

平成 31 年度 春期
データベーススペシャリスト試験
午後 I 問題

試験時間

12:30 ～ 14:00 (1 時間 30 分)

注意事項

1. 試験開始及び終了は、監督員の時計が基準です。監督員の指示に従ってください。
2. 試験開始の合図があるまで、問題冊子を開いて中を見てはいけません。
3. 答案用紙への受験番号などの記入は、試験開始の合図があってから始めてください。
4. 問題は、次の表に従って解答してください。

問題番号	問 1 ～ 問 3
選択方法	2 問選択

5. 答案用紙の記入に当たっては、次の指示に従ってください。
 - (1) B 又は HB の黒鉛筆又はシャープペンシルを使用してください。
 - (2) 受験番号欄に受験番号を、生年月日欄に受験票の生年月日を記入してください。
正しく記入されていない場合は、採点されないことがあります。生年月日欄については、受験票の生年月日を訂正した場合でも、訂正前の生年月日を記入してください。
 - (3) 選択した問題については、次の例に従って、選択欄の問題番号を○印で囲んでください。○印がない場合は、採点されません。3 問とも○印で囲んだ場合は、はじめの 2 問について採点します。
〔問 1、問 3 を選択した場合の例〕
 - (4) 解答は、問題番号ごとに指定された枠内に記入してください。
 - (5) 解答は、丁寧な字ではっきりと書いてください。読みにくい場合は、減点の対象になります。

選択欄	
2 問 選 択	問 1
	問 2
	問 3

注意事項は問題冊子の裏表紙に続きます。
こちら側から裏返して、必ず読んでください。

問題文中で共通に使用される表記ルール

概念データモデル、関係スキーマ、関係データベースのテーブル（表）構造の表記ルールを次に示す。各問題文中に注記がない限り、この表記ルールが適用されているものとする。

1. 概念データモデルの表記ルール

(1) エンティティタイプとリレーションシップの表記ルールを、図1に示す。

- ① エンティティタイプは、長方形で表し、長方形の中にエンティティタイプ名を記入する。
- ② リレーションシップは、エンティティタイプ間に引かれた線で表す。
 - “1対1”のリレーションシップを表す線は、矢を付けない。
 - “1対多”のリレーションシップを表す線は、“多”側の端に矢を付ける。
 - “多対多”のリレーションシップを表す線は、両端に矢を付ける。

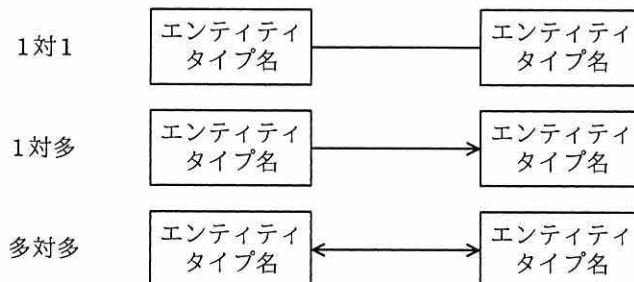


図1 エンティティタイプとリレーションシップの表記ルール

(2) リレーションシップを表す線で結ばれたエンティティタイプ間において、対応関係にゼロを含むか否かを区別して表現する場合の表記ルールを、図2に示す。

- ① 一方のエンティティタイプのインスタンスから見て、他方のエンティティタイプに対応するインスタンスが存在しないことがある場合は、リレーションシップを表す線の対応先側に“○”を付ける。
- ② 一方のエンティティタイプのインスタンスから見て、他方のエンティティタイプに対応するインスタンスが必ず存在する場合は、リレーションシップを表す線の対応先側に“●”を付ける。

“A” から見た “B” も，“B” から見た “A” も、インスタンスが存在しないことがある場合



“C” から見た “D” も，“D” から見た “C” も、インスタンスが必ず存在する場合



“E” から見た “F” は必ずインスタンスが存在するが，“F” から見た “E” はインスタンスが存在しないことがある場合

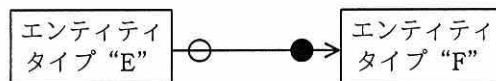
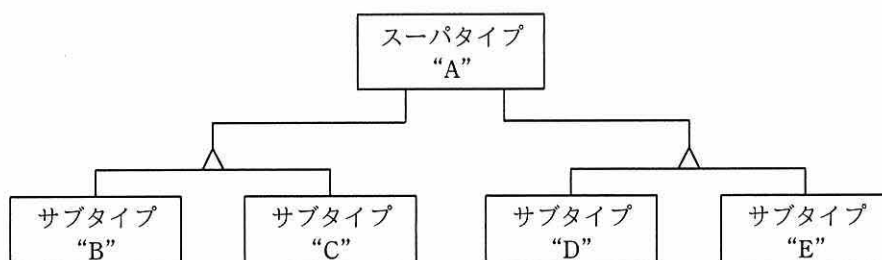


図 2 対応関係にゼロを含むか否かを区別して表現する場合の表記ルール

(3) スーパータイプとサブタイプ間のリレーションシップの表記ルールを、図 3 に示す。

- ① サブタイプの切り口の単位に “△” を記入し、スーパータイプから “△” に 1 本の線を引く。
- ② 一つのスーパータイプにサブタイプの切り口が複数ある場合は、切り口の単位ごとに “△” を記入し、スーパータイプからそれぞれの “△” に別の線を引く。
- ③ 切り口を表す “△” から、その切り口で分類されるサブタイプのそれぞれに線を引く。



