

第4回中級ソフトウェア品質技術者資格試験記述式問題の解説

※ここで解説している問題は、出題したすべての問題ではありません。特に正答率が低かった問題について解説しています。

※中級ソフトウェア品質技術者資格試験の記述式問題の採点においては、唯一の正解との適合のみをみるのではなく、受験者の意図を読み取って採点しています。

■穴埋め問題

空欄（ ）に入る適切な語句を答案用紙の該当箇所に解答せよ。

【問題】 品質の概念

「ISO/IEC 9126 ソフトウェア製品の品質」において、ソフトウェア製品品質に対し「外部および内部品質のための品質モデル」、および「（ ① ）のための品質モデル」の 2 つのモデルを規定している。外部および内部品質のための品質モデルは 6 つの品質特性とさらにそれぞれの特性を細分化した（ ② ）で表現している。（ ① ）のための品質モデルは 4 つの品質特性で表される。

【答案用紙】 実際の答案用紙のスタイルです（全 10 問）。

	①	②		②	②
問題 26			問題 31		
問題 27			問題 32		
問題 28			問題 33		
問題 29			問題 34		
問題 30			問題 35		

【解答例】

①利用時の品質

②品質副特性

【解説】

本問題は、品質特性、品質副特性というキーワードを知っていること、及び利用時の品質のための品質モデルが規定されたことを知っていることを確認する問題である。

ISO/IEC 9126 シリーズは、ソフトウェア製品の品質に関して規定されている国際規格である。ソフトウェアの品質は、ソフトウェアの「誤りの有無」のみで評価するのではなく、ユーザのニーズを満たすために、様々な視点のソフトウェア品質特性から評価することが大切である。この規格では、ソフトウェア品質特性、副特性からなる品質モデルとそれぞれの副特性を定量的に測定するメトリクスを示している。

ISO/IEC 9126-1 ではソフトウェア製品の品質に対して、「外部及び内部品質」と「利用時の品質」の 2 つのモデルを規定している。

ソフトウェアの外部品質及び内部品質モデルは、図 1 で示すように、ソフトウェアの外部及び内部の品質を 6 つの品質特性とさらにそれぞれの特性を細分化した品質副特性で表している。利用時の品質のための品質モデルは、図 2 で示すように、ソフトウェアの品質を 4 つの品質特性で表している。

外部品質は、外部の視点から見たソフトウェア製品の品質であり、ソフトウェアを実行して測定する。また、内部品質は、開発中の中間成果物（モデル、文書、ソースコードなど）の品質を示している。利用時の品質は、ある特定の環境で利用者が目標を達成できる程度を示している。

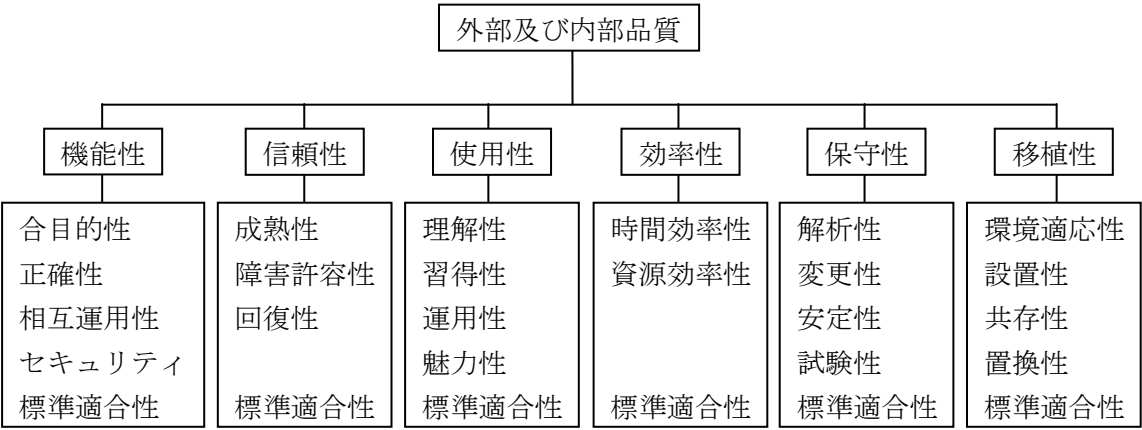


図 1 外部及び内部品質のための品質モデル

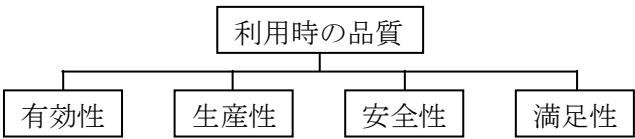


図 2 利用時の品質のための品質モデル

誤った解答の例としては、①に非機能特性、非機能要件、メトリクス、アクティビティ、プロセス、システム品質、安全、②に品質属性、従属特性と解答したものがあつた。

【問題】仕様記述の技法
ソフトウェアの仕様や振る舞いを、（ ① ）で記述するのではなく、厳密に規定された数学的な記法で定義する手法が（ ② ）である。最近では、モデル検証ツールを用いて、（ ② ）で記述された仕様の一貫性や正確性などを確認することができるようになった。

