

Chapter14 システム開発

14-1 システムを開発する流れ（解答・解説）

問 1 イ

〔解説〕アは運用プロセス，ウ，エは要件定義プロセスで実施する。

問 2 ウ

〔解説〕ア，イは要件定義プロセス，エは開発プロセスで定義する。

問 3 ア

〔解説〕イは要件定義プロセス，ウは問題解決プロセス，エは文書化プロセスの説明である。

問 4 イ

〔解説〕ア，エは要件定義プロセス，ウは開発プロセスの目的である。

問 5 ウ

〔解説〕アは要件定義プロセス，イは運用プロセス，エは開発プロセスで実施する。

問 6 エ

〔解説〕ア 誤り。 監査プロセスでは、客観性及び独立性を保証された立場から、 成果物及びプロセスが要求事項、計画及び合意に適合しているかどうかを判定する。

イ 誤り。 検証プロセスでは、要求事項を満たしているかを確認する。

ウ 誤り。 使用性向上プロセスでは、使いやすさや使い勝手を確認する。

問 7 イ

〔解説〕ア 自社の社員で開発する必要はない

ウ 他社と同じシステムにする必要はない

エ テスト計画，運用マニュアル及び障害対策は計画の立案時には不要である

問 8 ウ

〔解説〕アはS L A，イはR F P，エはシステム仕様書である。

問 9 イ

〔解説〕FF 関係 (Finish-to-Finish, 終了-終了) :

「受付が終了したら試験会場への入場を終了する」というように、あるアクティビティの終了が、他方のアクティビティの終了条件になっている関係。

FS 関係 (Finish-to-Start, 終了-開始) :

「受付が終了してから試験を開始する」というように、あるアクティビティの終了が、他方のアクティビティの開始条件になっている関係。

SF 関係 (Start-to-Finish, 開始-終了) :

「試験が開始されたら受付を終了する」というように、あるアクティビティの開始が、他方のアクティビティの終了条件になっている関係。

SS 関係 (Start-to-Start, 開始-開始) :

「受付が開始されたら試験会場への入場を開始する」というように、あるアクティビティの開始が、他方のアクティビティの開始条件になっている関係。

問 10 ア

〔解説〕イ RFP(Request for Proposal, 提案依頼書)の説明です。

ウ RFC(Request for Change, 変更要求書)の説明です。

エ 双方の役割分担を確認する文書はSOW(Statement of Work, 作業範囲記述書)と呼ばれるが、契約締結を要請するという意味に当てはまる用語はない。

問 11 イ

〔解説〕アは内部設計, ウはプログラム設計, エは外部設計で行う作業である。

問 12 ア

〔解説〕イは開発プロセス, ウは企画プロセス, エは運用プロセス段階で行う作業である。

問 13 ア

〔解説〕イは運用プロセス, ウは企画プロセス, エは開発プロセスで実施すべきである。

問 14 ウ

〔解説〕トレーサビリティとは、対象とする物品（製品や情報）の履歴を確認できることである。

問 15 エ

〔解説〕ア "利害関係者の識別"で実施する内容

イ "要件の評価"で実施する内容

ウ "要件の合意"で実施する内容

問 16 イ

〔解説〕非機能要件とは、機能として明確に定義できない品質要件や技術要件、運用・操作要件などのことである。

問 17 ウ

- 〔解説〕 ア 保守承認者が承認を行ってからプログラムを登録する
イ 必ずテストを行ってから更新する
エ 保守承認者が承認を行った時点で完了とする

問 18 エ

- 〔解説〕 ア 企画プロセスで行う
イ システム開発プロセスで行う
ウ ソフトウェア実装プロセスで行う

問 19 エ

- 〔解説〕 非機能要件：業務要件を実現するために必要なシステムの機能要件以外の要件のことで制約条件や品質要求がこれに該当する。
ア 業務要件に該当。
イ 業務要件に該当。
ウ 機能要件に該当。

14-2、3 システムの開発手法（解答・解説）

問 1 イ

問 2 イ

- 〔解説〕 a は基本計画，b は内部設計，c はプログラミング，d はテスト，e はプログラム設計，f は外部設計の説明である。

問 3 エ

- 〔解説〕 テスト駆動開発：XP の基本となる 12 のプラクティスのうちの 1 つで、求める機能を明確化するために、プログラムを記述するよりも前にテストケースを作成する開発手法。そのテストをパスする最低限の実装を行った後で、機能を維持したままコードを洗練していくという手順で開発を進める。

問 4 イ

- 〔解説〕 リファクタリングとは、プログラム外部から見た動作を変えずに、修正や拡張が簡単になるようにソースコードの内部構造を変更することである。

問 5 ウ

- 〔解説〕 ア 「週40時間労働」に反する。
イ 「ソースコードの共同所有」に反する。
ウ 正解。「常に統合」のプラクティスに適合する。
エ 「ペアプログラミング」に反する。

問 6 ウ

- 〔解説〕 X P（Extreme Programing）とは、柔軟性を重視したソフトウェア開発手法で、ペアプログラミング（2人のプログラマがペアを組み、コーディングを行うドライバと助言を行うナビゲータに分かれ、交代しながら作業を進める）を特徴としている。

問 7 エ

- 〔解説〕 ア 計画ゲームは、イテレーションごとに行われ、実現するユーザ機能(ストーリー)とリリース計画を顧客と交渉する作業
イ コーディング標準は、ソースコードを記述する際のルールを定めたもの
ウ テスト駆動開発は、求める機能を明確化するために、プログラムを記述するよりも前にテストケースを作成する開発手法

問 8 ア

- 〔解説〕 上流CASEツールの対象となるのは、基本設計・外部設計・内部設計。
下流CASEツールの対象となるのは、プログラム設計・プログラミング・テスト。

問 9 ウ

- 〔解説〕 SOA (Service Oriented Architecture) とは、ソフトウェアの機能を独立したサービスと見立て、それらを組み合わせ連携させることで、システム全体を構築していく手法である。

問 10 エ

- 〔解説〕 ア ERP(Enterprise Resource Planning)の説明です。
イ フィット & ギャップ分析の説明です。
ウ PDCA サイクルの説明です。

問 11 イ

- 〔解説〕 ア OSI 基本参照モデルの説明。
ウ SLM(Service Level Management)の説明。
エ SaaS(Software as a Service)の説明。

問 12 ウ

- 〔解説〕 ア フォワードエンジニアリングの説明。
イ リファクタリングの説明。
エ リエンジニアリングの説明。

問 13 エ

- 〔解説〕 ア スクライブの役割
イ オーナーの役割
ウ プレゼンタの役割

問 14 ア

- 〔解説〕 デザインレビューとは設計に対するレビューのことであり、仕様の不備や誤りの早期発見を目的として行う。

問 15 ア

- 〔解説〕 イ 開発者が主体となりエラーの早期発見を目的としてプログラムのステップごとにシミュレーションを行いながら確認をしていくレビュー手法
ウ 電子メールなどを使って、対象の成果物を複数のレビューに配布・回覧し、フィードバックを求めるレビュー方法
エ 二人一組でプログラムの実装を行い、一人が実際のコードをコンピュータに打ち込み、もう一人はそれをチェックしながらナビゲートするという役割を随時交代しながら作業を進める手法

問 16 ウ

- 〔解説〕 ア ウォークスルーの説明
イ ラウンドロビンの説明
エ パスアラウンドの説明

14-4 業務のモデル化〔解答・解説〕

問 1 ウ

- 〔解説〕 データ中心分析・設計技法は、システム設計の上流工程において、データの流れやその分析に重点をおく手法であり、モデル化に際しては実際の情報資源（データ）に着目する。

