平成 24 年度 春期



基本情報技術者

•午前問	題	• •	• •	•	• •	•	• •	• •	•	•	2
(全80問	試題	浄 日	計	引:	2	寺	뺩	13	0.	分)

●午後 問題 · · · · · · 48

(全7問 試験時間:2時間30分)

問1~問7:7問中5問選択※

問8 : 必須解答

問9~問13:5問中1問選択



平成 24 年度 春 午前問題

問 1 から問 50 までは、テクノロジ系の問題です。



次の10進小数のうち、8進数に変換したときに有限小数になるものはどれか。

- 7 0.3
- 0.4
- **9** 0.5
- **I** 0.8



 $m{2}$ 非負の 2 進数 $\mathsf{b}_1\mathsf{b}_2\cdots\mathsf{b}_n$ を 3 倍にしたものはどれか。

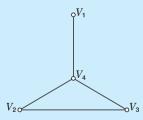
- $b_1b_2\cdots b_n0 + b_1b_2\cdots b_n$
- $b_1b_2\cdots b_n00 1$
- $b_1b_2\cdots b_n000$
- $b_1b_2\cdots b_n1$



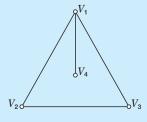
間 隣接行列 A で表されるグラフはどれか。ここで,隣接行列とは,n 個の節点から成るグラフの節点 V_i と V_j を結ぶ枝が存在するときは第 i 行第 j 列と第 j 行第 i 列の要素が 1 となり,存在しないときは 0 となる n 行 n 列の行列である。

〔隣接行列A〕

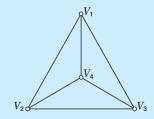




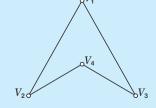




1



I



午前

8 進数の有限小数 ***ン!

8 進数の有限小数は、次のような 10 進数の分数の和に分解できます。

$$0.123\cdots = \frac{1}{8^1} + \frac{2}{8^2} + \frac{3}{8^3} + \cdots$$

選択肢の中から、上の形式に変換できる10進小数を選びます。変換できな いものは、8進数では無限小数になります。

$$\times$$
 $\boxed{7}$ $0.3 = \frac{3}{10}$

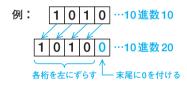
$$\times$$
 0.4 = $\frac{2}{5}$

$$0.5 = \frac{1}{2} = \frac{4}{8}$$
 \times I $0.8 = \frac{4}{5}$

$$\times$$
 \square $0.8 = \frac{4}{5}$

2進数の掛け算 キホン!

2進数の各桁を1つずつ左にずらし、末尾に0を付けると、新しい2進数は 元の数の2倍になります。

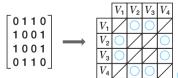


元の数の2倍に元の数を足せば、元の数の3倍になります。したがって、2 進数 $b_1b_2\cdots b_n$ を 3 倍にした数は,

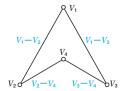
です。正解は \mathbf{r} です。 \mathbf{r} は4倍-1、 \mathbf{r} は8倍、 \mathbf{r} は2倍+1になります。

問 3 隣接行列

隣接行列 A を次のような表に直せば、その意味がはっきりします。



上の表から、枝で結ばれている節点の組合せは、 $V_1 - V_2$, $V_1 - V_3$, $V_2 - V_4$, $V_3 - V_4$ の4つである ことがわかります。このような枝をもつグラフは、 工だけです。



◆ 有限小数/無限小数 問1

10進小数を2進数や8進数 に変換したとき、小数点以下の 桁が有限の場合を有限小数, 小 数点以下の桁が無限に続く場合 を無限小数という。

例:10 進小数「0.4」→2進 小数「0.01100110011…|

200 覚えよう!

2進数の掛け算・割り算

元の数×2:2進数の各桁 を 1 つ左にずらし、末尾に 0を付ける(1ビット左シ

$$\begin{array}{c|c} & \times 2 \\ \hline & 1 & 1 & 1 \\ \hline \end{array}$$

元の数÷2:2進数の各桁 を 1 つ右にずらし、はみ出 た数は捨てる(1ビット右

$$\begin{array}{c|c}
1 & 0 & 0 & 0 \\
\hline
\end{array}$$

🔷 シフト演算

ビット列を左または右にずらす 演算。左にずらす場合を左シフ ト、右にずらす場合を右シフト という。

問 1 問 2 問3

周4

後置記法(逆ポーランド記法)では、例えば、式 $Y = (A - B) \times C$ を $YAB - C \times =$ と表現する。次の式を後置記法で表現したものはどれか。

$$Y = (A + B) \times (C - D \div E)$$

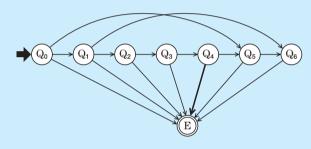
 γ YAB + C - DE \div × =

 \checkmark YAB + CDE ÷ - × =

 \rightarrow YAB + EDC \div - \times =

 \perp YBA + CD - E \div × =

問与 図は70円切符の自動販売機に硬貨が投入されたときの状態遷移を表している。 状態Q₄から状態Eへ遷移する事象はどれか。ここで、状態Q₀は硬貨が投入され ていない状態であり、硬貨が1枚投入されるたびに状態は矢印の方向へ遷移するものとする。 なお、状態 E は投入された硬貨の合計が70円以上になった状態であり、自動販売機は切 符を発行し、釣銭が必要な場合には釣銭を返す。また、自動販売機は10円硬貨、50円硬貨、 100円硬貨だけを受け付けるようになっている。



- ア 10 円硬貨が投入された。
- ✓ 10 円硬貨又は50 円硬貨が投入された。
- 10 円硬貨又は 100 円硬貨が投入された。
- 工 50 円硬貨又は100 円硬貨が投入された。

8 十分な大きさの配列 A と初期値が 0 の変数 p に対して、関数 f(x) と g() が次 のとおり定義されている。配列 A と変数 p は、関数 f(x) と g() だけでアクセ ス可能である。これらの関数が操作するデータ構造はどれか。

> function f(x) { p = p + 1; A[p] = x; return None; function g() { x = A[p]; p = p - 1; return x; }

アキュー

イ スタック

ウ ハッシュ

エ ヒープ

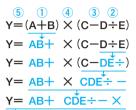
平成24年度春午前

問 4 後置記法(逆ポーランド記法) よく出る!