【解答例】

- ①自然言語、文章、言葉
- ②形式手法

【解説】

本問題は、レビューの技法のうち、仕様・コードに基づいた技法のひとつである形式手法の基本知識を問う問題である。形式手法とは、仕様上の矛盾を数学的な証明によって検出し、一貫性を保証する手法である。形式手法では、自然言語で定義された仕様書に見られる仕様の曖昧さ、不完全さ、誤りや矛盾を排除するために、システムの対象世界を集合論と論理的な記法で定義する。近年では、形式手法を適用したツールが登場しており、実用可能な技術となってきた。形式手法の詳しい内容までは求めないが、その特徴は理解してほしい。

誤った解答の例としては、②に「UML」と解答したものが目立った。レビュー技法にかかわらず、主要な技法の特徴と使用方法是理解してほしい。

【問題】テストのマネジメント

設計工程とテスト工程の対応関係を明確にすることにより、的確にテストできるよう工夫したプロセスモデルがある。（ ① ）は、ウォーターフォールモデルに基づいて、設計工程とテスト工程の対応関係を視覚的に表現することにより、各テスト工程のテストレベルを示したものである。（ ① ）の派生モデルである（ ② ）は、設計工程においてテスト設計などのテストのアクティビティを並行して行うプロセスモデルであり、テストの知見を開発工程の早期にフィードフォワードできるという利点がある。

【解答例】

- ①V字モデル（Vモデル でもよい）
- ②W字モデル（Wモデル でもよい）

【解説】

テストをマネジメントする上で、テストのアクティビティをいつ実施するか、テスト範囲をいかに設定するかは重要な観点である。それらの指針となるのがV字モデルやW字モデルであり、テストのマネジメントの基本知識である。その基本知識を保有しているかを問う問題である。

誤った解答の例としては、②に「スパイラルモデル」「テストファースト」「アジャイルモデル」などが見られた。

■説明問題

設問の指示に従って、答案用紙の該当箇所に解答せよ。

【問題】品質分析・評価の技法

結合テストを終えてシステムテストに入ろうとしている。モジュールあたりの障害（バグ）の密度を測定したところ、モジュールにより測定値が大きくばらついていた。品質の現状を把握するために層別をしたい。異なる 2 つの層別の観点と、それぞれの層別により明らかにできる事柄を各 25 字程度で述べよ。層別の観点と事柄の間は、『：』で区切って記述すること。

【答案用紙】実際の答案用紙のスタイルです。

注) 層別の観点と事柄の間は『：』で区切って記述してください。

・

・

【解答例】

次のような層別の観点とその観点に対応する事柄の組み合わせを 2 つ記述する。

・ 観点例 1「規模別（サイズ別）」

観点例 1 の事柄例 1「大きなモジュールに障害が集中しているか」

観点例 1 の事柄例 2「小さなモジュールに障害が集中しているか」

・ 観点例 2「開発担当者別」

観点例 2 の事柄例 1「同じプロセスでも担当者能力は異なっているか」

観点例 2 の事柄例 2「特定の担当者が多くの障害を作り込んでいるか」

観点例 2 の事柄例 3「熟練者の担当箇所は障害が少ないか」

・ 観点例 3「障害重大度別」

観点例 3 の事柄例「モジュールにより重大な障害が集中しているか」

・ 観点例 4「障害原因別（障害種別）」

観点例 4 の事柄例「障害が特定の障害原因に集中しているか」

・ 観点例 5「機能別（モジュールの役割別）」

観点例 5 の事柄例「複雑な機能に障害が集中しているか」

・観点例 6「テスト方法別」

観点例 6 の事柄例「テストの回数や実行有無が異なっているか」

・観点例 7「テスト担当者別」

観点例 7 の事例例 1 「同プロセスでも担当者能力は異なっているか」

観点例 7 の事柄例 2「特定担当者が軽微な障害を多数報告しているか」

観点例 7 の事柄例 3 「熟練者が経験により効果的にテストしているか」

【解説】

本問題は、品質を分析・評価するとき、層別という技法を適切に用いるかを問う問題である。

ばらつく測定情報を手にしたとき、ばらつきの原因を考察することは品質管理の基本である。QC 七つ道具のひとつである層別技法を、特定の目的のために適切に適用するためには、分析者の観点が決め手になる。明らかにしたい事柄が異なれば、層別の観点を変えなければいけない。このとき、ひとつの観点だけではなく、複数の観点によって、得られた情報をもとに多面的に分析を進めることが大切である。多面的な分析によって、ばらつきに大きく寄与している要因を浮彫りにすることができ、その結果、有効な対策につなげることができる。