問 2 エ

- 〔解説〕 アはフローチャート、イは状態遷移図、ウはE-R図の説明である。

問 3 ウ

問 4 エ

- 〔解説〕 ア 2本の平行線はデータストア（ファイル）を意味するので、ファイル名などを付ける
イ 円はプロセス（処理）を意味するので、処理名を付ける
ウ 四角はデータの源泉と吸収を意味するので、ユーザ名や部門名などを付ける

問 5 エ

- 〔解説〕 DFD(Data Flow Diagram)は、構造化分析手法の一つでデータの流れに着目して、その対象となる業務のデータの流れと処理の関係をわかりやすく図式化する手法。

問 6 ア

問 7 ア

- 〔解説〕 イ (誤)状態遷移→(正)リレーションシップ
ウ インスタンスとはエンティティの実現値である
(例えば、エンティティがプログラム言語ならインスタンスはC、VB、J A V Aなどとなる)
エ 抽象的な概念もエンティティになりうる

問 8 イ

- 〔解説〕 ア 関係データベースの実装を前提としていない
ウ D F Dの説明である
エ リレーションシップはエンティティ間の関連を表現する

問 9 ウ

- 〔解説〕 アはUML、イは状態遷移図、エはD F Dの説明である。

問 10 ア

- 〔解説〕 イ エンティティタイプ間の関連を表す線には「 ———— ： 1対1」「 ———→ ： 1対多」「 ←——→ ： 多対多」という意味がある
ウ D F D(Data Flow Diagram)に関する記述
エ リレーションシップは、実体がもつ関連を表す

問 11 エ

〔解説〕状態遷移図とは、状態変化の契機となる事象とそれに伴う動作を記述するものであり、時間の経過や状況の変化に基づくリアルタイム制御システムの設計に適している。

問 12 エ

〔解説〕投入される硬貨は、10円、50円、100円のいずれかであり、投入による遷移は以下のようになる。

- Q0状態から10円投入でQ1、50円投入でQ5、100円投入でEに遷移
- Q1状態から10円投入でQ2、50円投入でQ6、100円投入でEに遷移
- Q2状態から10円投入でQ3、50円又は100円投入でEに遷移
- Q3状態から10円投入でQ4、50円又は100円投入でEに遷移
- Q4状態から10円投入でQ5、50円又は100円投入でEに遷移

問 13 ウ

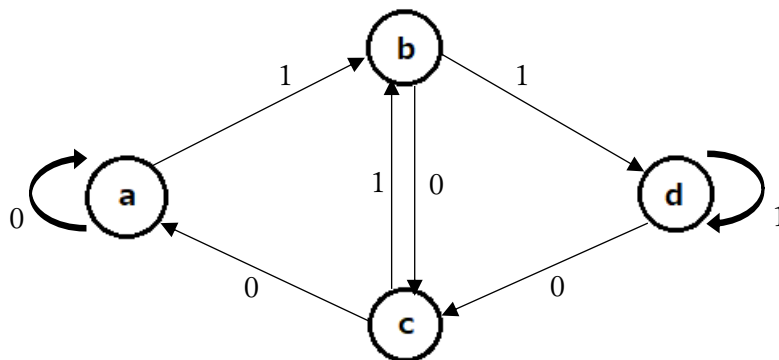
〔解説〕設問の有限オートマトンは初期状態で1が入力されると受理状態へ遷移するので文の最後は1になる。受理する過程ですが以下の2つのパターンがある。

- ・初期状態で0を繰り返して1で受理状態に遷移する
- ・初期状態と受理状態を1→0で繰り返して1で受理状態に遷移する

0または10を0回以上繰り返した後に1で終了する文字列なので、「 $(0|10)^*1$ 」が適切な正規表現。他の正規表現は少なくとも、初期状態で0を繰り返した後に1で受理する000...01というパターンにマッチしないので誤りと判断できる。

問 14 ウ

〔解説〕表の有限オートマトンを図にすると次のようになる。



ビット列「110」が入力されるとき、最後の「0」が入力されて遷移するのはaかc。
ビット列2番目の「1」が入力されている状態を踏まえるとcが適切。

問 15 イ

〔解説〕0が入力されたとき、出力は0か1かのいずれかなので、 $a + b = 1$
1が入力されたとき、出力は0か1かのいずれかなので、 $c + d = 1$

問 16 ウ

- 〔解説〕ア 矢印の○から始め、0で矢印の○に戻る。その後、0を4回繰り返しても、◎に行けない。
- イ 0で矢印の○に戻る。次の1で次の○に行く。次の1で同じ○を繰り返すため、◎に行けない。
- ウ ○→1で次の○に行く。0で◎に行く。1で◎に戻る。0で◎に戻る。◎にいるため正解。
- エ 1で次の○に行く。1で次の○に戻る。その後、1を2回繰り返しても、◎に行けない。

問 17 ウ

〔解説〕この有限オートマトンを文章で表すと、

1. S1 状態から始まる。
2. S1 状態で 0 を出力した時は S3 へ、1 を出力した時は S2 へ状態遷移する。
3. S2 状態で 0 を出力した時は S2 を維持、1 を出力した時は S1 へ状態遷移する。
4. S3 状態で 0 を出力した時は S2 へ状態遷移し、1 を出力した時は S3 を維持する。
5. 入力列は S3 で受理される。

ということになります。

選択肢の中で唯一 S3 が受理可能な入力列は「1101」。

問 18 ア

〔解説〕イ 親組織から見た子組織の多重度は 0 以上なので、子組織が存在しない場合もある

ウ 子組織が親組織となり 3 以上の階層構造となる場合もあるので誤り

エ 1 つの親組織は複数の子組織を持つことができ、1 つの子組織は複数の親組織を持つことができるので関連は網状のネットワーク構造になる

14-5 ユーザインタフェース〔解答・解説〕

問 1 ウ

〔解説〕ア 入力項目が一定でない場合、直接入力ができなければ使いづらい

イ 初心者のことを考慮すると、ダブルクリックは避けるべきである

エ 入力しやすさの観点から、入力原票と同じフォーマットにするのが望ましい

問 2 エ

〔解説〕ア 回顧法は、被験者にタスクを実行してもらい、専門家がその行動を観察し、事後の質問への回答とともに分析する手法

イ 被験者にタスクを実行してもらうのは回顧法と同様だが、思考発話法ではタスク実行時に操作方法を認知する過程をその都度、発話してもらう

ウ 認知的ウォークスルー法は、使いながら操作を理解していく際の認知モデルである探索学習理論に基づいた評価手法。まず対象とするユーザ像を設定し、ユーザの行動をシミュレーションしていくことで問題点を明らかにしていく

問 3 エ

〔解説〕パンくずリストは、比較的大規模な Web サイトで、利用者が現在どこの階層のページにいるのかをユーザに知らせ、現在位置を見失わないようにする目的で設置されるユーザインタフェース(U I)部品。

問 4 エ

〔解説〕E U C (End User Computing) とは、情報システムを利用して業務を行なうユーザが、自らシステムの構築や運用・管理に積極的に携わることである。

問 5 ウ

〔解説〕期間ごと、購入金額ごとの優良顧客数が求めやすいものがよい

ア 期間ごとの優良顧客数がわかりにくい

イ 金額ごとの優良顧客数がわかりにくい

エ 期間と購入回数との関連がわかりにくい

問 6 イ

〔解説〕 アクセシビリティとは、高齢者や障害者を含む誰もが利用できる環境であり、音声による文章の読み上げや、手書き文字による入力が行えるようにすることなどがある。