通常の数式(中置記法)から後置記法への変換は、次の手順で行います。

- ①式に含まれる演算子に, 計算する順番に番号を付ける。
- ②演算子とその前後の項を,番号順に後置記法に変換する。ただし,変換済みの部分はひとまとまりの項として扱い,カッコは取り除く。

手順にしたがって式を変換すると、次のようになります。



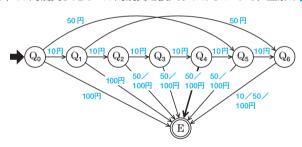
 $\overrightarrow{YAB}+\overrightarrow{CDE}+\times=$

← 正解は イ

問 5 状態遷移

状態 $Q_0 \sim Q_6$ は、投入金額がまだ 70 円に満たない状態を表します。それぞれの投入金額は順に 0 円、10 円、20 円…60 円になると考えられます。状態 Q_4 は、投入金額が 40 円になった状態です。

投入金額が40 円の状態 (Q_4) から、投入金額が70 円以上の状態 (E) に遷移するには、50 円硬貨または100 円硬貨を投入すればよいので、正解は \blacksquare です。



問 6 データ構造

関数 f(x) は,配列 A の末尾にデータを格納し,関数 g() は,配列 A の末尾からデータを取り出します。つまり,データは最後に格納したデータから順に取り出される構造になっています。このようなデータ構造をスタックといい,関数 f(x) の操作をブッシュ,関数 g() の操作をボップといいます。

×アキューは、最初に格納したデータを最初に取り出すデータ構造です。

○ 1 正解です。

× フ ハッシュは、格納するデータを基に計算されるハッシュ値によって格納位置を決めるデータ構造です。

× **エ ヒープ**は、親の値が子の値より常に小さく(または大きく)なるように配置された2分木のデータ構造です。

3、午前のカギ

☆ 逆ポーランド記法(後置記法)

演算子を被演算子の後に置く表記法。コンピュータで処理しやすいので、プログラムでよく利用される。ポーランド記法(前置記法)の逆なので、逆ポーランド記法という。

登覚えよう! □ 4

中置記法→逆ポーランド記 法変換の手順

- ①計算する順に演算子に番 号を付ける
- ②番号順に後置記法に変換

問 5

対策 実際の自動販売機の動作を思い浮かべると、わかりやすいよ。



➡ 状態遷移図

システムがある状態から別の状態へと移り変わる様子を表した図。

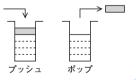
※ 覚えよう! 闘

スタックといえば

最後に格納したデータを最初に取り出す(LIFO:Last-In First-Out)データ構造

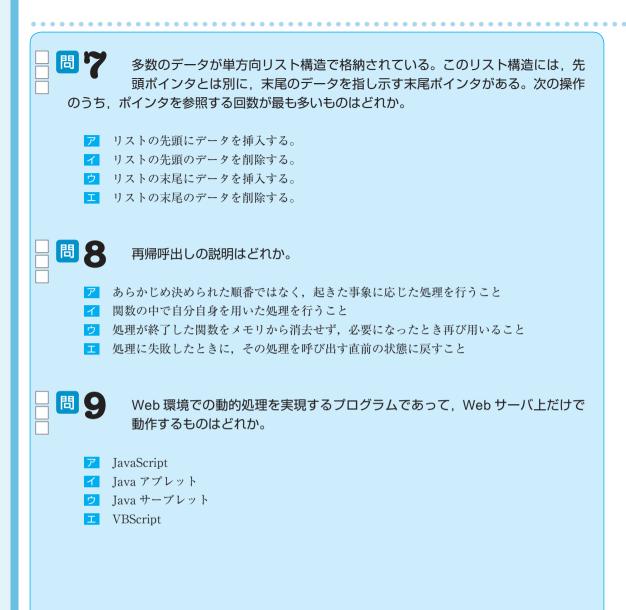
プッシュ:データをスタック に格納

ポップ: データをスタックから取り出す



解答

問4 **1** 問5 **工** 問6 **1**



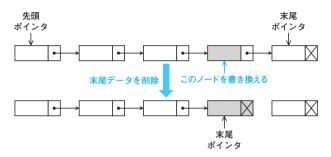
解説

問7 単方向リスト構造

リストの末尾のデータを削除するには、末尾の1つ前のノードのポインタを書き換える必要があります。**単方向リスト構造**では、ノードを末尾から先頭方向に向かってたどることができないため、末尾の1つ前のノードにアクセスするには、リストを先頭から順にたどるしかありません。したがってポインタの参照回数は、ノード数-1回になります。正解はエです。



- × **プ 先頭にデータを挿入**:挿入ノードのポインタに現在の先頭ポインタ を代入し、先頭ポインタに挿入ノードのアドレスを代入します。
- × **ウ 末尾にデータを挿入**:現在の末尾ポインタが指すノード(現在の末尾 ノード)のポインタに挿入ノードのアドレスを代入し、同じアドレスを未尾ポインタにも代入します。
- **末尾データを削除**:先頭から順にノードをたどり、末尾ノードの1つ 前のノードのポインタを NULL 値にします。次に、このノードのアドレスを末尾ポインタに代入します。



問 8 再帰呼出し キホン!

再帰呼出し(リカーシブ・コール)とは、ある関数の処理中に、その関数自身を呼び出す処理のことです。

- × **ア** 起きた事象 (イベント) に応じた処理を行う方式を, **イベントドリブ ン**といいます。
- イ 正解です。
- × ウ いったん実行したプログラムを、メモリに読み込み直さずに繰り返し 実行できることを、再使用可能(リユーザブル)といいます。
- × エ エラーが発生したとき、エラーが発生する前の状態に戻すことをロー ルバックといいます。

問9 Web プログラム よく出る!

選択肢はいずれも Web 環境での動的処理を行いますが、Web サーバ上での み動作するのは Java サーブレットだけです。

- × フ JavaScript は、HTML や XML ファイル内に記述し、Web ブラウザによって実行されるスクリプト言語です。
- × **Java アブレット**は、Web サーバからダウンロードされ、Web ブラウザ上で実行される Java プログラムです。
- D 正解です。Java サーブレットは、Web サーバ上で実行される Java プログラムです。
- × **I** VBScript はマイクロソフトが開発したスクリプト言語で、Web サーバや Windows PC などで動作します。



●単方向リスト構造 複数のノードを数珠つなぎに連結したデータ構造。1つのノードはデータとポインタから構成され、ポインタには次のノードのアドレスが格納されている。ノードの挿入や削除はポインタを付け替えるだけでできるが、目的のデータにアクセスするには、ノードを先頭から順にたどっていく必要がある。

 $\vec{r} - \vec{y} \xrightarrow{\bullet} \vec{r} - \vec{y} \xrightarrow{\bullet} \vec{r} - \vec{y} \xrightarrow{\bullet} \cdots$

登覚えよう!

