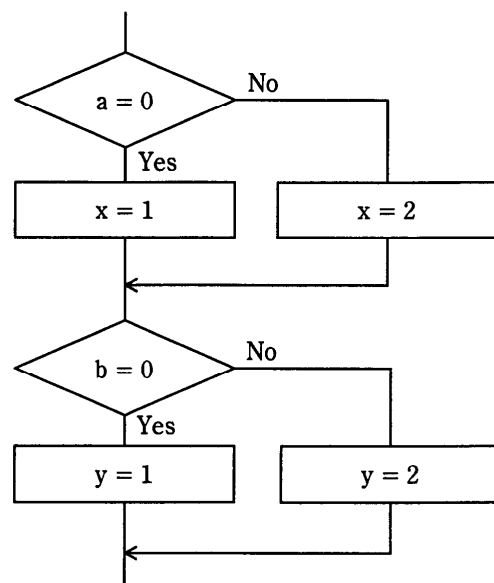
リンカの機能として、適切なものはどれか。

- ア 作成したプログラムをライブラリに登録する。
- イ 実行に先立ってロードモジュールを主記憶にロードする。
- ウ 相互参照の解決などを行い、複数の目的モジュールなどから一つのロードモジュールを生成する。
- エ プログラムの実行を監視し、ステップごとに実行結果を記録する。

問134

流れ図で表される部分を命令網羅によってテストするとき、テストケースは少なくとも幾つ用意する必要があるか。

- ア 2
- イ 3
- ウ 4
- エ 5



問135

整数 1 ~ 1,000 を有効とする入力値が, 1 ~ 100 の場合は処理 A を, 101 ~ 1,000 の場合は処理 B を実行する入力処理モジュールを, 同値分割法と境界値分析によってテストする。次の条件でテストするとき, テストデータの最小個数は幾つか。

[条件]

- ① 有効同値クラスの 1 クラスにつき, 一つの値をテストデータとする。ただし, テストする値は境界値でないものとする。
- ② 有効同値クラス, 無効同値クラスの全ての境界値をテストデータとする。

ア 5

イ 6

ウ 7

エ 8

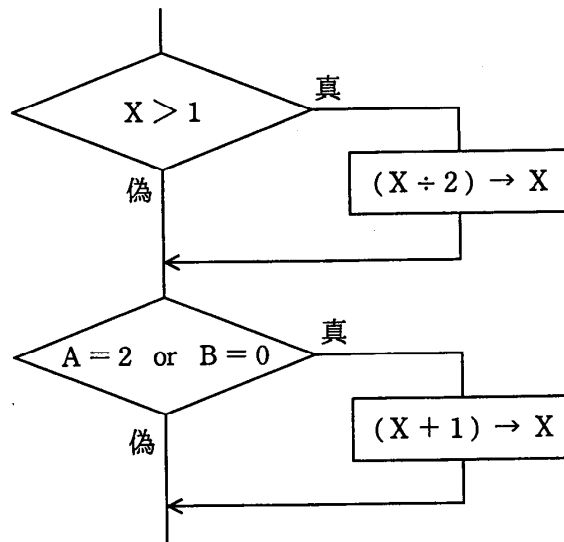
問136

システム結合テストにおける状態遷移テストに関する記述として, 適切なものはどれか。

- ア イベントの発生によって内部状態が変化しない計算処理システムのテストに適した手法
- イ システムの内部状態に着目しないブラックボックステスト用の手法
- ウ 設計されたイベントと内部状態の組合せどおりにシステムが動作することを確認する手法
- エ データフロー図, 決定表を使用してシステムの内部状態を解析する手法

問137

流れ図において, 判定条件網羅 (分岐網羅) を満たす最少のテストケース数は幾つか。



ア 1

イ 2

ウ 3

エ 4

問138

Perlの実行に関する記述のうち、適切なものはどれか。

- ア UNIX用として開発されており、Windows用の言語処理系はない。
- イ 実行にWebサーバを必要とする言語であり、CGIの開発に適している。
- ウ 動的デバッグは、言語処理系から独立したプログラムを実行して行う。
- エ プログラムをコンパイルしたファイルを事前に用意する必要はない。