スーパータイプ “A” に二つの切り口があり、それぞれの切り口にサブタイプ “B” と “C” 及び “D” と “E” がある例

図 3 スーパータイプとサブタイプ間のリレーションシップの表記ルール

(4) エンティティタイプの属性の表記ルールを、図 4 に示す。

- ① エンティティタイプの長方形内を上下 2 段に分割し、上段にエンティティタイプ名、下段に属性名の並びを記入する。¹⁾
- ② 主キーを表す場合は、主キーを構成する属性名又は属性名の組に実線の下線を付ける。
- ③ 外部キーを表す場合は、外部キーを構成する属性名又は属性名の組に破線の下線を付ける。ただし、主キーを構成する属性の組の一部が外部キーを構成する場合は、

破線の下線を付けない。

エンティティタイプ名
<u>属性名 1</u> , 属性名 2, … …, 属性名 n

図 4 エンティティタイプの属性の表記ルール

2. 関係スキーマの表記ルール及び関係データベースのテーブル（表）構造の表記ルール

(1) 関係スキーマの表記ルールを、図 5 に示す。

関係名 (属性名 1, 属性名 2, 属性名 3, …, 属性名 n)

図 5 関係スキーマの表記ルール

- ① 関係を、関係名とその右側の括弧でくくった属性名の並びで表す。¹⁾ これを関係スキーマと呼ぶ。
 - ② 主キーを表す場合は、主キーを構成する属性名又は属性名の組に実線の下線を付ける。
 - ③ 外部キーを表す場合は、外部キーを構成する属性名又は属性名の組に破線の下線を付ける。ただし、主キーを構成する属性の組の一部が外部キーを構成する場合は、破線の下線を付けない。
- (2) 関係データベースのテーブル（表）構造の表記ルールを、図 6 に示す。

テーブル名 (列名 1, 列名 2, 列名 3, …, 列名 n)

図 6 関係データベースのテーブル（表）構造の表記ルール

関係データベースのテーブル（表）構造の表記ルールは、(1) の ① ～ ③ で“関係名”を“テーブル名”に、“属性名”を“列名”に置き換えたものである。

注 ¹⁾ 属性名と属性名の間は“,”で区切る。

問1 データベース設計に関する次の記述を読んで、設問1～3に答えよ。

A社は、スポーツイベント（以下、大会という）の運営サービスを主催者に提供している大会運営サービス会社である。A社では、大会運営システムを新たに構築することになり、B君がデータベース設計を任された。

〔大会の登録から参加申込受付の準備まで〕

1. 主催者

- (1) 大会を主催する団体を主催者という。
- (2) 主催者は、主催者番号で識別し、主催者名、代表者氏名、住所、電話番号、メールアドレスを登録する。

2. 種目と種目分類

- (1) フルマラソン、ハーフマラソン、自転車ロードレースなどを種目という。
- (2) 種目は、種目コードで識別し、種目分類コードで、ランニング、自転車レースなどに分類する。

3. 大会

大会は、大会番号で識別し、大会名、開催年月日、開催場所の都道府県コード、主催者番号を登録する。

4. 運営サービス

- (1) A社が主催者に提供するサービスを運営サービスという。運営サービスには、大会に関する告知サービス、大会への参加申込みを受け付けるエントリーサービス、記録計測サービスなどがある。
- (2) 運営サービスは、運営サービスコードで識別し、運営サービス名、課金単位、単価を登録する。
- (3) 主催者は、大会ごとに一つ以上の運営サービスを選択する。A社は、主催者が選択した運営サービスを登録する。

5. エントリー枠

- (1) 大会において、参加希望者からの参加申込みを受け付ける単位をエントリー枠という。主催者は、大会ごとに一つ以上のエントリー枠を登録する。エントリー枠は、大会番号とエントリー枠番号で識別し、エントリー枠名、エントリー枠説明、種目コー

ド、定員、参加費用、募集期間（募集開始年月日～募集終了年月日）などを登録する。

- (2) 一つの大会において、幾つかのエントリ枠に同じ種目を登録することがある。
例えば、フルマラソンに対して、一般枠、地元優先枠、アスリート優先枠の三つのエントリ枠を登録することがある。
- (3) エントリ枠に対する参加者を決める方式には、先着順と抽選があり、先着順抽選区分で分類する。抽選の場合は、抽選年月日を登録する。抽選年月日には、募集終了年月日より後の日付を登録する。
- (4) エントリ枠には、エントリ枠状態を保持する。エントリ枠状態の取り得る値には、参加者を決める方式ごとに、先着順の場合は、‘募集前’、‘募集中’、‘参加者確定’があり、抽選の場合は、‘募集前’、‘募集中’、‘抽選中’、‘参加者確定’がある。

6. アイテム

- (1) 大会で、参加者に配布する参加賞や、ナンバカード、IC タグなどをアイテムという。
- (2) アイテムは、アイテムコードで識別し、アイテム名を登録する。
- (3) 主催者は、大会ごとに利用するアイテムを複数登録することができる。

[大会への参加申込みから参加費用の入金まで]

1. 会員

- (1) 大会の参加希望者は、あらかじめ会員登録をする。
- (2) 会員は、会員番号で識別し、会員氏名、性別、生年月日、住所、電話番号、メールアドレスを登録する。

2. 参加申込み及びエントリ枠状態の設定

- (1) 会員は、参加したい大会に対して、エントリ枠を指定して参加申込みを行う。
- (2) 会員は、一つの大会について一つのエントリ枠だけ参加申込みできる。
- (3) 参加申込みは、大会番号、会員番号で識別し、参加申込年月日を登録する。
- (4) 参加申込みと同時に、エントリ枠の参加申込数も合わせて更新する。
- (5) エントリ枠状態は、次のように設定する。
 - ① エントリ枠の登録においては、初期値を‘募集前’にする。