問 7 エ

〔解説〕 ユーザビリティとは使いやすさのことであり、ユーザがいかにストレスやとまどいを感じずに目標を達成できるかという度合いを指す。

問 8 エ

〔解説〕 ア ユニバーサルデザインの説明
イ アクセシビリティの説明
ウ 情報バリアフリーの説明

問 9 ウ

〔解説〕 デジタルディバイドとは、IT技術を使いこなすことができる者とできない者との待遇や貧富の格差が広がることをいう。

問10 ウ

〔解説〕 ユーザビリティとは「利用者にとっての使いやすさ」のこと。
5つの評価手法を"利用者の立場から"と"専門家の立場から"に分類すると次のようになる。

[利用者の立場からの評価手法]

アンケート、回顧法、思考発話法

[専門家の立場からの評価手法]

認知的ウォークスルー法、ヒューリスティック評価法

よって、正しい組み合わせは「ウ」となる。

	利用者の立場からの評価手法		専門家の立場からの評価手法	
ア	アンケート	利用者	回顧法	利用者
イ	回顧法	利用者	思考発話法	利用者
ウ	思考発話法	利用者	ヒューリスティック評価専門家	
エ	認知的ウォークスルー法	専門家	ヒューリスティック評価専門家	

問11 ウ

〔解説〕 アクセシビリティとは、高齢者・障害者を含む誰もが、さまざまな製品や建物やサービスなどを支障なく利用できるかどうか、あるいはその度合いのこと。

Web ページにおいては、そのページが、高齢者や障害者も含めた、誰もが情報を取得・発信できるように柔軟に設計されていて、アクセスした誰もが同様に情報を共有できるか、あるいはその度合いを意味する。

問12 イ

〔解説〕 ア アクセシビリティとは、等しい水準のユーザビリティをすべての個人について達成することではなく、少なくともある程度のユーザビリティをすべての個人について達成すること
ウ アクセシビリティ設計は、一般的な人間工学指針に従うことが重要
エ アクセシビリティ設計の対象には、平均的能力の人々だけでなく、様々な障害をもつ人々を含む最も幅広い層の人々を含む

問13 ウ

14-6 コード設計と入力のチェック〔解答・解説〕

問 1 ア

〔解説〕チェックディジットとは、入力誤りの検出のため、コードの各桁の値から計算された値をコードの末尾に付加するものである。

問 2 イ

〔解説〕チェックディジットチェックは、コードの各桁の値から計算された値をコードの末尾に付加するものである。入力の際には同じ計算を行い、結果が異なっていれば入力誤りであることがわかる。

問 3 ウ

〔解説〕 $(4 \times 3 + 2 \times 2 + \square) \div 7$ の余りが 6 であるから、 \square は 4

問 4 ウ

〔解説〕 $N_1 \times 1 + 7 \times 2 + 6 \times 3 + 2 \times 4 = N_1 + 40$ を 10 で割った余りが 4 となるのだから、 N_1 の値は 4 となる。

問 5 イ

〔解説〕アはチェックディジットチェック、ウはバランスチェック、エは重複チェックの説明である。

問 6 ウ

〔解説〕ア 区分コードは、ブロックコードとも言い、営業部は 1000～1999、企画部は 2000～2999 というようにグループごとに定められた範囲内でコードを割り当てる方式。上位桁を見るだけで、どのグループに属しているのか判断できる特徴がある。

イ 桁別コードは、各桁（または桁のグループ）に分類上の意味を持たせたコード体系。JAN コード、ISBN コード、運転免許証の番号などが桁別コードに該当。

エ 連番コードは、シーケンスコードとも言い、0001→0002→0003→…→9999 というように連続した値をデータに 1 つずつ割り当てていく方法。

問 7 エ

〔解説〕4 けたの数値データ "2 1 3 1" に設問に与えられた〔規則〕を適用すると以下のようになる。

(1) : 与えられたデータの各けたに、先頭から係数 4, 3, 2, 1 を割り当てる。

$$2(4), 1(3), 3(2), 1(1)$$

(2) : 各けたの数値と割り当てた係数とのそれぞれの積の和を求める。

$$2 \times 4 + 1 \times 3 + 3 \times 2 + 1 \times 1 = 18$$

(3) : (2) で求めた和を 11 で割って余りを求める。

$$18 \div 11 = 1 \cdots 7$$

(4) : (3) で求めた余りの数字を検査文字とする。余りが 10 の時は、"X" を検査文字とする。

余りの「7」が検査文字になる

14-7 モジュールの分割〔解答・解説〕

問 1 ウ

〔解説〕モジュール強度が最も高いのはモジュール内のすべての要素が一つの機能を実行するために関連しあっている機能的強度である。

問 2 ア

〔解説〕ア 情動的強度をもつモジュール(最もモジュール強度の高い設計)
イ 時間的強度をもつモジュール
ウ 論理的強度をもつモジュール
エ 連絡的強度をもつモジュール

問 3 ウ

〔解説〕アは共通結合、イは制御結合、エは外部結合の説明である。

問 4 ウ

〔解説〕引数を共通域におき共有すれば、ある関数を変更したときに、他の関数に影響を及ぼす可能性が高くなる

問 5 イ

〔解説〕ア モジュール強度に関する説明で、モジュール結合度とは直接的な関係ない
ウ 外部結合に関する説明
エ 制御結合に関する説明

問 6 イ

〔解説〕ア 外部結合
イ データ結合
ウ 共通結合
エ スタンプ結合
結合度の強弱は以下の通り
[弱い] データ結合 スタンプ結合 制御結合 外部結合 共通結合 内容結合 [強い]

14-8 テスト〔解答・解説〕

問 1 イ

問 2 ウ

〔解説〕ブラックボックステストは、入力と出力だけに着目して様々な入力に対して仕様書どおりの出力が得られるかどうかを確認していく、システムの内部構造とは無関係に外部から見た機能について検証するテスト方法。

問 3 イ

〔解説〕ア 有効なデータだけでなく、無効なデータがエラーとして処理されるかも検証する
ウ 発生頻度ではなく入力データの範囲を分析してテストデータを作成する
エ プログラムの内部構造に着目しない

問 4 ウ

〔解説〕 ホワイトボックステストは、プログラムの内部構造に注目して行われるテストで、制御の流れの正当性を検証する目的で行われる。

問 5 ウ

〔解説〕 ア、イ、エはブラックボックステストの説明である。

問 6 エ

〔解説〕 テストカバレッジ分析とは、テストのカバレッジ（網羅率）を分析する手法であり、命令網羅、分岐網羅、条件網羅などがある。

問 7 ア

〔解説〕 イ 実際に実験を行うのではなくコンピュータプログラム上で模擬試験をするためのツール
ウ 命令単位でプログラム実行し、実行直後のレジスタやメモリの内容などの情報が得られるツール
エ HTML ファイルを解析し Web ページを表示するソフトウェア

問 8 エ

〔解説〕 上位モジュールの機能を代行するテスト用プログラムをドライバ、下位モジュールの機能を代行するテスト用プログラムをスタブという。

問 9 エ

〔解説〕 ア スタブは、テスト対象の上位モジュールからの呼び出しに対して適切な値を返却する
イ テスト対象モジュールを呼び出すのはドライバ
ウ ドライバは、テスト対象の下位モジュールを呼び出すモジュール