Java サーブレット

といえば

Webサーバ上で動作する Javaプログラム

JavaScript といえば

Webブラウザ上で動作するスクリプト言語

Java アプレットといえば

Webサーバからダウンロードされ、Webブラウザ上で動作する Javaプログラム

○ 解答

問7 <u>工</u> 問8 <u>イ</u> 問9 <u>ウ</u>

\square コアの個数を n 倍にすると、プロセッサ全体の処理性能は n^2 倍になる。 ✓ 消費電力を抑えながら、プロセッサ全体の処理性能を高められる。 ウ 複数のコアが同時に動作しても、共有資源の競合は発生しない。 プロセッサのクロック周波数をシングルコアより高められる。 問 外部割込みに分類されるものはどれか。 ア インターバルタイマによって、指定時間経過時に生じる割込み ウ 仮想記憶管理において、存在しないページへのアクセスによって生じる割込み エソフトウェア割込み命令の実行によって生じる割込み ア 一定時間ごとに内容を外部記憶装置に書き込む。 ✓ システムの電源投入時に、全領域を0で初期化する。 ヴ データを保持するために、一定時間ごとにアクセスする。 ■ 内容を更新するときに、データを一旦消去する。 問 📘 🔀 キャッシュメモリに関する記述のうち、適切なものはどれか。 ア キャッシュメモリの転送ブロックの大きさを仮想記憶のページの大きさと同じにする と、プログラムの実行効率が向上する。 ✓ キャッシュメモリは高速アクセスが可能なので、汎用レジスタと同じ働きをする。 す 主記憶のアクセス時間とプロセッサの命令実行時間の差が大きいマシンでは、多段の キャッシュ構成にすることで実効アクセス時間が短縮できる。

ミスヒットが発生するとキャッシュ全体は一括消去され、主記憶から最新のデータの転

解説

問10 マルチコアプロセッサ

送処理が実行される。

マルチコアプロセッサは、1つのパッケージ上に、CPUのコア(中核部分) を複数搭載したマイクロプロセッサです。



- × ア マルチコアプロセッサは、複数の処理を各コアが並列に実行することで処理を高速化します。したがって処理性能は n 倍より大きくはなりません。
- I 正解です。マルチコアになっても CPU は 1 台だけなので,複数の CPU を搭載するより消費電力を抑えることができます。
- × **ウ** 複数の処理を同時に実行すれば、共有資源の競合が発生する可能性があります。
- × **エ** マルチコアプロセッサのクロック周波数は、同性能のシングルコアより低くなります。

十則の刀干

11 外部割込み よく出る!

実行中のプログラムを中断し、CPUに強制的に別の処理を実行させることを割込みといいます。ソフトウェアによって発生する割込みを内部割込み、周辺機器などのハードウェアによって発生する割込みを外部割込みといいます。

- ア 正解です。タイマによって発生する割込みは外部割込みの一種で、タ イマ割込みといいます。
- × **オ**ーバフローやゼロによる除算はソフトウェアのエラーなので、内部 割込みです。
- × り 仮想記憶へのアクセスはシステムソフトウェアが管理するので、内部 割込みです。
- × I ソフトウェア割込み命令はソフトウェアの実行によって生じるので、 内部割込みです。

問12 DRAMのリフレッシュ動作 キホン!

DRAMは、電源を切ると内容が消えてしまう揮発性のメモリの一種で、構造が単純で大容量化が簡単にできるため、主に主記憶装置に用いられています。 DRAMは一定時間ごとに再書込みをしないと、内容が消えてしまう特徴があります。この再書込み動作をリフレッシュ動作といいます。正解はってす。

問13 キャッシュメモリ

キャッシュメモリは、CPUと主記憶との間に設置される高速なメモリです。 CPUが処理するデータを一時的に蓄えておき、読み書きの遅い主記憶へのアクセス頻度を減らして処理を高速化します。CPUと主記憶の速度差が大きい場合には、1次キャッシュ、2次キャッシュのように多段構成になっています。

- ×ア キャッシュメモリと仮想記憶のページは関係ありません。
- ウ 正解です。
- × I ミスヒット(キャッシュメモリに目的のデータがなかった場合)が発生すると、データは主記憶から読み出されます。キャッシュメモリの内容は一部更新されるだけです。

≥ 覚えよう!

内部割込みといえば

プログラムによって発生

外部割込みといえば

ハードウェアによって発生

入出力割込み	入出力処理の完了や エラーで発生
タイマ割込み	タイマによって発生
機械チェック 割込み	バードウェアの障害 などによって発生
コンソール 割込み	割込みスイッチの操 作で発生

29 覚えよう!

DRAM といえば

- 構造が単純で比較的安価に 高集積化が可能
- リフレッシュ動作が必要
- 主記憶装置に用いる

SRAM といえば

- フリップフロップ回路を用い、リフレッシュ動作が不要
- 高速だが構造が複雑で高価
- キャッシュメモリに用いる

★ キャッシュメモリ主記憶へのアクセス速度とCPUの処理速度の差を埋めるためのメモリ。

○ 解答

問10 4 問11 7 問12 **ウ** 問13 **ウ**

□ 14 プラズマディスプレイの説明として、適切なものはどれか。

- ア ガス放電によって発生する光を利用して、映像を表示する。
- ✓ 自身では発光しないので、バックライトを使って映像を表示する。
- **工** 電子銃から発射した電子ビームを管面の蛍光体に当てて発光させ、文字や映像を表示する。

$igspace m{ ext{l}}$ $m{ ext{l}}$ クラウドコンピューティングの説明として、最も適切なものはどれか。

- ア あらゆる電化製品をインテリジェント化しネットワークに接続することによって、いつでもどこからでもそれらの機器の監視や操作ができるようになること
- ウ コンピュータの資源をネットワークを介して提供することによって、利用者がスケーラ ビリティやアベイラビリティの高いサービスを容易に受けられるようになること
- ▼ 特定のサーバを介することなく、ネットワーク上の PC 同士が対等の関係で相互に通信を行うこと

- ア システム構成に冗長性をもたせ、部品が故障してもその影響を最小限に抑えることで、 システム全体には影響を与えずに処理を続けられるようにする。
- √ システムに障害が発生したときの原因究明や復旧のため、システム稼働中のデータベースの変更情報などの履歴を自動的に記録する。
- 工 操作ミスが発生しにくい容易な操作にするか、操作ミスが発生しても致命的な誤りとな らないように設計する。

平成24年度春午

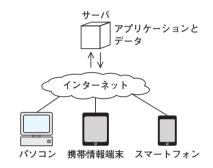
問14 プラズマディスプレイ よく出る!