② 募集期間中は‘募集中’にする。

③ エントリ枠が先着順の場合

- ・募集期間が終わったら‘参加者確定’にする。

- ・参加申込数が定員に達したら、募集期間中であっても‘参加者確定’にする。

④ エントリ枠が抽選の場合

- ・募集期間が終わり、参加申込数が定員以下だったら‘参加者確定’にする。

- ・募集期間が終わり、参加申込数が定員を超えていれば‘抽選中’にし、その後、抽選年月日に抽選を実施した上で‘参加者確定’にする。

(6) エントリ枠状態が‘募集中’の間だけ、参加申込みを受け付ける。

3. 抽選結果の登録

抽選を実施したら、参加申込みに抽選結果を登録する。

4. 参加費用の入金及びポイントの付与

(1) 参加が確定したら、会員は参加費用を支払う。

(2) A 社は、会員の参加費用の支払を確認して入金年月日を登録し、参加費用に対して一定割合のポイントを会員に付与する。

(3) 会員は、保持しているポイントを、1 ポイント=1 円として、参加費用に充てることができる。

(4) 会員は、ポイントを使用する場合、使用ポイントを登録し、参加費用から使用ポイントを差し引いた額を支払う。

〔概念データモデルと関係スキーマの設計〕

B 君が設計した概念データモデルを図 1 に、関係スキーマを図 2 に示す。

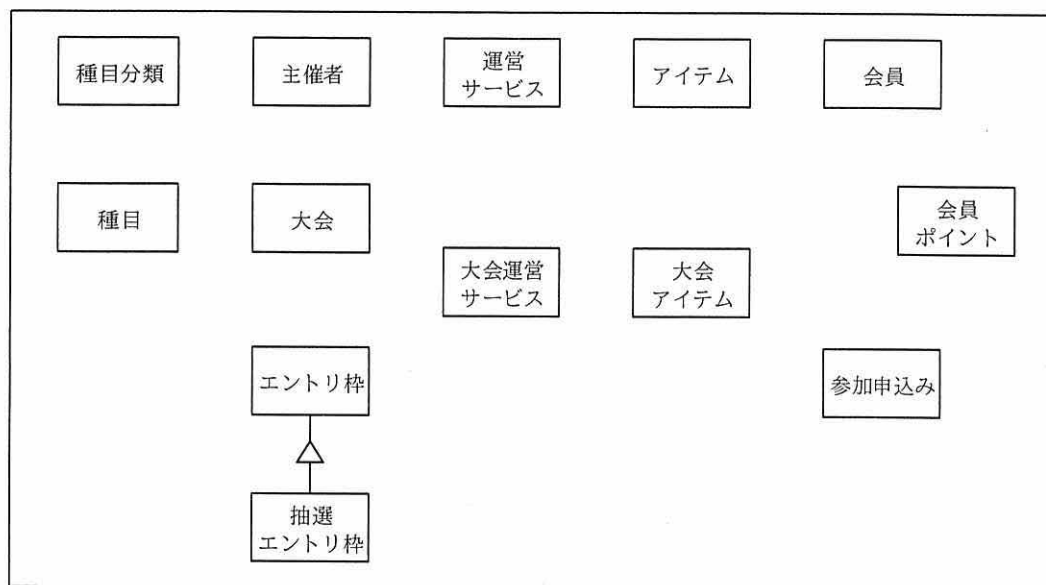


図 1 概念データモデル（未完成）

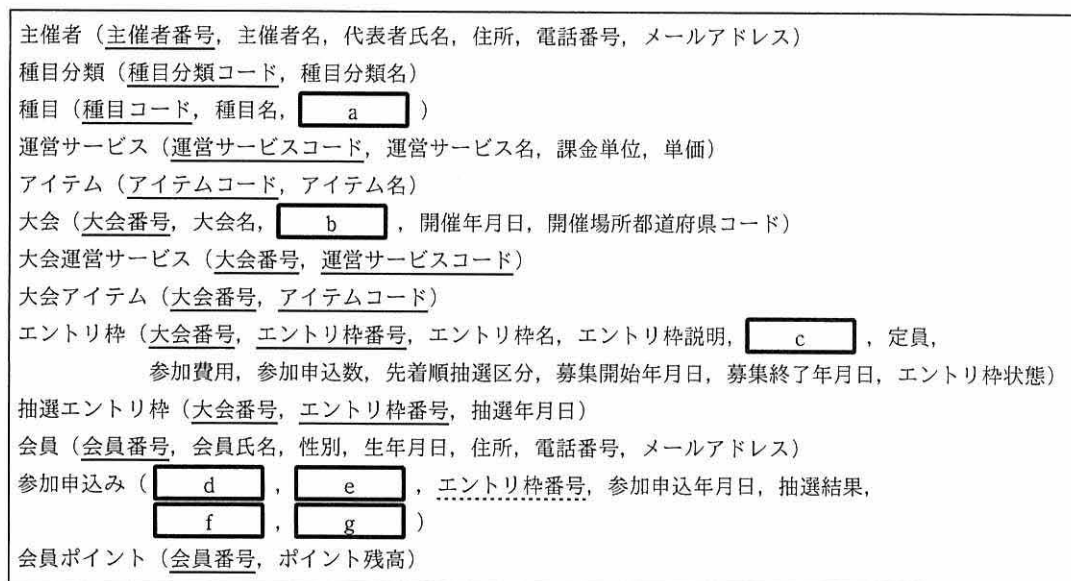


図 2 関係スキーマ（未完成）

〔指摘事項〕

C 部長は、概念データモデル及び関係スキーマに対して、次を指摘した。

- ・ エントリ枠状態と抽選実施を決める決定表が必要である。
- ・ この決定表は、日付が変わった時点及び参加申込受付時点で評価する。

この指摘を受けて作成した、日付が変わった時点及び参加申込受付時点で評価する決定表を表1に示す。この決定表の各条件の取り得る値は次のとおりである。

なお、() 内に略字がある場合、表1は略字で表す。

先着順抽選区分 : 先着順, 抽選

募集期間に対する本日 : 募集期間よりも前の日 (前), 募集期間中 (中),
募集期間よりも後の日 (後)