問 10 ア

〔解説〕 イ ダイナミックテストは、プログラムを動的な振る舞いを検証するための動的テストツール
ウ デバッガは、デバッグ作業においてバグの発見や訂正を支援するソフトウェア
エ ドライバは、下位モジュールから検証を開始するボトムアップテストにおいて、未完成の上位モジュールの代わりとなるテスト用モジュール

問 11 イ

〔解説〕 回帰テスト(リグレッションテスト)は、情報システムに変更作業を実施した場合に、それによって以前まで正常に機能していた部分に不具合や影響が出ていないかを検証するテストです。退行テストとも呼ぶ。

問 12 ウ

〔解説〕 リグレッションテストともいう。

問 13 ウ

〔解説〕 運用テストとは、開発したシステムを本番と同じ動作環境と運用体制で稼働させ、テストする方法であり、エンドユーザが主に担当する。

問 14 イ

〔解説〕 ユーザ部門が優先して確認すべき事項なので、システム開発を依頼した側としての視点で検証することになる。つまり、依頼どおりにできていること、すなわち決められた業務手順どおりに稼働することの確認になる。

問 15 エ

〔解説〕 ア 共通データ領域を使用するとモジュール結合度が上がるので、できるだけ使用しないことが望ましい
イ 結合度が高いものだけでなく低いものも含め、ソフトウェア全体でどのように分布しているかで判断する
ウ 結合度の観点から考えると共通データ領域の使用はベストではない
エ 正解。最も結合度が弱くなるのは単一のデータ項目の受け渡しのみの関連である

問 16 ア

〔解説〕 ストレステスト（負荷テスト）とは、システムの処理限界になるような高い負荷をかけ、正常に動作するかを検証するテストである。
イはユーザビリティテスト、ウはベンチマークテスト、エはリグレッションテストの説明である。

問 17 ア

〔解説〕 システム適格性確認テストは、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を用いてシステムを検証するテストである。
イは結合テスト、ウはホワイトボックステストの判定条件網羅、エはホワイトボックステストの命令網羅で用意しておくべきテストデータ。

問 18 ウ

〔解説〕 ア 命令網羅では、すべての命令を実行するようテストケースを設計する
イ 判定条件網羅では、条件の真と偽を実行するようテストケースを設計する
ウ 複数条件網羅では、複合条件それぞれの真と偽すべての組合せを実行するようテストケースを設計する（正解）
エ 条件網羅では、複合条件それぞれの真と偽を実行するようテストケースを設計する

問 19 イ

〔解説〕 $A = 0$ 、 $B = 0$ 、 $A = 1$ 、 $B = 1$ に関して、2つの判断記号両方で No になってしまうので、どちらか一つあれば十分である。
また、 $A = 1$ 、 $B = 0$ は、2つの判断記号両方で Yes となる。よって、イが適切。

問 20 ア

〔解説〕 潜在エラーのうちテストで発見されたエラー数は、次の式で表すことができる。

$$\begin{aligned} & \text{発見された総エラー数} - \text{埋め込まれたエラーのうち発見されたエラー数} \\ & = n - m \end{aligned}$$

埋め込まれたエラーの発見率 $\frac{m}{S}$ と潜在エラーの発見率 $\frac{n-m}{T}$ は同程度と考えられるので、

$$\frac{m}{S} = \frac{n-m}{T} \text{ の関係式が成り立つ。}$$

問 21 イ

〔解説〕 判定条件網羅(分岐網羅)とは、
「プログラム中の判定条件で結果が真となる場合、偽となる場合を少なくとも 1 回は実行する」
ようにテストケースを設計するですと。
分岐先の各処理を最低 1 回実行できれば良い。

問 22 ア

〔解説〕 命令網羅では、すべての命令を少なくとも 1 回は通るようにテストする。
命令「X = 1」, 「Y = 1」, 「X = 2」, 「Y = 2」のすべてを少なくとも 1 回は通るようにテストするためには、「a と b の両方の値が 0」というテストケースと、「a と b のどちらの値も 0 ではない」というテストケースを使用すればよい。

問 23 ア

〔解説〕 テスト項目の消化件数はテストの実施状況を表すので、進捗管理の指標として使用できる。

問 24 イ

〔解説〕 それぞれのテストで検出されたバグ数を N_A , N_B 、共通のバグ数を N_{AB} とすると、総バグ数は以下の公式で求めることができる。

$$\text{総バグ数} = (N_A \times N_B) / N_{AB}$$

設問の数値をこの公式に代入すると、

$$\text{総バグ数} = (30 \times 40) / 20 = 60 \text{ (個)}$$

問25 ウ

〔解説〕 ア ソフトウェアの品質に問題があり、上流工程から見直すべきである
イ テスト方法に問題がないか見直すべきである
エ ほぼ正確に予測できるとはいえない

問 26 ア

〔解説〕 未消化テスト項目数が減っておらず、バグ摘出累計も頭打ちになっていることから、解決困難なバグに直面していると考えられる。

問 27 ア

〔解説〕 イ トップダウンテストは、システムの核となる上位モジュールを先にテストするので、全体に影響する致命的な不具合を早期に発見しやすい
ウ ホワイトボックステストの説明です。ブラックボックステストは、システムの入出力の関係に注目して行われるテスト
エ プログラムテストは、プログラム中の欠陥（バグ）をできる限り多く発見することを目標として行われる。欠陥が存在することは証明できるが、欠陥が存在しないことは証明できない

問 28 イ

〔解説〕 静的解析ツールとは、ソースプログラムを解析することによって、プログラムのバグなどを発見するツールのことである。

ア 関数ごとの実行処理時間は動的なデバッグツールで計測する
ウ 処理の記述漏れは、機能テストにより検出する
エ テストケースで実行されなかった命令は動的なデバッグツールで検出する

問 29 ウ

〔解説〕 ソフトウェアの品質が高いということは、すなわち、欠陥が少ないということなので、レビューを十

分に行ってできる限りのバグを検出することが、ソフトウェア品質の管理指標の一つとなる。

問30 ウ

- 〔解説〕
- ア 開発者が主体となりエラーの早期発見を目的としてプログラムのステップごとにシミュレーションを行いながら確認をしていくレビュー手法
 - イ ソフトウェアを実際に動かすことなく、仕様書やプログラムを人間の目で見えて検証するレビュー手法
 - エ 退行テスト／回帰テストとも呼ばれ、システムに変更作業を実施した場合に、それによって以前まで正常に機能していた部分に不具合や影響が出ていないかを検証するテスト

問31 ア

- 〔解説〕
- イ 結合テスト工程で実施するテスト
 - ウ 単体テスト工程で実施するテスト
 - エ レグレッションテストは、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テスト

問32 ウ

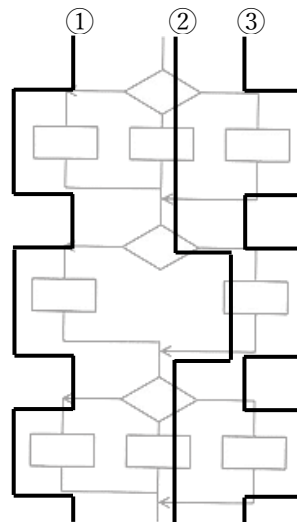
- 〔解説〕
- ア 内部状態が変化するシステムのテストで用いられる
 - イ 単体テストや結合テストで用いられるホワイトボックステストの手法
 - エ 状態遷移図、状態遷移表を使用する

問33 エ

- 〔解説〕
- ア バグの摘出数がテスト終盤になっても収束していません。まだ潜在バグが多数あると判断できる
 - イ 縦軸は累積バグ数を表しています。累積バグ数はテストの進行に伴い増加していくためグラフは不適切
 - ウ バグの摘出数がテスト終盤になっても収束していません。まだ潜在バグが多数あると判断できる