プラズマディスプレイは、コントラストが高く、広い視野が得られることが 特徴の薄型ディスプレイ装置で、ガス放電による発光を利用しています。

- ア正解です。
- ×イ 液晶ディスプレイの説明です。
- × ウ 有機 EL ディスプレイの説明です。
- × I CRT ディスプレイの説明です。

問15 クラウドコンピューティング ※初モノ!*

クラウドコンピューティングとは、サーバからサービスとして提供される各種の機能を、インターネットを介して各利用者が利用する形態です。データをネットワーク上に配置して共有したり、同じ機能を PC、携帯情報端末、スマートフォンといった異なる機器で利用できるといったメリットがあります。



- × ア ホームネットワークの説明です。
- × イ クラスタリングの説明です。
- ウ 正解です。
- × エ ピアツーピアネットワークの説明です。

問16 フォールトトレラントシステム ******

システムの一部に障害が発生しても、システム全体を停止させずに処理を継続できるように構成されたシステムをフォールトトレラントシステムといいます。フォールトトレラントシステムを実現するには、同じ構成部品を複数用意しておき、1台が故障してももう1台の部品で正常に処理を続けるなどの方法をとります。この方式を、システムの冗長化といいます。

- ア正解です。
- × 履歴の記録はシステムが停止したときに備えるもので、フォールトトレラントシステムの要素ではありません。
- ×ゥバックアップはシステムが停止したときに備えて行う作業です。
- ×エフールプルーフ設計の説明です。



登覚えよう!

プラズマディスプレイ

といえば

• ガス放電により発光

有機 EL ディスプレイ

といえば

有機化合物で自ら発光

☆ フールプルーフ 利用者が誤場作をした

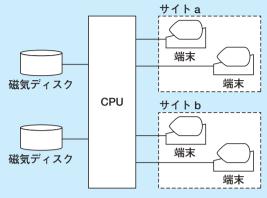
利用者が誤操作をしにくくしたり、誤操作が致命的な影響を及ぼさないようにシステムを設計すること。

機器を二重化(冗長化)するなどして、システムの一部が故障 しても支障がないようにしたシステム。

解答

問14 **ア** 問15 **ウ** 問16 **ア**

□ 1 図に示すシステム構成全体の稼働率を表す式はどれか。ここで、システムが正常に稼働するためには、磁気ディスクは2台とも正常でなければならず、それぞれのサイトで少なくとも1台の端末は正常でなければならない。



装置	1台の稼働率
磁気ディスク	D
CPU	C
端末	T

 $D^2C (1-T^2)^2$

 $D^2C (1-(1-T)^2)^2$

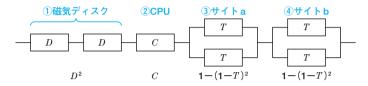
 $(1-D)^2C(1-T^2)^2$

- $(1-D)^2C(1-(1-T)^2)^2$
- ullet 問 $oxed{18}$ ベンチマークテストの説明として、適切なものはどれか。
 - ア 監視・計測用のプログラムによってシステムの稼働状態や資源の状況を測定し、システム構成や応答性能のデータを得る。
 - ✓ 使用目的に合わせて選定した標準的なプログラムを実行させ、その処理性能を測定する。
 - ウ 将来の予測を含めて評価する場合などに、モデルを作成して模擬的に実験するプログラムでシステムの性能を評価する。
- 問 **19** オンラインリアルタイム処理における一つのトランザクションについて、端末側で応答時間、回線伝送時間、端末処理時間が測定できるとき、サーバ処理時間を求める式として適切なものはどれか。ここで、他のオーバヘッドは無視するものとする。
 - ア サーバ処理時間 = 応答時間 + 回線伝送時間 + 端末処理時間
 - ✓ サーバ処理時間 = 応答時間 + 回線伝送時間 端末処理時間
 - ウ サーバ処理時間=応答時間-回線伝送時間+端末処理時間
 - エ サーバ処理時間 = 応答時間 − 回線伝送時間 − 端末処理時間

問17 稼働率の式

システム構成を, ①磁気ディスク, ② CPU, ③サイト a, ④サイト b に分 けて、それぞれの部分ごとの稼働率を考えます。

- ①磁気ディスク:2台とも稼働していなければならないので直列システムとみ なし、稼働率は $D \times D = D^2$ となります。
- ②**CPU**: 稼働率は*Cと*なります。
- ③サイトa, ④サイトb: 少なくとも1台の端末が稼働していればいいので並 列システムとみなし、稼働率はそれぞれ $1-(1-T)^2$ となります。



①~④がすべて稼働していなければならないので、システム全体の稼働率は、

 $D^2 \times C \times (1 - (1 - T)^2) \times (1 - (1 - T)^2) = D^2 \times C \times (1 - (1 - T)^2)^2$ となります。正解は 1です。

ベンチマークテスト ***ン!

ベンチマークテストは、基準となる標準的なプログラムを実行してその実行 時間を測定し、コンピュータの性能を比較可能な数値で評価するテストです。

- × ア モニタリングの説明です。
- イ 正解です。
- × ウ シミュレーションの説明です。
- ×II 机上テストの説明です。

問19 サーバ処理時間の計算

利用者が要求を入力してから、端末にその応答が返ってくるまでの時間を応 答時間(レスポンスタイム)といいます。

応答時間は、端末処理時間と回線伝送時間、サーバ処理時間の合計なので、 次のように求められます。

応答時間=端末処理時間十回線伝送時間十サーバ処理時間

サーバ処理時間=応答時間-回線伝送時間-端末処理時間

以上から、正解はエです。



200 覚えよう!

直列システムといえば

- 1 台が故障すると全体が停 止する
- 稼働率= a×b



並列システムといえば

- どれか 1 台でも稼働してい れば全体が稼働する
- 稼働率=1-(1-a)×(1-b)



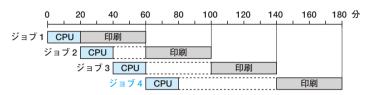
解答 問17 ☐ 問18 問19

問 20 次の条件で四つのジョブが CPU 処理及び印刷を行う場合に、最初の CPU 処理 を開始してから最後の印刷が終了するまでの時間は何分か。
 〔条件〕 (1) 多重度 1 で実行される。 (2) 各ジョブの CPU 処理時間は 20 分である。 (3) 各ジョブは CPU 処理終了時に 400M バイトの印刷データをスプーリングする。スプーリング終了後に OS の印刷機能が働き、プリンタで印刷される。 (4) プリンタは 1 台であり、印刷速度は 100M バイト当たり 10 分である。 (5) CPU 処理と印刷機能は同時に動作可能で、互いに影響を及ぼさない。 (6) スプーリングに要する時間など、条件に記述されていない時間は無視できる。
7 120
問 21 仮想記憶システムにおいて主記憶の容量が十分でない場合,プログラムの多重度を増加させるとシステムのオーバヘッドが増加し、アプリケーションのプロセッサ使用率が減少する状態を表すものはどれか。
ア スラッシング イ フラグメンテーション ウ ページング エ ボトルネック
問 22 ページ置換えアルゴリズムにおけるLRU方式の説明として、適切なものはどれか。
ア 最後に参照されたページを置き換える方式 イ 最後に参照されてからの経過時間が最も長いページを置き換える方式 ウ 最も参照回数の少ないページを置き換える方式 エ 最も古くから存在するページを置き換える方式
問 23 様々なサイズのメモリ資源を使用するリアルタイムシステムのメモリプール管理において、可変長方式と比べた場合の固定長方式の特徴として、適切なものはどれか。
ア メモリ効率が良く、獲得及び返却の処理速度は遅く一定である。 イ メモリ効率が良く、獲得及び返却の処理速度は遅く不定である。 ウ メモリ効率が悪く、獲得及び返却の処理速度は速く一定である。 エ メモリ効率が悪く、獲得及び返却の処理速度は速く不定である。

平成24年度春午前

問20 ジョブスケジューリング よく出る!