参加申込数 : 定員未満 (未満), 定員以下 (以下), 定員到達 (到達),
定員超過 (超過)

抽選年月日に対する本日 : 抽選年月日よりも前の日 (前), 当日,
抽選年月日よりも後の日 (後)

表1 日付が変わった時点及び参加申込受付時点で評価する決定表 (未完成)

先着順抽選区分	先着順	先着順	先着順	先着順	抽選	抽選	抽選	抽選	抽選	抽選
募集期間に対する本日	前	中	中	後						
参加申込数	—	未満	到達	—						
抽選年月日に対する本日	—	—	—	—						後
エントリ枠状態を ‘募集中’ にする	—	X	—	—	—	X	—	—	—	—
エントリ枠状態を ‘抽選中’ にする	—	—	—	—	—	—	—	X	—	—
抽選実施	—	—	—	—	—	—	—	—	X	—
エントリ枠状態を ‘参加者確定’ にする	—	—	X	X	—	—	X	—	X	X

〔新たな要件の追加〕

1. 多段階抽選方式

例えば、地元優先枠、アスリート優先枠、一般枠の三つの枠があり、会員が地元優先枠又はアスリート優先枠に参加申込みをして落選したら、その後に抽選を行う一般枠の抽選対象に加えるというような多段階に抽選するサービスを新たに追加することになった。

このサービスを実現するために、多段階抽選方式の仕様を次のように決定した。

- ・多段階抽選の対象のエントリ枠には、後続のエントリ枠を一つ設定する。
- ・後続のエントリ枠が設定されたエントリ枠で落選した会員は、後続するエントリ枠の抽選対象に加える。
- ・エントリ枠の抽選ごとに抽選結果を登録する。

2. ポイント有効期限

ポイントに有効期限を設けることにした。ポイントの有効期限は、付与された日から 1 年であり、有効期限を超過したポイントは消失する。ポイントの使用は、有効期限の近いものから行う。

解答に当たっては、巻頭の表記ルールに従うこと。ただし、エンティティタイプ間の対応関係にゼロを含むか否かの表記は必要ない。また、関係スキーマの表記又は関係スキーマに入れる属性名を答える場合、主キーを表す実線の下線及び外部キーを表す破線の下線を明記すること。

なお、エンティティタイプ間のリレーションシップには“多対多”のリレーションシップを用いないこと。

設問 1 図 1 の概念データモデル、図 2 の関係スキーマについて、(1)、(2)に答えよ。

- (1) 図 2 中の

a

 ～

g

 に入れる適切な属性名を答えよ。
- (2) 図 1 のリレーションシップは未完成である。必要なリレーションシップを全て記入し、図を完成させよ。

設問 2 表 1 は、太枠で示した部分が未完成である。太枠外の例に倣って表を完成させよ。

設問 3 〔新たな要件の追加〕について、(1)、(2)に答えよ。

- (1) 多段階抽選方式に対応できるよう、図 2 の関係スキーマに次の変更を行う。
 - ① ある関係に一つの属性を追加する。属性を追加する関係名及び追加する属性名を答えよ。
 - ② ある関係から一つの属性を削除する。属性を削除する関係名及び削除する属性名を答えよ。
 - ③ 新たに一つの関係を追加する。追加する関係の関係スキーマを答えよ。
- (2) ポイント有効期限に対応できるよう、関係“会員ポイント”を変更する。変更後の関係の属性名を全て答えよ。

問2 データベースでのトリガの実装に関する次の記述を読んで、設問1～3に答えよ。

オフィスじゅう器メーカーのY社は、部品の入出庫、発注を行う在庫管理システムを構築している。

〔RDBMSの主な仕様〕

在庫管理システムに用いているRDBMSの主な仕様は、次のとおりである。

1. ISOLATION レベル

選択できるトランザクションのISOLATIONレベルとその排他制御の内容は、表1のとおりである。ただし、データ参照時にFOR UPDATE句を指定すると、対象行に専有ロックを掛け、トランザクション終了時に解放する。

ロックは行単位で掛ける。共有ロックを掛けている間は、他のトランザクションからの対象行の参照は可能であり、更新は共有ロックの解放待ちとなる。専有ロックを掛けている間は、他のトランザクションからの対象行の参照、更新は専有ロックの解放待ちとなる。

表1 トランザクションのISOLATIONレベルとその排他制御の内容

ISOLATION レベル	排他制御の内容
READ COMMITTED	データ参照時に共有ロックを掛け、参照終了時に解放する。 データ更新時に専有ロックを掛け、トランザクション終了時に解放する。
REPEATABLE READ	データ参照時に共有ロックを掛け、トランザクション終了時に解放する。 データ更新時に専有ロックを掛け、トランザクション終了時に解放する。

索引を使わずに、表探索で全ての行に順次アクセスする場合、検索条件に合致するか否かにかかわらず全行をロック対象とする。索引探索の場合、索引から読み込んだ行だけをロック対象とする。

2. トリガ

テーブルに対する変更操作（挿入・更新・削除）を契機に、あらかじめ定義した処理を実行する。

(1) 実行タイミング（テーブルに対する変更操作の前又は後。前者を BEFORE トリガ、後者を AFTER トリガという）を定義することができる。

- (2) 列値による実行条件を定義することができる。
- (3) トリガ内では、トリガを実行する契機となった変更操作を行う前と後の行を参照することができる。挿入では操作後の行の内容を、更新では操作前と操作後の行の内容を、削除では操作前の行の内容を参照することができる。参照するには、操作前と操作後の行に対する相関名をそれぞれ定義し、相関名で列名を修飾する。
- (4) BEFORE トリガの処理開始から終了までの同一トランザクション内では、全てのテーブルに対して変更操作を行うことはできない。
- (5) トリガ内で例外を発生させることによって、契機となった変更操作をエラーとして終了することができる。