問 34 ア

- 〔解説〕命令網羅は、すべてのプログラム中の全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とで、問題文の流れ図のテストを命令網羅で実行すると以下のようになり、テストケースが最低3つあれば全ての命令を検証することができる。



Chapter14 システム開発

14-1 システムを開発する流れ（解答・解説）

問 1 イ

〔解説〕アは運用プロセス，ウ，エは要件定義プロセスで実施する。

問 2 ウ

〔解説〕ア，イは要件定義プロセス，エは開発プロセスで定義する。

問 3 ア

〔解説〕イは要件定義プロセス，ウは問題解決プロセス，エは文書化プロセスの説明である。

問 4 イ

〔解説〕ア，エは要件定義プロセス，ウは開発プロセスの目的である。

問 5 ウ

〔解説〕アは要件定義プロセス，イは運用プロセス，エは開発プロセスで実施する。

問 6 エ

〔解説〕ア 誤り。 監査プロセスでは、客観性及び独立性を保証された立場から、成果物及びプロセスが要求事項、計画及び合意に適合しているかどうかを判定する。

イ 誤り。 検証プロセスでは、要求事項を満たしているかを確認する。

ウ 誤り。 使用性向上プロセスでは、使いやすさや使い勝手を確認する。

問 7 イ

〔解説〕ア 自社の社員で開発する必要はない

ウ 他社と同じシステムにする必要はない

エ テスト計画，運用マニュアル及び障害対策は計画の立案時には不要である

問 8 ウ

〔解説〕アはS L A，イはR F P，エはシステム仕様書である。

問 9 イ

〔解説〕FF 関係 (Finish-to-Finish, 終了-終了) :

「受付が終了したら試験会場への入場を終了する」というように、あるアクティビティの終了が、他方のアクティビティの終了条件になっている関係。

FS 関係 (Finish-to-Start, 終了-開始) :

「受付が終了してから試験を開始する」というように、あるアクティビティの終了が、他方のアクティビティの開始条件になっている関係。

SF 関係 (Start-to-Finish, 開始-終了) :

「試験が開始されたら受付を終了する」というように、あるアクティビティの開始が、他方のアクティビティの終了条件になっている関係。

SS 関係 (Start-to-Start, 開始-開始) :

「受付が開始されたら試験会場への入場を開始する」というように、あるアクティビティの開始が、他方のアクティビティの開始条件になっている関係。

問 10 ア

〔解説〕イ RFP(Request for Proposal, 提案依頼書)の説明です。

ウ RFC(Request for Change, 変更要求書)の説明です。

エ 双方の役割分担を確認する文書はSOW(Statement of Work, 作業範囲記述書)と呼ばれるが、契約締結を要請するという意味に当てはまる用語はない。

問 11 イ

〔解説〕アは内部設計, ウはプログラム設計, エは外部設計で行う作業である。

問 12 ア

〔解説〕イは開発プロセス, ウは企画プロセス, エは運用プロセス段階で行う作業である。

問 13 ア

〔解説〕イは運用プロセス, ウは企画プロセス, エは開発プロセスで実施すべきである。

問 14 ウ

〔解説〕トレーサビリティとは、対象とする物品（製品や情報）の履歴を確認できることである。

問 15 エ

〔解説〕ア "利害関係者の識別"で実施する内容

イ "要件の評価"で実施する内容

ウ "要件の合意"で実施する内容

問 16 イ

〔解説〕非機能要件とは、機能として明確に定義できない品質要件や技術要件、運用・操作要件などのことである。

問 17 ウ

- 〔解説〕 ア 保守承認者が承認を行ってからプログラムを登録する
イ 必ずテストを行ってから更新する
エ 保守承認者が承認を行った時点で完了とする

問 18 エ

- 〔解説〕 ア 企画プロセスで行う
イ システム開発プロセスで行う
ウ ソフトウェア実装プロセスで行う

問 19 エ

- 〔解説〕 非機能要件：業務要件を実現するために必要なシステムの機能要件以外の要件のことで制約条件や品質要求がこれに該当する。
ア 業務要件に該当。
イ 業務要件に該当。
ウ 機能要件に該当。

14-2、3 システムの開発手法（解答・解説）

問 1 イ

問 2 イ

- 〔解説〕 a は基本計画，b は内部設計，c はプログラミング，d はテスト，e はプログラム設計，f は外部設計の説明である。

問 3 エ

- 〔解説〕 テスト駆動開発：XP の基本となる 12 のプラクティスのうちの 1 つで、求める機能を明確化するために、プログラムを記述するよりも前にテストケースを作成する開発手法。そのテストをパスする最低限の実装を行った後で、機能を維持したままコードを洗練していくという手順で開発を進める。

問 4 イ

- 〔解説〕 リファクタリングとは、プログラム外部から見た動作を変えずに、修正や拡張が簡単になるようにソースコードの内部構造を変更することである。

問 5 ウ

- 〔解説〕 ア 「週40時間労働」に反する。
イ 「ソースコードの共同所有」に反する。
ウ 正解。「常に統合」のプラクティスに適合する。
エ 「ペアプログラミング」に反する。

問 6 ウ

- 〔解説〕 X P（Extreme Programing）とは、柔軟性を重視したソフトウェア開発手法で、ペアプログラミング（2人のプログラマがペアを組み、コーディングを行うドライバと助言を行うナビゲータに分かれ、交代しながら作業を進める）を特徴としている。

問 7 エ

- 〔解説〕 ア 計画ゲームは、イテレーションごとに行われ、実現するユーザ機能(ストーリー)とリリース計画を顧客と交渉する作業
イ コーディング標準は、ソースコードを記述する際のルールを定めたもの
ウ テスト駆動開発は、求める機能を明確化するために、プログラムを記述するよりも前にテストケースを作成する開発手法

問 8 ア

- 〔解説〕 上流CASEツールの対象となるのは、基本設計・外部設計・内部設計。
下流CASEツールの対象となるのは、プログラム設計・プログラミング・テスト。

問 9 ウ

- 〔解説〕 SOA (Service Oriented Architecture) とは、ソフトウェアの機能を独立したサービスと見立て、それらを組み合わせ連携させることで、システム全体を構築していく手法である。

問 10 エ

- 〔解説〕 ア ERP(Enterprise Resource Planning)の説明です。
イ フィット & ギャップ分析の説明です。
ウ PDCA サイクルの説明です。

問 11 イ

- 〔解説〕 ア OSI 基本参照モデルの説明。
ウ SLM(Service Level Management)の説明。
エ SaaS(Software as a Service)の説明。

問 12 ウ

- 〔解説〕 ア フォワードエンジニアリングの説明。
イ リファクタリングの説明。
エ リエンジニアリングの説明。

問 13 エ

- 〔解説〕 ア スクライブの役割
イ オーナーの役割
ウ プレゼンタの役割

問 14 ア

- 〔解説〕 デザインレビューとは設計に対するレビューのことであり、仕様の不備や誤りの早期発見を目的として行う。

問 15 ア

- 〔解説〕 イ 開発者が主体となりエラーの早期発見を目的としてプログラムのステップごとにシミュレーションを行いながら確認をしていくレビュー手法
ウ 電子メールなどを使って、対象の成果物を複数のレビューに配布・回覧し、フィードバックを求めるレビュー方法
エ 二人一組でプログラムの実装を行い、一人が実際のコードをコンピュータに打ち込み、もう一人はそれをチェックしながらナビゲートするという役割を随時交代しながら作業を進める手法