多重度が 1 なので、CPU は同時に 1 つのジョブしか実行できません。ただし、CPU 処理と印刷処理は同時に動作可能なので、あるジョブが印刷処理をしている間、別のジョブが CPU 処理を行うことはできます。印刷処理は、1 つのジョブにつき $400\div100\times10=40$ 分かかります。



上図のように、4つ目のジョブが完了するのは開始から 180 分後です。正解は $\stackrel{\bullet}{\text{D}}$ です。

問21 仮想記憶システム

仮想記憶システムで主記憶の容量が十分でない場合,実記憶と磁気ディスク間でデータの入れ替えが頻繁に起こるため,システムのオーバヘッドが増加します。この現象をスラッシングといいます。

○ ア 正解です。

- × **フラグメンテーション**とは、記憶領域の割当てと解放を繰り返すうちに、空き領域が細分化されてしまう現象です。
- × つ ページングとは、仮想記憶システムにおいて、記憶領域を一定サイズ のページに分割し、ページ単位で内容を管理することです。
- × **エ ボトルネック**とは、システム全体の処理性能を制約する原因となって いる要素のことです。

問22 ページ置換えアルゴリズム よく出る!

ページ置換えアルゴリズムは、仮想記憶システムで主記憶から磁気ディスクに移動するページを選択する方式で、代表的なものに LRU 方式や FIFO 方式 があります。

LRU (Least Recently Used) 方式は、最後に参照されてから最も長く時間が経過しているページを置き換える方式なので、イが正解です。

問23 可変長方式と固定長方式

可変長方式は、個々の獲得要求に応じて、必要なサイズの記憶領域を確保する方式です。この方式では、メモリ効率はよくなりますが、空き領域の探索などの処理が余計に必要な分、獲得及び返却の処理速度は遅く、不定になります。

固定長方式は、常に一定サイズの記憶領域を確保する方式です。この方式では、メモリ効率は悪くなりますが、獲得及び返却の処理速度は速く、一定です。以上より、正解は [→]です。



問 20

対策 ジョブスケジューリン グの問題は、図にするとわか りやすいよ。



● 仮想記憶

問 21

主記憶上にある実記憶領域の一部を必要に応じてハードディスクなどに退避させることで、物理的な容量より大きな主記憶領域を利用できるようにする仕組み。

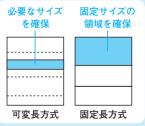
登覚えよう! [

ページ置換えアルゴリズム といえば

- LRU:最後に参照されてから最も長く時間が経過しているページを置き換える
- FIFO: いちばん古くからあるページを置き換える

🚭 可変長方式と固定長方式

問 23

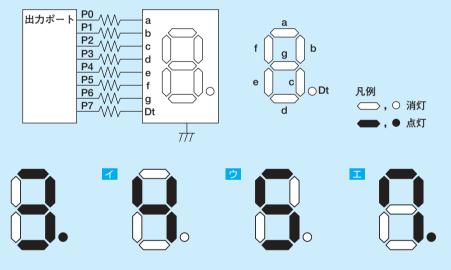


○ 解答

問20 **立** 問21 **ア** 問22 **イ** 問23 **立**

- □ **24** シェルのリダイレクト機能による実現の可否に関する記述のうち、適切なものはどれか。

 ▽ 標準出力をファイルに切り替えることはできないが、標準入力をファイルに切り替えることはできる。
 - ✓ 標準出力をファイルに追加することはできないが、標準入力と標準出力をファイルに切り替えることはできる。
 - 戸 標準入力と標準出力をファイルに切り替えることができ、標準出力をファイルに追加することもできる。
 - 標準入力をファイルに切り替えることはできないが、標準出力をファイルに切り替えることはできる。



- □ **26** 60 分の音声信号(モノラル)を、標本化周波数 44.1kHz、量子化ビット数 16 ビットの PCM 方式でディジタル化した場合、データ量はおよそ何 M バイトか。 ここで、データの圧縮は行わないものとする。
 - **7** 80 **1** 160 **□** 320 **□** 640

問24 リダイレクト機能

Linux などのUNIX系 OSでは、シェルというプログラムが利用者の要求(コ マンド)を受け付け、プログラムの実行を制御しています。多くのプログラム は、処理データを標準入力から読み込み、結果を標準出力に出力する仕様になっ ています。通常、標準入力はキーボード、標準出力はディスプレイですが、こ れをファイルやプリンタなどに切り替えるのが、リダイレクト機能です。

% command > outfile #標準出力をファイル outfile に切り替え

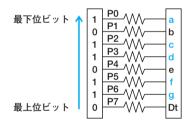
% command < infile #標準入力をファイル infile に切り替え

% command >> addfile #標準出力を既存のファイル addfileに追加出力

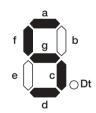
上記のように、標準入力と標準出力はファイルに切り替えることができ、標 準出力は既存のファイルに追加出力することもできます。以上から、正解は **ウ** です。

問25 LED 点灯回路

16 進数 6D を 2 進数に変換すると、「0110 1101」になります。この 2 進数 の各ビットを、次のように $P0 \sim P7$ に割り当てます。



上図のうち、ポート出力が1のLEDは、a, c, d, f, qの5つです。この5つのLEDを点灯させると、右図の ようになります。正解は ウ です。



B26 PCM 方式 よく出る!