〔在庫管理システムのテーブル〕

在庫管理システムの主なテーブルのテーブル構造は、図 1 のとおりである。索引は、主キーだけに定義している。

生産ライン (<u>生産ラインコード</u> , ...)
部材メーカ (<u>部材メーカコード</u> , 会社名, ...)
部品 (<u>部品番号</u> , 部品名, 部品単価, <u>発注先部材メーカコード</u>)
在庫 (<u>部品番号</u> , 実在庫数量, 引当済数量, 基準在庫数量, 補充ロットサイズ, 発注済フラグ)
出庫要求 (<u>出庫要求番号</u> , 出庫年月日, 処理状況, <u>生産ラインコード</u>)
出庫要求明細 (<u>出庫要求番号</u> , <u>部品番号</u> , 出庫要求数量)
出庫 (<u>出庫番号</u> , <u>出庫要求番号</u> , <u>部品番号</u> , 出庫年月日, 出庫数量)
発注 (<u>発注番号</u> , 発注年月日, <u>部品番号</u> , 発注数量, 処理状況)
入庫 (<u>入庫番号</u> , 入庫年月日, <u>部品番号</u> , 入庫数量)

図 1 主なテーブルのテーブル構造

〔在庫管理業務の概要〕

- (1) 組立工場では、複数の生産ラインが稼働し、それぞれ異なる製品を組み立てている。各製品の組立てに必要な部品は倉庫に保管し、必要に応じて生産ラインに出庫する。
- (2) 部品は、部品番号で識別する。倉庫内に存在する在庫を、実在庫と呼ぶ。このうち、出庫対象となったものを、引当済在庫と呼ぶ。
- (3) 部品の発注の方式は定量発注である。Y 社の定量発注では、部品ごとの実在庫

数量から引当済数量を差し引いた値が、基準在庫数量を下回った都度、部品ごとに決められた部材メーカーに対して、決められた数量（補充ロットサイズ）を発注する。

- (4) 出庫要求では、倉庫に対して部品の出庫を要求する。“出庫要求”テーブル及び“出庫要求明細”テーブルに出庫要求の内容を登録し、“出庫要求”テーブルの処理状況に‘要求発生’を記録する。出庫要求番号は、出庫要求の発生順に一意な連番である。組立てに必要な複数の部品を一つの出庫要求とし、1 トランザクションで処理する。生産ラインごとに様々な組合せの部品を要求する。また、部品の要求は生産ラインでの組立ての状況に応じて任意の契機で発生する。

- (5) 在庫引当では、出庫要求に応じて、“出庫要求明細”テーブルに指定した部品番号の部品について出庫要求数量の出庫が可能かどうか判定し、出庫可能であれば“在庫”テーブルの引当済数量を更新する。全ての部品の在庫引当が完了したら、“出庫要求”テーブルの処理状況を‘引当実施’に更新する。

在庫引当できない部品が存在した場合は、在庫引当を破棄して処理状況を‘引当失敗’に更新し、部品が入庫されるのを待って改めて出庫要求する。

- (6) 出庫では、在庫引当が完了した部品を倉庫から搬出する。毎朝、“出庫要求”テーブルの処理状況が‘引当実施’のものを対象に実施する。それぞれの部品の出庫が完了したら、“在庫”テーブルの実在庫数量及び引当済数量を更新し、“出庫要求”テーブルの処理状況を‘出庫実施’に更新する。出庫は出庫要求単位に 1 トランザクションで処理し、全ての部品をまとめて出庫する。

- (7) 発注では、“発注”テーブルの処理状況に‘要求発生’を記録し、“在庫”テーブルの発注済フラグをオンにする。

- (8) 入庫では、部材メーカーから納品される都度、“在庫”テーブルの実在庫数量を更新し、発注済フラグをオフにする。また、“発注”テーブルの処理状況を‘入庫実施’に更新する。納品された複数の部品をまとめて、1 トランザクションで処理する。

〔トリガでの在庫引当処理の設計〕

出庫要求に連動した在庫引当を実行させたいので、トリガを利用するように処理を見直すことにした。トリガでの在庫引当処理を図 2 に示す。

なお、引当失敗の場合は、出庫要求側でロールバックを行った後、“出庫要求”テ

ーブルの処理状況を‘引当失敗’に更新する。

“**ア**”テーブルへの行の**イ**に対して、BEFORE トリガと AFTER トリガを定義する。

BEFORE トリガでは、“**ア**”テーブルに対して**イ**した行の、**ウ**列の値を検索条件に指定して、“**エ**”テーブルから**オ**，**カ**を参照する。出庫要求に指定した出庫要求数量と比較して、引当可能か判定し、不可能ならば契機となった変更操作を引当失敗のエラーにする。

AFTER トリガでは、“**エ**”テーブルの**カ**を更新する。

図 2 トリガでの在庫引当処理（未完成）

〔トリガでの定量発注の設計〕

在庫引当時の定量発注のために、発注の具体的な処理はストアードプロシージャで用意し、トリガから呼び出すことにした。ストアードプロシージャでは、“発注”テーブルと“在庫”テーブルに変更操作を行う。トリガを定義する SQL を図 3 に示す。さらに、図 3 の内容のレビューを行った。レビューでの指摘内容と対策を表 2 に示す。

```
CREATE TRIGGER TR1 a UPDATE OF 引当済数量 ON 在庫
REFERENCING b ROW AS CHKROW
c
d (CHKROW.実在庫数量 - CHKROW.引当済数量 <= CHKROW.基準在庫数量)
BEGIN ATOMIC
CALL PARTSORDER(CHKROW.部品番号);
END
```