問 16 ウ

- 〔解説〕 ア ウォークスルーの説明
イ ラウンドロビンの説明

14-4 業務のモデル化〔解答・解説〕

問 1 ウ

〔解説〕データ中心分析・設計技法は、システム設計の上流工程において、データの流れやその分析に重点をおく手法であり、モデル化に際しては実際の情報資源（データ）に着目する。

問 2 エ

〔解説〕アはフローチャート、イは状態遷移図、ウはE-R図の説明である。

問 3 ウ

問 4 エ

〔解説〕ア 2本の平行線はデータストア（ファイル）を意味するので、ファイル名などを付ける
イ 円はプロセス（処理）を意味するので、処理名を付ける
ウ 四角はデータの源泉と吸収を意味するので、ユーザ名や部門名などを付ける

問 5 エ

〔解説〕DFD(Data Flow Diagram)は、構造化分析手法の一つでデータの流れに着目して、その対象となる業務のデータの流れと処理の関係をわかりやすく図式化する手法。

問 6 ア

問 7 ア

〔解説〕イ (誤)状態遷移→(正)リレーションシップ
ウ インスタンスとはエンティティの実現値である
(例えば、エンティティがプログラム言語ならインスタンスはC, VB, JAVAなどとなる)
エ 抽象的な概念もエンティティになりうる

問 8 イ

〔解説〕ア 関係データベースの実装を前提としていない
ウ DFDの説明である
エ リレーションシップはエンティティ間の関連を表現する

問 9 ウ

〔解説〕アはUML、イは状態遷移図、エはDFDの説明である。

問 10 ア

〔解説〕イ エンティティタイプ間の関連を表す線には「——— : 1対1」「———→ : 1対多」
「←——→ : 多対多」という意味がある
ウ DFD(Data Flow Diagram)に関する記述
エ リレーションシップは、実体がもつ関連を表す

問 11 エ

〔解説〕状態遷移図とは、状態変化の契機となる事象とそれに伴う動作を記述するものであり、時間の経過や状況の変化に基づくリアルタイム制御システムの設計に適している。

問 12 エ

〔解説〕投入される硬貨は、10円、50円、100円のいずれかであり、投入による遷移は以下のようになる。

- Q0状態から10円投入でQ1、50円投入でQ5、100円投入でEに遷移
- Q1状態から10円投入でQ2、50円投入でQ6、100円投入でEに遷移
- Q2状態から10円投入でQ3、50円又は100円投入でEに遷移
- Q3状態から10円投入でQ4、50円又は100円投入でEに遷移
- Q4状態から10円投入でQ5、50円又は100円投入でEに遷移

問 13 ウ

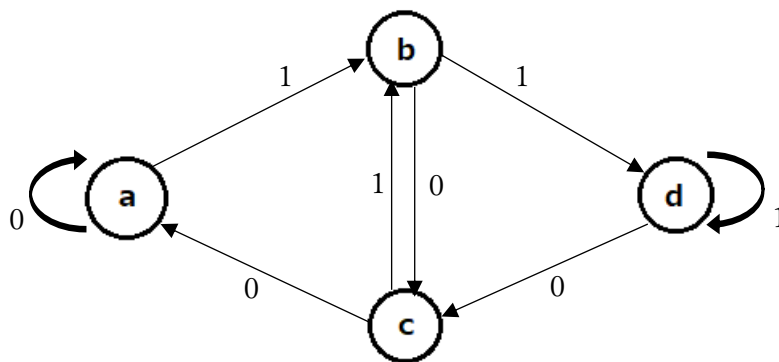
〔解説〕設問の有限オートマトンは初期状態で1が入力されると受理状態へ遷移するので文の最後は1になる。受理する過程ですが以下の2つのパターンがある。

- ・初期状態で0を繰り返して1で受理状態に遷移する
- ・初期状態と受理状態を1→0で繰り返して1で受理状態に遷移する

0または10を0回以上繰り返した後に1で終了する文字列なので、「 $(0|10)^*1$ 」が適切な正規表現。他の正規表現は少なくとも、初期状態で0を繰り返した後に1で受理する $000...01$ というパターンにマッチしないので誤りと判断できる。

問 14 ウ

〔解説〕表の有限オートマトンを図にすると次のようになる。



ビット列「110」が入力されるとき、最後の「0」が入力されて遷移するのはaかc。
ビット列2番目の「1」が入力されている状態を踏まえるとcが適切。

問 15 イ

〔解説〕0が入力されたとき、出力は0か1かのいずれかなので、 $a + b = 1$
1が入力されたとき、出力は0か1かのいずれかなので、 $c + d = 1$

問 16 ウ

- 〔解説〕
- ア 矢印の○から始め、0で矢印の○に戻る。この後、0を4回繰り返しても、◎に行けない。
 - イ 0で矢印の○に戻る。次の1で次の○に行く。次の1で同じ○を繰り返すため、◎に行けない。
 - ウ ○→1で次の○に行く。0で◎に行く。1で◎に戻る。0で◎に戻る。◎にいるため正解。
 - エ 1で次の○に行く。1で次の○に戻る。この後、1を2回繰り返しても、◎に行けない。

問 17 ウ

〔解説〕この有限オートマトンを文章で表すと、

1. S1 状態から始まる。
2. S1 状態で 0 を出力した時は S3 へ、1 を出力した時は S2 へ状態遷移する。
3. S2 状態で 0 を出力した時は S2 を維持、1 を出力した時は S1 へ状態遷移する。
4. S3 状態で 0 を出力した時は S2 へ状態遷移し、1 を出力した時は S3 を維持する。
5. 入力列は S3 で受理される。

ということになります。

選択肢の中で唯一 S3 が受理可能な入力列は「1101」。

問 18 ア

〔解説〕イ 親組織から見た子組織の多重度は 0 以上なので、子組織が存在しない場合もある

ウ 子組織が親組織となり 3 以上の階層構造となる場合もあるので誤り

エ 1 つの親組織は複数の子組織を持つことができ、1 つの子組織は複数の親組織を持つことができるので関連は網状のネットワーク構造になる

14-5 ユーザインタフェース〔解答・解説〕

問 1 ウ

〔解説〕ア 入力項目が一定でない場合、直接入力ができなければ使いづらい

イ 初心者のことを考慮すると、ダブルクリックは避けるべきである

エ 入力しやすさの観点から、入力原票と同じフォーマットにするのが望ましい

問 2 エ

〔解説〕ア 回顧法は、被験者にタスクを実行してもらい、専門家がその行動を観察し、事後の質問への回答とともに分析する手法

イ 被験者にタスクを実行してもらうのは回顧法と同様だが、思考発話法ではタスク実行時に操作方法を認知する過程をその都度、発話してもらう

ウ 認知的ウォークスルー法は、使いながら操作を理解していく際の認知モデルである探索学習理論に基づいた評価手法。まず対象とするユーザ像を設定し、ユーザの行動をシミュレーションしていくことで問題点を明らかにしていく

問 3 エ

〔解説〕パンくずリストは、比較的大規模な Web サイトで、利用者が現在どこの階層のページにいるのかをユーザに知らせ、現在位置を見失わないようにする目的で設置されるユーザインタフェース(U I)部品。

問 4 エ

〔解説〕E U C (End User Computing) とは、情報システムを利用して業務を行なうユーザが、自らシステムの構築や運用・管理に積極的に携わることである。