標本化周波数が 44.1 kHz なので、1 秒間に 44.1×10^3 個のサンプリングデー タが録音されます。音声信号(モノラル)は60分なので、サンプリング総数は $44.1 \times 10^3 \times 3.600$ 個になります (ステレオ音声の場合は、さらに倍にします)。 量子化ビット数が16ビット=2バイトの場合,サンプリングデータ1個に つき2バイトなので、データ量は次のようになります。

 $44.1 \times 10^3 \times 3600 \times 2 = 317520 \times 10^3 \text{ if } 1 = 317.52 \text{M if$

以上から、正解はウです。

問 24

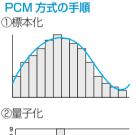
参考 Windows PC でも「コ __ マンドプロンプト」でリダイ レクト機能を使えるよ。

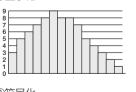


🔷 標準出力

UNIX系OSで、プログラムの 既定の出力先となる装置のこ と。プログラムは出力結果を標 準出力に出力し、OS 側が標準 出力を実際の装置に結びつける 仕組み。標準出力は通常ディス プレイだが、リダイレクト機能 を使えばファイルやプリンタな どに切り替えることができる。 C 言語の printf() 関数や Java 言語の System.out.println() は、テキストを標準出力に出力 する。

28 覚えよう!





③符号化. 011011010110111010...

解答

問24 問25

問26

- □ 27 3次元コンピュータグラフィクスに関する記述のうち、ポリゴンの説明はどれ か。 プ ある物体 A を含む映像 a から他の形状の異なる物体 B を含む映像 b へ. 滑らかに変化 する映像 ✓ コンピュータ内部に記録されているモデルを、ディスプレイに描画できるように2次元 ▶ 閉じた立体となる多面体を構成したり、2次曲面や自由曲面を近似するのに用いられた りする基本的な要素 ■ モデリングされた物体の表面に貼り付ける柄や模様などの画像 問 **28** E-R 図に関する記述として,適切なものはどれか。 ア関係データベースの表として実装することを前提に作成する。 ✓ 業務で扱う情報をエンティティ及びエンティティ間のリレーションシップとして表現す る。 データの生成から消滅に至るデータ操作を表現できる。 リレーションシップは、業務上の手順を表現する。 □ 29 関係データベース管理システム (RDBMS) におけるビューに関する記述のうち、 適切なものはどれか。 アビューとは、名前を付けた導出表のことである。
 - ✓ ビューに対して、ビューを定義することはできない。
 - ヴ ビューの定義を行ってから、必要があれば、その基底表を定義する。
 - ビューは一つの基底表に対して一つだけ定義できる。

問27 ポリゴン

3次元コンピュータグラフィクスで、物体を三角形や四角形などの図形の組 合せで表現する際、物体を構成する基本的な要素をポリゴンといいます。

- ×ア モーフィングの説明です。
- ×イレンダリングの説明です。
- ご 下解です。
- ×II 質感を表現するための柄や模様などをテクスチャといい。テクスチャ を物体の表面に貼り付ける処理をテクスチャマッピングといいます。



問28 E-R図 キホン!

E-R図は実体-関連図ともいい、分析対象から抽象したエンティティ(実体) と、エンティティ間のリレーションシップ(関連)を表した図です。

エンティティ"社員" とエンティティ "部署" の間 を、リレーションシップ"所属"で結んだ例。

- × ア E-R 図は関係データベースの設計によく利用されますが、業務過程の 分析やプログラム設計などにも幅広く用いられています。
- 1 正解です。
- × ウ データの生成や消滅は E-R 図では表現できません。
- ×
 エ
 リレーションシップは、エンティティ間の関連を表現します。

問29 ビュー

ビュー(仮想表)は、データベース上に実際に存在する表(実表)ではなく、 実表を加工して得られる表(導出表)に名前を付け、実表と同様に扱えるよう にしたものです。

- ア下解です。
- × イ ビューをさらに加工して、別のビューを作ることができます。
- ×ゥ ビューのもとになる表(基底表)が先に定義されていなければなりま せん。
- × I 1つの基底表から複数のビューを定義でき、複数の基底表から1つ ビューを定義することもできます。



20 覚えよう!

E-R図といえば

エンティティ間の関連をモ デル化した図

(関連)

実体

データベースに実際に記録され ている表(実表)を基に、演算 処理によって作成される仮想的 な表。CREATE VIEW 文で定 義する。

解答

問27 ウ 問28

問29

問 🚺 🚺 "注文"表と"製品"表に対して,次の SQL 文を実行したときに得られる結果 はどれか。

SELECT 製品名, 数量 FROM 注文, 製品 WHERE 注文. 製品コード = 製品. 製品コード

注力

仁义		
日付	製品コード	数量
2011-04-10	P2	120
2011-04-15	P1	100
2011-04-22	P4	50
2011-04-30	P8	80
2011-05-06	P1	100
2011-05-08	Р3	70

製品

製品コード	製品名
P1	PC
P2	テレビ
P3	掃除機
P4	冷蔵庫
P5	エアコン
P6	電話機
P7	時計

ア

製品名	数量
テレビ	120
PC	100
冷蔵庫	50
掃除機	70

イ

製品名	数量
テレビ	120
PC	200
冷蔵庫	50
掃除機	70

ウ

製品名	数量
テレビ	120
PC	100
冷蔵庫	50
PC	100
掃除機	70

製品名	数量
テレビ	120
PC	100
冷蔵庫	50
NULL	80
PC	100
掃除機	70

問 🛂 📘 表 R に対する次の SQL 文の操作はどの関係演算か。

K				
A1	A2	A3	A4	A5

〔SQL文〕

SELECT A1, A3, A5 FROM R

ア結合 イ差

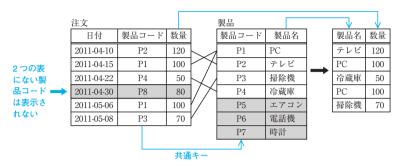
ウ射影

工 直積



問30 SQL文

問題の SQL 文は、"注文"表と"製品"表を製品コードを共通キーにして結 合し、製品名と数量を表示します。



× ア 重複する行が取り除かれています。この結果が得られる SQL 文は以下 のとおりです。

SELECT DISTINCT 製品名, 数量 FROM 注文, 製品 WHERE 注文.製品コード = 製品.製品コード

× 数量が集計されています。この結果が得られる SQL 文は以下のとおり です。

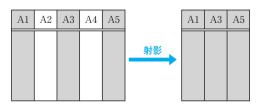
SELECT 製品名, SUM(数量) AS 数量 FROM 注文, 製品 WHERE 注文.製品コード = 製品.製品コード GROUP BY 製品名

- ウ 正解です。
- × 工 製品表にない製品コード「P8」の行も出力されています。この結果が 得られる SQL 文は以下のとおりです。

SELECT 製品名, 数量 FROM 注文 LEFT JOIN 製品 ON 注文.製品コード = 製品.製品コード

関係演算 キホン! 出31

問題の SQL 文は、表 R から A1、A3、A5 の 3 つの列だけを取り出します。 このような関係演算を射影といいます。



- × ア 結合は、複数の表を共通の項目をキーにして1つの表にします。
- ウ 正解です。
- \times \mathbf{I} 直積は、表1 の各行につき、表2 の各行を接続した表を作ります。新 しい表の行数は、表1の行数×表2の行数になります。



200 覚えよう!