注記 1 CHKROW は、トリガを実行する契機となったテーブルに対する変更操作の行を参照するために定義する相関名を表す。

注記 2 CALL PARTSORDER は、発注を行うストアードプロシージャの呼出しを表す。

図 3 トリガを定義する SQL（未完成）

表 2 指摘内容と対策（未完成）

指摘内容	対策
在庫引当以外に、 あ でも“ い ”テーブルの う 列が更新され、図 3 で定義したトリガが発動し、発注が繰り返されることになる。	発注が繰り返されないように、トリガ内で在庫を確認する際に、“ い ”テーブルの え 列の値も判定に加えることにした。

設問1 〔トリガでの在庫引当処理の設計〕について、(1)、(2)に答えよ。

- (1) 図2中の ～ に入れる適切な字句を答えよ。
- (2) 図2のトリガを実行するトランザクションの ISOLATION レベルについて、
(a)、(b)に答えよ。
- (a) READ COMMITTED で、複数の出庫要求が同時に発生した場合、トリガに起因して引当済数量が不正になることがある。どのようにして問題が発生するか、80字以内で述べよ。
- (b) READ COMMITTED で、(a)の問題を発生させないためには、BEFORE トリガで行を参照する際に、どのような対策を施す必要があるか、20字以内で述べよ。

設問2 〔トリガでの定量発注の設計〕について、(1)、(2)に答えよ。

- (1) 図3中の ～ に入れる適切な字句を解答群の中から選び、記号で答えよ。

解答群

ア AFTER

イ BEFORE

ウ FOR EACH ROW

エ FOR EACH STATEMENT

オ NEW

カ OLD

キ WHEN

ク WHERE

- (2) 表2中の ～ に入れる適切な字句を答えよ。

設問3 デッドロックについて、(1)、(2)に答えよ。

- (1) 出庫要求と入庫でデッドロックが発生することがある。対象のテーブル名を答えよ。
- (2) (1)のデッドロックの回避策を二つ挙げ、それぞれ50字以内で具体的に述べよ。

問3 部品表の設計及び処理に関する次の記述を読んで、設問1～4に答えよ。

E社は、機械メーカーである。E社では、RDBMSに構築した生産部品表（以下、部品表という）を用いて生産管理を行っている。情報システム部門のFさんは、新たに配属されたDB管理者のために、部品表に関する研修を担当することになった。

〔RDBMSの主な仕様〕

- (1) 索引は、ユニーク索引と非ユニーク索引に分けられる。
- (2) DMLのアクセスパスは、RDBMSによって索引探索又は表探索に決められる。
- (3) 索引探索に決められるためには、WHERE句のANDだけで結ばれた一つ以上の等値比較の述語の対象列が、索引キーの全体又は先頭から連続した一つ以上の列に一致していなければならない。ON句の場合も同様である。

〔部品表の概要〕

部品表は、E社が製造する製品と製品を構成する部品との関係を表すものである。Fさんは、研修で使用するためにE社の製品を簡略化して表した製品AX、AY及びAZの構成図を図1に示し、次のように説明することにした。

1. 品目

- (1) 品目は、品番で識別し、品目ごとに在庫をもっている。
- (2) 品目には、製品と、他の部品を使って組み立てられる中間部品、単独で使われる単体部品があり、品目区分で分類する。例えば、図1中のAXは製品、P1及びP2は中間部品、P3、P4及びP9は単体部品である。

2. 製品の構成

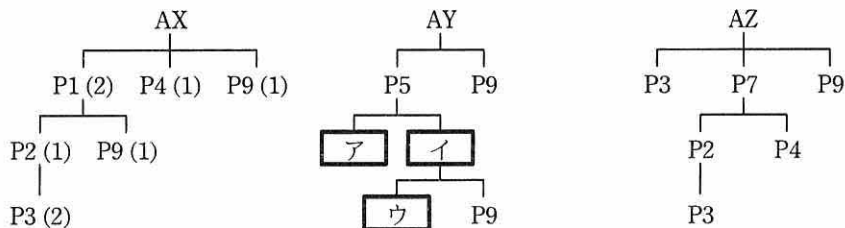
- (1) 製品は、複数種類の部品で構成され、構成図は、階層で表現される。製品からの階層の深さをレベルという。製品のレベルを0として、階層を一つ下ごとにレベルに1を加算する。例えば、製品AXを構成する部品について、レベル1に部品P1、P4及びP9、レベル2に部品P2及びP9、レベル3に部品P3がある。
- (2) 各部品について、当該部品を使う全ての製品の構成図の中で、当該部品が出現するレベルの最大値を、最も深い階層を示すことから、ローレベルコード（以下、LLCという）という。例えば、図1では、部品P5のLLCは1、部品

P9のLLCは3である。

(3) ある品目が他の品目から構成される場合、当該品目を親品目といい、親品番で識別する。また、当該親品目の一つ下のレベルの品目を子品目といい、子品番で識別する。

(4) 親品目1個あたりに使う各子品目の個数を構成数という。例えば、製品AXの製造に使う各部品の構成数は、次のとおりである。

- ① 製品AXは、1個あたり、部品P1を2個、部品P4、P9をそれぞれ1個ずつ使う。
- ② 部品P1は、1個あたり、部品P2、P9をそれぞれ1個ずつ使う。
- ③ 部品P2は、1個あたり、部品P3を2個使う。