問 5 ウ

〔解説〕期間ごと、購入金額ごとの優良顧客数が求めやすいものがよい

ア 期間ごとの優良顧客数がわかりにくい

イ 金額ごとの優良顧客数がわかりにくい

エ 期間と購入回数との関連がわかりにくい

問 6 イ

〔解説〕 アクセシビリティとは、高齢者や障害者を含む誰もが利用できる環境であり、音声による文章の読み上げや、手書き文字による入力が行えるようにすることなどがある。

問 7 エ

〔解説〕 ユーザビリティとは使いやすさのことであり、ユーザがいかにストレスやとまどいを感じずに目標を達成できるかという度合いを指す。

問 8 エ

〔解説〕 ア ユニバーサルデザインの説明
イ アクセシビリティの説明
ウ 情報バリアフリーの説明

問 9 ウ

〔解説〕 デジタルディバイドとは、IT技術を使いこなすことができる者とできない者との待遇や貧富の格差が広がることをいう。

問10 ウ

〔解説〕 ユーザビリティとは「利用者にとっての使いやすさ」のこと。
5つの評価手法を"利用者の立場から"と"専門家の立場から"に分類すると次のようになる。

〔利用者の立場からの評価手法〕

アンケート、回顧法、思考発話法

〔専門家の立場からの評価手法〕

認知的ウォークスルー法、ヒューリスティック評価法

よって、正しい組み合わせは「ウ」となる。

	利用者の立場からの評価手法		専門家の立場からの評価手法	
ア	アンケート	利用者	回顧法	利用者
イ	回顧法	利用者	思考発話法	利用者
ウ	思考発話法	利用者	ヒューリスティック評価法	専門家
エ	認知的ウォークスルー法	専門家	ヒューリスティック評価法	専門家

問11 ウ

〔解説〕 アクセシビリティとは、高齢者・障害者を含む誰もが、さまざまな製品や建物やサービスなどを支障なく利用できるかどうか、あるいはその度合いのこと。

Webページにおいては、そのページが、高齢者や障害者も含めた、誰もが情報を取得・発信できるように柔軟に設計されていて、アクセスした誰もが同様に情報を共有できるか、あるいはその度合いを意味する。

問12 イ

〔解説〕 ア アクセシビリティとは、等しい水準のユーザビリティをすべての個人について達成することではなく、少なくともある程度のユーザビリティをすべての個人について達成すること
ウ アクセシビリティ設計は、一般的な人間工学指針に従うことが重要
エ アクセシビリティ設計の対象には、平均的能力の人々だけでなく、様々な障害をもつ人々を含む最も幅広い層の人々を含む

問13 ウ

14-6 コード設計と入力のチェック〔解答・解説〕

問 1 ア

〔解説〕チェックディジットとは、入力誤りの検出のため、コードの各桁の値から計算された値をコードの末尾に付加するものである。

問 2 イ

〔解説〕チェックディジットチェックは、コードの各桁の値から計算された値をコードの末尾に付加するものである。入力の際には同じ計算を行い、結果が異なっていれば入力誤りであることがわかる。

問 3 ウ

〔解説〕 $(4 \times 3 + 2 \times 2 + \square) \div 7$ の余りが 6 であるから、 \square は 4

問 4 ウ

〔解説〕 $N_1 \times 1 + 7 \times 2 + 6 \times 3 + 2 \times 4 = N_1 + 40$ を 10 で割った余りが 4 となるのだから、 N_1 の値は 4 となる。

問 5 イ

〔解説〕アはチェックディジットチェック、ウはバランスチェック、エは重複チェックの説明である。

問 6 ウ

〔解説〕ア 区分コードは、ブロックコードとも言い、営業部は 1000～1999、企画部は 2000～2999 というようにグループごとに定められた範囲内でコードを割り当てる方式。上位桁を見るだけで、どのグループに属しているのか判断できる特徴がある。

イ 桁別コードは、各桁（または桁のグループ）に分類上の意味を持たせたコード体系。JAN コード、ISBN コード、運転免許証の番号などが桁別コードに該当。

エ 連番コードは、シーケンスコードとも言い、0001→0002→0003→…→9999 というように連続した値をデータに 1 つずつ割り当てていく方法。

問 7 エ

〔解説〕4 けたの数値データ "2 1 3 1" に設問に与えられた〔規則〕を適用すると以下のようになる。

(1)：与えられたデータの各けたに、先頭から係数 4, 3, 2, 1 を割り当てる。

$$2(4), 1(3), 3(2), 1(1)$$

(2)：各けたの数値と割り当てた係数とのそれぞれの積の和を求める。

$$2 \times 4 + 1 \times 3 + 3 \times 2 + 1 \times 1 = 18$$

(3)：(2)で求めた和を 11 で割って余りを求める。

$$18 \div 11 = 1 \cdots 7$$

(4)：(3)で求めた余りの数字を検査文字とする。余りが 10 の時は、"X"を検査文字とする。

余りの「7」が検査文字になる

14-7 モジュールの分割〔解答・解説〕

問 1 ウ

〔解説〕モジュール強度が最も高いのはモジュール内のすべての要素が一つの機能を実行するために関連している機能的強度である。

問 2 ア

〔解説〕ア 情動的強度をもつモジュール(最もモジュール強度の高い設計)
イ 時間的強度をもつモジュール
ウ 論理的強度をもつモジュール
エ 連絡的強度をもつモジュール

問 3 ウ

〔解説〕アは共通結合、イは制御結合、エは外部結合の説明である。

問 4 ウ

〔解説〕引数を共通域におき共有すれば、ある関数を変更したときに、他の関数に影響を及ぼす可能性が高くなる

問 5 イ

〔解説〕ア モジュール強度に関する説明で、モジュール結合度とは直接的な関係ない
ウ 外部結合に関する説明
エ 制御結合に関する説明

問 6 イ

〔解説〕ア 外部結合
イ データ結合
ウ 共通結合
エ スタンプ結合
結合度の強弱は以下の通り
[弱い] データ結合 スタンプ結合 制御結合 外部結合 共通結合 内容結合 [強い]

14-8 テスト〔解答・解説〕

問 1 イ

問 2 ウ

〔解説〕ブラックボックステストは、入力と出力だけに着目して様々な入力に対して仕様書どおりの出力が得られるかどうかを確認していく、システムの内部構造とは無関係に外部から見た機能について検証するテスト方法。

問 3 イ

〔解説〕ア 有効なデータだけでなく、無効なデータがエラーとして処理されるかも検証する
ウ 発生頻度ではなく入力データの範囲を分析してテストデータを作成する
エ プログラムの内部構造に着目しない

問 4 ウ

〔解説〕ホワイトボックステストは、プログラムの内部構造に注目して行われるテストで、制御の流れの正当性を検証する目的で行われる。

問 5 ウ

〔解説〕ア、イ、エはブラックボックステストの説明である。

問 6 エ

〔解説〕テストカバレッジ分析とは、テストのカバレッジ（網羅率）を分析する手法であり、命令網羅、分岐網羅、条件網羅などがある。

問 7 ア

〔解説〕イ 実際に実験を行うのではなくコンピュータプログラム上で模擬試験をするためのツール
ウ 命令単位でプログラム実行し、実行直後のレジスタやメモリの内容などの情報が得られるツール
エ HTML ファイルを解析し Web ページを表示するソフトウェア

問 8 エ

〔解説〕上位モジュールの機能を代行するテスト用プログラムをドライバ、下位モジュールの機能を代行するテスト用プログラムをスタブという。

問 9 エ

〔解説〕ア スタブは、テスト対象の上位モジュールからの呼び出しに対して適切な値を返却する
イ テスト対象モジュールを呼び出すのはドライバ
ウ ドライバは、テスト対象の下位モジュールを呼び出すモジュール