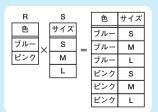
関係演算といえば

射影:特定の列を取り出す 選択:特定の行を取り出す 結合:2つの表を共通の項

目をキーにして結合する

● 直積

問 31 関係Rの1行につき、関係S のすべての行を組み合わせた集 合。



解答

問30 問31



	3	2 DBMS にお	けるログファイル	の説明として,	適切なもの	りはどれか。	
	7	上の更新データ	が発生したときにテ を定期的にディスク	に書き出したも	のである。		
	1		あってもシステムを サイトのデータベー			- 同一アータのコ	16-8
	ウ		らデータベースを回)内容をディスク	単位で
		複写したもので					
		データベースの[新記録を取った	回復処理のため,テ ものである。	゛ ータの更新前後	の値を書き出	:してデータベー	-スの更
	問 3	3 DBMSの排	他制御に関する記	述のうち、適切	刀なものはと	ごれか 。	
	ア	アクセス頻度のア	高いデータの処理速	度を上げるため	には、排他制	御が必要である	0
	1		るため、排他制御を				0
	ウ エ		時のデッドロックを に更新する可能性の				
	I	後数の八か門時	に史制りる可能性の	のあるケータには	, 护门巴即1447	が必安である。	
	問3 とする	時間は何秒	のデータを 100,0 か。ここで,回線				
	ア	200	400	9 800		1,600	
	3	5 LAN 間接続	装置に関する記述	めうち, 適切な	なものはどれ	いか。	
	ア		OSI 基本参照モデ			プロトコルを変	換する。
	1		アドレスを基にして			- 仁兴明離る延月	: - 2
	ウ エ		锺のセグメント間で アドレスを基にし			川ム医距離を延長	:90°
•	• • • • •	• • • • • • • •	anex	• • • • • • •			_
			解説		8	《一种》	のカギ
	当32	ログファイル	キホン!		C	Mr.	• • • • • • •

ログファイルは、データベースの更新前や更新後の値を書き出して、更新履歴として保存したファイルです。データベースに障害が発生した際、障害直前

22

の状態に回復させるために使用します。

- × **ア** チェックポイントファイルの説明です。
- × イ レプリケーションの説明です。
- × ウ バックアップファイルの説明です。
- I 正解です。

問33 排他制御

複数の利用者が同時に同じデータを更新しようとすると、更新が正常に行われない場合があります。DBMS はこれを防ぐため、ある利用者がデータを更新中のときは、ほかの利用者が同じデータにアクセスできないようにデータをロックします。この仕組みを排他制御といいます。

- ×アアクセス頻度は排他制御の必要性とは関係ありません。
- × **ラ デッドロック**は、複数の利用者が互いに相手のロック解除を待ち合い、 処理が停止してしまう現象です。デッドロックは排他制御の副作用で す。
- 正解です。

問34 転送時間の計算

回線の伝送効率が 50%なので、1 秒間に転送できるデータ量は 100,000 ビット× 50% = 50,000 ビットになります。10M バイト = 80M ビット = 80,000,000 ビットなので、転送時間は次のようになります。

80.000.000÷50.000 = 1.600 秒 ← 正解は エ です。

問35 LAN間接続装置 よく出る!

OSI 基本参照モデルは、データ通信を7つの層に切り分けて、それぞれの層での役割を規定しています。LAN 間接続装置も、OSI 基本参照モデルのどの層の通信を中継するかによって役割が異なります。

- \times **ア** ゲートウェイは、OSI 基本参照モデルの第 $1 \sim 7$ 層すべてのプロトコルを変換します。
- × ブリッジは、ネットワークインタフェース(NIC)に一意に割り振られている MAC アドレスを基にしてフレームを中継する装置です。
- 」 正しい記述です。リピータは OSI 基本参照モデルの第 1 層で、データ 通信の信号を物理的に中継します。
- × エ ルータは、IP アドレスを基にしてパケットを中継する装置です。



問 32

「参考」ログファイルはジャー
ナルファイルともいうよ。



営えよう! OSI基本参照モデルとLAN

解答

問32 三 問33

I

問	3	6 インターネ テキストだ	ットにおける電子メ けでなく,音声,画	ールの規約で, ヘッ: 「像なども扱えるよう」	ダフィールドの拡張を行い, こしたものはどれか。
	ア	HTML	MHS	• MIME	■ SMTP
問	3	7 プライベー てインター:	ト IP アドレスの複数 ネット接続を利用す	数の端末が, 一つのグ ・る仕組みを実現する:	ローバル IP アドレスを使っ ものはどれか。
	ア	DHCP	✓ DNS	ס NAPT	T RADIUS
問	3	8電子メールなものはど		るプロトコルである	POP3 の説明として,適切
	ア	PPP のリンク確 するプロトコル		パスワードによって利	J用者を認証するときに使用
	イ ウ			: 交換するときに使用す 電子メールを取り出すと	「るプロトコルである。 : きに使用するプロトコルで
	Ξ	利用者が電子メ	ールを送るときに使月	目するプロトコルである	0
問	3		ML の要素をどのよ ント側で処理される		示する場合に用いられ,表
	ア	CGI	← CSS	⊅ PHP	▼ SSI

解説

問36 電子メールの規約 よく出る!

インターネットの電子メールで、音声や画像ファイルなどの様々な形式のデータをやり取りするための規格をMIME(Multipurpose Internet Mail Extension)といいます。

- × <mark>ア HTML</mark> (HyperText Markup Language) は、インターネットの Web ページを記述するためのマークアップ言語です。
- × MHS (Message Handling System) は, OSI 基本参照モデルに基づ く電子メールサービスの国際規格です。
- ウ 正解です。
- × I SMTP (Simple Mail Transfer Protocol) は、インターネットの電子 メールの送信プロトコルです。

3、午前のカギ

営えよう! MIME といえば

- 電子メールで音声,動画などをやり取りするための規格
- データを文字データに変換 して送信。このときの符号 化方式を Base64 という

前

問37 プライベートIPアドレスの変換

LAN に接続されている各端末のプライベート IP アドレスを、グローバル IP アドレスに変換する機能を **NAT**(Network Address Translation)といいます。

NATでは、インターネットに同時に接続する端末の数だけグローバル IP アドレスを用意しなければなりませんが、IP アドレスとポート番号を組み合わせて、グローバル IP アドレスは 1 つだけ用意すればよい方式もあります。この方式を NAPT(Network Address and Port Translation),または IP マスカレードといいます。

- × プ DHCP (Dynamic Host Configuration Protocol) は、LAN に接続した端末に IP アドレスを自動的に割り当てるためのプロトコルです。
- × IP アドレスを変換する仕組みです。
- × **T RADIUS** (Remote Authentication Dial In User Service) は、ネットワークを介してアクセスするサーバ利用者を認証するためのプロトコルです。

問38 POP3 キホン!