注記 製品AXの各部品の（）内の数字は、当該部品の親品目1個あたりに使う当該部品の構成数を示す。製品AY及びAZの部品については構成数を省略している。

図1 製品AX、AY及びAZの構成図（未完成）

3. 主なテーブルのテーブル構造

E社が生産管理に用いている主なテーブルのテーブル構造を図2に示す。図1に基づいて登録した“品目”テーブルの行を表1に，“構成”テーブルの行を表2に示す。

なお、各テーブルには主索引だけが定義されている。索引キーが複合列の場合、テーブル構造に示した列の順番で定義される。

品目（ <u>品番</u> 、品名、品目区分、LLC、…）
構成（親品番、 <u>子品番</u> 、構成数、…）
在庫（ <u>品番</u> 、在庫数、引当可能数、…）

図2 主なテーブルのテーブル構造（一部省略）

表1 “品目” テーブルの行（一部省略）

品番	品名	品目区分	LLC	...
AX	AX	製品	0	...
AY	AY	製品	0	...
AZ	AZ	製品	0	...
P1	P1	中間部品	1	...
P2	P2	中間部品	2	...
P3	P3	単体部品	3	...
P4	P4	単体部品	2	...
P5	P5			...
P6	P6			...
P7	P7			...
P8	P8			...
P9	P9	単体部品	3	...

注記 網掛け部分は表示していない。

表2 “構成” テーブルの行（未完成）

親品番	子品番	構成数	...
AX	P1	2	...
AX	P4	1	...
AX	P9	1	...
AY	P5	1	...
AY	P9	1	...
AZ	P3	1	...
AZ	エ	2	...
AZ	P9	2	...
P1	P2	1	...
P1	P9	1	...
P2	P3	2	...
P5	P3	1	...
P5	P6	1	...
P6	P8	1	...
P6	P9	1	...
オ	カ	1	...
P7	P4	1	...

〔部品表に対する基本的な処理〕

1. 正展開処理、逆展開処理及び所要量計算処理の概要

Fさんは、部品表に対する三つの基本的な処理として、正展開処理、逆展開処理及び所要量計算処理を挙げた。

- (1) 正展開処理は、親品目がどの子品目を使っているかを、階層を上から下に1階層ずつたどることで調べる。
- (2) 逆展開処理は、子品目がどの親品目に使われているかを、正展開処理とは逆に、階層を下から上に1階層ずつたどることで調べる。逆展開処理は、ある部品が廃番になったとき、その部品がどの品目に影響するかを調べるときなどに行われる。
- (3) 所要量計算処理は、製品の生産計画に基づいて、各製品の製造に必要な部品の所要量を計算し、計算した所要量を部品の引当可能数から差し引くことで、在庫を引き当てる。

2. 正展開処理、逆展開処理及び所要量計算処理に用いるSQL

Fさんは、部品表に対する三つの基本的な処理に用いられる SQL の構文の例を、表 3 に示した。

表 3 SQL の構文の例（未完成）

SQL	上段：SQL の目的，下段：SQL の構文
SQL1	正展開処理において、図 1 中の製品 AZ を構成する部品のうち、レベル 2 までの部品ごとの品番と所要量を調べる。
	<pre> SELECT PNUM, SUM(QTY) SUMQTY FROM (SELECT 子品番 PNUM, 構成数 QTY FROM 構成 WHERE 親品番='AZ' UNION ALL SELECT L2.子品番 PNUM, L2.構成数*L1.構成数 QTY FROM 構成 L1 JOIN 構成 L2 ON L1. a = L2. b WHERE L1.親品番='AZ') TEMP GROUP BY PNUM ORDER BY PNUM </pre>
SQL2	逆展開処理において、レベル 2 に部品 P9 を使っている全ての製品の品番を調べる。
	<pre> SELECT L1.親品番 FROM 構成 L2 JOIN 構成 L1 ON L2. c = L1. d JOIN 品目 ON L1.親品番=品番 WHERE L2.子品番='P9' AND LLC=0 </pre>
SQL3	HPNUM に指定した品目の一つ下のレベルの全ての部品について、品番、構成数及び品目区分を調べる。ここで、当該品目の一つ下のレベルの値を、HLLC に設定する。
	<pre> SELECT 子品番, 構成数, 品目区分 FROM 構成 JOIN 品目 ON 子品番=品番 WHERE 親品番=:HPNUM AND LLC>=:HLLC ORDER BY 子品番 </pre>
SQL4	HPNUM に指定した部品について、所要量を HQT Y に設定し、在庫の引当可能数を更新する。
	<pre> UPDATE 在庫 SET 引当可能数=引当可能数-HQT Y WHERE 品番=:HPNUM </pre>

注記 HPNUM, HLLC 及び HQT Y は、ホスト変数である。

〔所要量計算処理プログラムの概要〕

所要量計算処理プログラムは、表 3 中の SQL3 及び SQL4 を用いる。Fさんは、製品を N 個製造する場合の所要量計算処理プログラムの処理手順を、表 4 に示した。

なお、当該処理は、次の前提で行うものとする。

- (1) 各部品の所要量の計算は、SQL3 を用いて調べた構成数に基づいて、プログラムのロジックで行う。
- (2) ISOLATION レベルは、READ COMMITTED とする。
- (3) 製品が複数ある場合、製品の品番順に、製品ごとに手順①～⑥を繰り返す。
- (4) 在庫は、適切に管理されているので、引当可能数が負の数になることはない。

表 4 所要量計算処理プログラムの処理手順

手順	処理
手順①	製品の品番を設定した SQL3 を用いて、当該製品の一下下のレベルの全ての部品について品番、構成数及び品目区分を調べる。
手順②	手順①で調べた全ての部品について、部品ごとに製品を N 個製造するために必要な当該レベルでの所要量を計算し、その所要量を設定した SQL4 を用いて、在庫の引当可能数を品番順に更新する。
手順③	前の手順（手順②又は⑤）で在庫を引き当てた全ての部品について、部品ごとに部品の品番を設定した SQL3 を用いて、当該部品の一下下のレベルの部品の品番、構成数及び品目区分を調べる。
手順④	手順③を実行した結果、一つでも部品が存在すれば手順⑤に進み、なければ手順⑥に進む。
手順⑤	手順③で調べた全ての部品について、部品ごとに製品を N 個製造するために必要な当該レベルでの所要量を計算し、その所要量を設定した SQL4 を用いて、製品ごとレベルごと部品ごとに、在庫の引当可能数を品番順に更新し、手順③に戻る。
手順⑥	COMMIT 文を発行し、他に処理すべき製品があれば手順①に戻り、なければ終了する。