問 10 ア

〔解説〕イ ダイナミックテストは、プログラムを動的な振る舞いを検証するための動的テストツール
ウ デバッガは、デバッグ作業においてバグの発見や訂正を支援するソフトウェア
エ ドライバは、下位モジュールから検証を開始するボトムアップテストにおいて、未完成の上位モジュールの代わりとなるテスト用モジュール

問 11 イ

〔解説〕回帰テスト(リグレッションテスト)は、情報システムに変更作業を実施した場合に、それによって以前まで正常に機能していた部分に不具合や影響が出ていないかを検証するテストです。退行テストとも呼ぶ。

問 12 ウ

〔解説〕リグレッションテストともいう。

問 13 ウ

〔解説〕運用テストとは、開発したシステムを本番と同じ動作環境と運用体制で稼働させ、テストする方法であり、エンドユーザが主に担当する。

問 14 イ

〔解説〕 ユーザ部門が優先して確認すべき事項なので、システム開発を依頼した側としての視点で検証することになる。つまり、依頼どおりにできていること、すなわち決められた業務手順どおりに稼働することの確認になる。

問 15 エ

〔解説〕 ア 共通データ領域を使用するとモジュール結合度が上がるので、できるだけ使用しないことが望ましい
イ 結合度が高いものだけでなく低いものも含め、ソフトウェア全体でどのように分布しているかで判断する
ウ 結合度の観点から考えると共通データ領域の使用はベストではない
エ 正解。最も結合度が弱くなるのは単一のデータ項目の受け渡しのみの関連である

問 16 ア

〔解説〕 ストレステスト（負荷テスト）とは、システムの処理限界になるような高い負荷をかけ、正常に動作するかを検証するテストである。
イはユーザビリティテスト、ウはベンチマークテスト、エはリグレッションテストの説明である。

問 17 ア

〔解説〕 システム適格性確認テストは、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を用いてシステムを検証するテストである。
イは結合テスト、ウはホワイトボックステストの判定条件網羅、エはホワイトボックステストの命令網羅で用意しておくべきテストデータ。

問 18 ウ

〔解説〕 ア 命令網羅では、すべての命令を実行するようテストケースを設計する
イ 判定条件網羅では、条件の真と偽を実行するようテストケースを設計する
ウ 複数条件網羅では、複合条件それぞれの真と偽すべての組合せを実行するようテストケースを設計する（正解）
エ 条件網羅では、複合条件それぞれの真と偽を実行するようテストケースを設計する

問 19 イ

〔解説〕 $A = 0$ 、 $B = 0$ 、 $A = 1$ 、 $B = 1$ に関して、2つの判断記号両方で **N o** となってしまうので、どちらか一つあれば十分である。
また、 $A = 1$ 、 $B = 0$ は、2つの判断記号両方で **Y e s** となる。よって、イが適切。

問 20 ア

〔解説〕 潜在エラーのうちテストで発見されたエラー数は、次の式で表すことができる。

$$\begin{aligned} & \text{発見された総エラー数} - \text{埋め込まれたエラーのうち発見されたエラー数} \\ & = n - m \end{aligned}$$

埋め込まれたエラーの発見率 $\frac{m}{S}$ と潜在エラーの発見率 $\frac{n-m}{T}$ は同程度と考えられるので、

$$\frac{m}{S} = \frac{n-m}{T} \text{ の関係式が成り立つ。}$$

問 21 イ

〔解説〕判定条件網羅(分岐網羅)とは、
「プログラム中の判定条件で結果が真となる場合、偽となる場合を少なくとも 1 回は実行する」
ようにテストケースを設計するですと。
分岐先の各処理を最低 1 回実行できれば良い。

問 22 ア

〔解説〕命令網羅では、すべての命令を少なくとも 1 回は通るようにテストする。
命令「X = 1」, 「Y = 1」, 「X = 2」, 「Y = 2」のすべてを少なくとも 1 回は通るようにテストするためには、「a と b の両方の値が 0」というテストケースと、「a と b のどちらの値も 0 ではない」というテストケースを使用すればよい。

問 23 ア

〔解説〕テスト項目の消化件数はテストの実施状況を表すので、進捗管理の指標として使用できる。

問 24 イ

〔解説〕それぞれのテストで検出されたバグ数を N_A 、 N_B 、共通のバグ数を N_{AB} とすると、総バグ数は以下の公式で求めることができる。

$$\text{総バグ数} = (N_A \times N_B) / N_{AB}$$

設問の数値をこの公式に代入すると、

$$\text{総バグ数} = (30 \times 40) / 20 = 60 \text{ (個)}$$

問 25 ウ

〔解説〕ア ソフトウェアの品質に問題があり、上流工程から見直すべきである
イ テスト方法に問題がないか見直すべきである
エ ほぼ正確に予測できるとはいえない

問 26 ア

〔解説〕未消化テスト項目数が減っておらず、バグ摘出累計も頭打ちになっていることから、解決困難なバグに直面していると考えられる。

問 27 ア

〔解説〕イ トップダウンテストは、システムの核となる上位モジュールを先にテストするので、全体に影響する致命的な不具合を早期に発見しやすい
ウ ホワイトボックステストの説明です。ブラックボックステストは、システムの入出力の關係に注目して行われるテスト
エ プログラムテストは、プログラム中の欠陥（バグ）をできる限り多く発見することを目標として行われる。欠陥が存在することは証明できるが、欠陥が存在しないことは証明できない

問 28 イ

〔解説〕静的解析ツールとは、ソースプログラムを解析することによって、プログラムのバグなどを発見するツールのことである。

ア 関数ごとの実行処理時間は動的なデバッグツールで計測する
ウ 処理の記述漏れは、機能テストにより検出する
エ テストケースで実行されなかった命令は動的なデバッグツールで検出する

問 29 ウ

〔解説〕 ソフトウェアの品質が高いということは、すなわち、欠陥が少ないということなので、レビューを十分に行ってできる限りのバグを検出することが、ソフトウェア品質の管理指標の一つとなる。

問30 ウ

- 〔解説〕
- ア 開発者が主体となりエラーの早期発見を目的としてプログラムのステップごとにシミュレーションを行いながら確認をしていくレビュー手法
 - イ ソフトウェアを実際に動かすことなく、仕様書やプログラムを人間の目で見えて検証するレビュー手法
 - エ 退行テスト／回帰テストとも呼ばれ、システムに変更作業を実施した場合に、それによって以前まで正常に機能していた部分に不具合や影響が出ていないかを検証するテスト

問31 ア

- 〔解説〕
- イ 結合テスト工程で実施するテスト
 - ウ 単体テスト工程で実施するテスト
 - エ レグレッションテストは、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テスト

問32 ウ

- 〔解説〕
- ア 内部状態が変化するシステムのテストで用いられる
 - イ 単体テストや結合テストで用いられるホワイトボックステストの手法
 - エ 状態遷移図、状態遷移表を使用する

問33 エ

- 〔解説〕
- ア バグの摘出数がテスト終盤になっても収束していません。まだ潜在バグが多数あると判断できる
 - イ 縦軸は累積バグ数を表しています。累積バグ数はテストの進行に伴い増加していくためグラフは不適切
 - ウ バグの摘出数がテスト終盤になっても収束していません。まだ潜在バグが多数あると判断できる

問 34 ア

〔解説〕 命令網羅は、すべてのプログラム中の全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とで、問題文の流れ図のテストを命令網羅で実行すると以下のようになり、テストケースが最低 3 つあれば全ての命令を検証することができる。