【問題】 代用特性

詳細設計からテストの前までの間で、レスポンスタイムを要求値に近づけるべく管理したい。しかし、開発中なので性能を直接測れない。そこで、直接測る代わりに管理できそうな代用特性をあげて管理したい。異なる 2 つの代用特性と、それぞれが性能目標の達成に有効な理由を各 25 字程度で述べよ。代用特性と理由の間は、『:』で区切って記述すること。

【答案用紙】 実際の答案用紙のスタイルです。

注) 代用特性と理由の間は『:』で区切って記述してください。

[illegible]

【解答例】

次のような代用特性とその効果（性能目標の達成に有効な理由）の組み合わせを 2 つ記述する。

- 例 1 SQL 発行回数：(効果例 1) I/O 処理時間を減らす
(効果例 2) CPU 処理時間を減らす
- 例 2 I/O 回数：I/O 時間を減らす
- 例 3 スーパーバイザー・コール (SPV Call) 回数：CPU 処理時間を減らす
- 例 4 ダイナミックステップ数：CPU 走行時間を減らす
- 例 5 プログラムサイズ：(効果例 1) ページングオーバーヘッドを減らす
(効果例 2) ロード時間を減らす
(効果例 3) キャッシュヒット率を向上する
- 例 6 割込回数：割込処理のオーバーヘッドを減らす
- 例 7 伝送パケット数：伝送時間を減らす
- 例 8 伝送パケット長：伝送時間を減らす
- 例 9 関数の呼出し回数：CPU 走行時間を減らす

【解説】

本問題は、ソフトウェア開発過程で、出荷後のソフトウェア性能をどのように管理したらよいかを問う問題である。出荷後のソフトウェア性能を開発過程で管理することは難しい。一般には、性能目標を達成するために、ソフトウェアが完成してからチューニングする形を取りがちである。しかし、適切に設計されていないソフトウェアを後からチューニングするやり方では限界があり、極力避けたい。特に大規模、かつ将来的にも拡張が予見されるシステムでは、きちんと開発過程で性能管理をすることが大切である。

開発途上では、ソフトウェアの最終的な動作時間を直接測定できないので、開発過程で測定可能な代用特性を慎重に選び出し、代用特性にもとづいて管理していくことが重要である。

なお、本問題はレスポンスタイムを管理することが主題であるので、解答の際にレスポンスタイムとシステムのスループットを混同しないように注意が必要である。

■解説問題

設問の指示に従って、問題答案用紙の該当箇所に解答せよ。

【問題】

ソフトウェア設計において、プロジェクトのメンバーがこれまで経験したことの無い新技術に挑戦することとなった。この技術はソフトウェア開発の成否に関わる重要な技術である。もちろん、プロジェクト開始前に中核メンバーの何人かは、この技術研修を受講している。社内にこの技術を経験した技術者はいたが、このプロジェクトには参加できなかった。なお、この技術は数年前によく使われた技術の延長線上にあるが、適用方法を誤ると期待効果が得られない。

ソフトウェア設計のレビューにあたって注意すべきことを 3 つあげ、それぞれ 50 字程度で述べよ。

【答案用紙】 実際の答案用紙のスタイルです。

[illegible][illegible][illegible]

【解答例】

- ① 社内の新技術経験者をレビューに参加させ、技術の特徴や、適用上の注意点などを重点的に確認してもらう。
- ② 当該技術を過去に適用したときに期待効果が得られなかった原因を調べ、チェック項目として挙げる。
- ③ 既存技術と新技術との違いを調査し、流用可能な既存チェック項目や、新技術に固有のチェック項目を整理する。
- ④ 新技術に精通した社内の技術者がレビューに参加できない場合は、社外の有識者にレビューに協力してもらう。

【解説】

問題の狙い：

既存技術の延長上にある重要な新技術について、社内での技術蓄積が限られており、プロジェクトのメンバーには技術経験がないという状況における設計レビューの留意点を問う問題である。レビューとして適切な人員を選定し、当該技術を適用する上でのポイントを把握し、社内に蓄積された技術ノウハウを利用するなどして、適切なレビューを実施できる体制を整えることが求められる。そのような観点に受験者の目が向けられているかどうかを判断することが、この問題の狙いである。

解答のポイント：

当該技術はソフトウェア開発の成否に関わる重要な技術であるため、まず、この技術に精通した技術者をレビューとして選定することが求められる。そのような人材が社内にはない場合は、社外のコンサルティングを受けるなどして適切なレビューが実施できる体制を整えておくことが望ましい。また、既存技術の延長上にある新技術であるため、

既存技術と新技術の違いを分析することで、新技術を適用する上でのポイントを洗い出すことも求められる。

不十分な解答の特徴の例：

- ・ レビューの目的を明確にする、レビューの観点を挙げる、早期にレビューを行う、適切なレビュー技法を用いるなど、問題が設定された状況とは無関係に、一般的なレビューの留意点しか挙げられていない。
- ・ 新技術経験者ではない技術者をレビューに参加させて、その技術者のスキル向上や組織内での横展開を図るなど、当該プロジェクトの成否とは関係のない副次的な効果を狙った内容を記述している。

以上