〔Fさんの研修内容に対する K 部長の指示〕

表 4 について、K 部長から次のような指示があった。

指示 1 表 4 中の手順③の SQL3 の発行回数を減らすために、手順①及び③の SQL3 で部品の品目区分を調べている。その理由を説明すること。

指示 2 品目の設計変更において、例えば、製品 AX の部品 P3 を新部品 P11 に置き換えるべきところ、誤って部品 P1 を“構成”テーブルに登録してしまった場合、SQL3 の構文中に下線部分の述語が指定されていなければ、プログラムは不具合を起こすことを説明すること。

指示 3 製品は多品種なので、スループット向上のために所要量計算を製品ごとに分割して並行処理している。しかし、表 4 の処理手順のままではデッドロックが起きるので、プログラムを改良したことを説明すること。

設問 1 図 1 及び表 2 について、(1), (2)に答えよ。

(1) 図 1 中の ア ～ ウ に入れる適切な字句を答えよ。

(2) 表 2 中の エ ～ カ に入れる適切な字句を答えよ。

設問 2 〔部品表に対する基本的な処理〕の正展開処理について、(1), (2)に答えよ。

(1) 表 3 中の SQL1 の a , b に入れる適切な字句を答えよ。

- (2) 製品 AZ を 1 個製造するのに必要な、部品 P2, P3 及び P4 の所要量をそれぞれ答えよ。

設問 3 「部品表に対する基本的な処理」の逆展開処理について、(1), (2)に答えよ。

- (1) 表 3 中の SQL2 の c , d に入れる適切な字句を答えよ。
また、SQL2 を用いて得られる図 1 中の製品の品番を全て答えよ。
- (2) SQL2 が参照する全てのテーブルのアクセスパスは、索引探索に決められるようにしたい。“構成”テーブルにユニーク索引を追加する場合、その索引を構成する全ての列名を定義順に答えよ。

設問 4 「F さんの研修内容に対する K 部長の指示」について、(1)～(4)に答えよ。

- (1) 指示 1 に対して、なぜ部品の品目区分を調べれば、SQL3 の発行回数を減らすことができるのか、その理由を 30 字以内で述べよ。
- (2) 指示 2 に対して、プログラムが起こす不具合とは、処理がどのようなことか、20 字以内で述べよ。
- (3) 指示 3 で述べられたデッドロックについて、F さんは、図 1 の製品 AX と AZ の間で起きるデッドロックの一つのケースを、ケース 1 として図 3 に示し、デッドロックに関わる 2 種類の部品の組合せを丸印で囲んだ。図 3 に倣って、他にデッドロックが起きるケースをケース 2 として、図 4 を完成させよ。

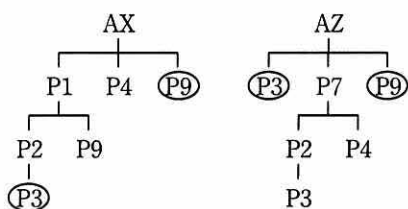


図 3 デッドロックのケース 1

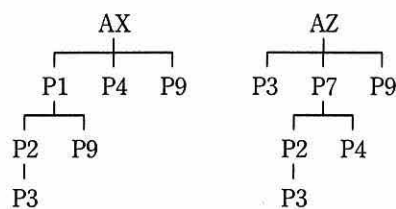


図 4 デッドロックのケース 2 (未完成)

- (4) 指示 3 に対して、F さんは、プログラムの改良について、次のように説明した。“SQL4 を、製品ごとレベルごと部品ごとに実行するのではなく、製品ごと部品ごとに集計した所要量をホスト変数 HQTY に設定してから表 4 の手順⑥の前に実行するように、手順②～⑤を改良しました。”

この説明に加えて、複数回の SQL4 をどのように実行するべきか、20 字以内で述べよ。

[メモ用紙]

6. 退室可能時間中に退室する場合は、手を挙げて監督員に合図し、答案用紙が回収されてから静かに退室してください。

退室可能時間	13:10 ～ 13:50
--------	---------------

7. 問題に関する質問にはお答えできません。文意どおり解釈してください。
8. 問題冊子の余白などは、適宜利用して構いません。ただし、問題冊子を切り離して利用することはできません。
9. 試験時間中、机上に置けるものは、次のものに限ります。
- なお、会場での貸出しは行っていません。
- 受験票、黒鉛筆及びシャープペンシル（B 又は HB）、鉛筆削り、消しゴム、定規、時計（時計型ウェアラブル端末は除く。アラームなど時計以外の機能は使用不可）、ハンカチ、ポケットティッシュ、目薬
- これら以外は机上に置けません。使用もできません。
10. 試験終了後、この問題冊子は持ち帰ることができます。
11. 答案用紙は、いかなる場合でも提出してください。回収時に提出しない場合は、採点されません。
12. 試験時間中にトイレへ行きたくなったり、気分が悪くなったりした場合は、手を挙げて監督員に合図してください。
13. 午後Ⅱの試験開始は 14:30 です。14:10 までに着席してください。

試験問題に記載されている会社名又は製品名は、それぞれ各社又は各組織の商標又は登録商標です。

なお、試験問題では、TM 及び ® を明記していません。