インターネットの電子メールでは、届いたメールがメールサーバ上の各利用者のメールボックスに保存されます。POP3(Post Office Protocol version 3)は、メールサーバのメールボックスに保存されている電子メールを取り出すためのプロトコルです。

- × ア PPP の認証プロトコルは、PAP または CHAP です。
- × **SMTP** の説明です。
- ウ 正解です。
- × I POP3 は利用者が電子メールを受信するときに使用します。送信する ときのプロトコルは SMTP です。

問39 HTML/XML文書の表示指定

HTMLやXMLは、主に文書の内容や論理構造を記述する言語で、その内容をどのように表示するかについては最低限の指定しかできません。表示についての細かい指定には、CSS(Cascading Style Sheets)などのスタイルシート言語を利用します。

- × CGI は、Web サーバ上で外部プログラムを実行し、その結果を Web ページに取り込む仕組みです。
- × ウ PHP は、主に Web ページの作成に用いられるプログラム言語です。
- × I SSI は、HTML 文書に Web サーバ上で実行する外部プログラムを埋め込んでおき、その結果を Web ページに取り込む仕組みです。



答覚えよう!

NAT といえば

- プライベートIPアドレス とグローバルIPアドレス を相互に変換
- インターネットに接続する 端末の数だけグローバル IP アドレスが必要

NAPT (IP マスカレード) といえば

- プライベートIPアドレス とグローバルIPアドレス を相互に変換
- IP アドレスとポート番号と を組合せ、グローバル IP アドレスは 1 つでよい

≥ 覚えよう!

問 38

POP3 といえば

メール受信に使われるプロトコル

SMTP といえば

メール送信とメール転送に 使われるプロトコル

css 🌣

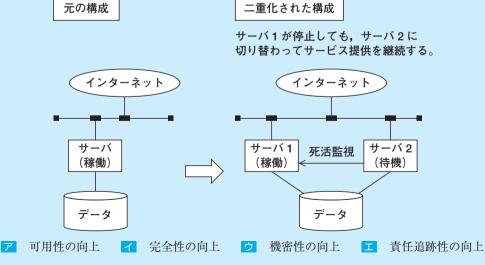
問

HTML や XML で記述された Web ページの見た目を定義す る言語。

解答

問36 **夕** 問37 **夕** 問38 **夕** 問39 **イ**

- 世 **40** 送信者から電子メール本文とそのハッシュ値を受け取り、そのハッシュ値と、 受信者が電子メール本文から求めたハッシュ値とを比較することで実現できる ことはどれか。ここで、受信者が送信者から受け取るハッシュ値は正しいものとする。
 - ア 電子メールの送達の確認
- 電子メール本文の改ざんの有無の検出
- ウ 電子メール本文の盗聴の防止
- エ なりすましの防止
- - ア 暗号化鍵と復号鍵は公開するが、暗号化アルゴリズムは秘密にしなければならない。
 - → 暗号化鍵は公開するが、復号鍵と暗号化アルゴリズムは秘密にしなければならない。
 - □ 暗号化鍵と暗号化アルゴリズムは公開するが、復号鍵は秘密にしなければならない。
 - 復号鍵と暗号化アルゴリズムは公開するが、暗号化鍵は秘密にしなければならない。
- - ア BCP の有効性を検証するためのテストを実施する。
 - 「情報システム障害時の代替手順と復旧手順について関係者を集めて教育する。
 - 情報システムに関する内外の環境の変化を踏まえて BCP の内容を見直す。
 - **工** 情報システムに許容される最大停止時間を決定する。
- □ **43** 図のようなサーバ構成の二重化によって期待する効果はどれか。



問40 ハッシュ値の利用 キホン!

ハッシュ値は、電子メール本文から**ハッシュ関数**を使って生成する固定長のデータです。同一データのハッシュ値は必ず同じ値になるので、受信者がメール本文から求めたハッシュ値と、送られてきたハッシュ値を照合すれば、メール本文が改ざんされていないかどうかを確認できます。

- ×ア ハッシュ値は送達の確認には利用できません。
- 11112345677899
- ×ゥ 盗聴の防止には、メール本文を暗号化します。
- × エ なりすまし防止にはディジタル署名などを利用します。

問41 公開鍵暗号方式

暗号化アルゴリズムとは、暗号を作る手順のことです。近年の暗号方式は暗号化や復号に「鍵」を用いることで、暗号化アルゴリズム自体は秘密にする必要がなくなっています。また、公開鍵暗号方式では、暗号化用の鍵と復号用の鍵が別々になっています。暗号化鍵で暗号を読むことはできないので、暗号化鍵は公開しても問題ありません。一方、復号鍵は暗号の受信者本人以外の人には秘密にしておく必要があります。以上から プが正解です。

問42 ビジネスインパクト分析

ビジネスインパクト分析とは、事故や災害などの不測の事態によって特定の業務がストップしたとき、それが事業全体にどのような影響を及ぼすかを分析することです。この分析結果に基づいて、事業継続計画 (BCP) を策定します。

- × ア BCP のテストは、BCP の策定後に行います。
- × イ 教育や訓練は、BCPの策定後に行います。
- × D BCPの見直しは、事業継続マネジメント(BCM)の一環として定期的に行います。
- I 正解です。業務停止の影響を測定するため、ビジネスインパクト分析では損失額や最大許容停止時間(業務の停止状態をどのくらいの期間まで許容できるか)を算出します。

問43 サーバ二重化の効果 ***ン!

サーバを二重化することで、サーバ1が故障で停止しても、サーバ2に切り替わってサービスを継続できます。このように、停止時間が短く、継続してサービスを提供できる能力を**可用性(アベイラビリティ)**といいます。

- ア 正解です。
- × イ 完全性とは、データが正確で、破壊や改ざんがないことです。
- × ウ 機密性とは、権限のない利用者がデータにアクセスできないことです。
- × **工 責任追跡性**とは、データがいつ・誰によって更新されたかといった、 履歴がたどれることをいいます。



※ 覚えよう! ■

改ざん検知といえば

- 送信者: 本文からハッシュ値を求め、本文といっしょに送信
- **受信者**: 受信した本文から ハッシュ値を求め, 受信し たハッシュ値と照合

200 覚えよう!

公開鍵暗号方式といえば

受信者の公開鍵で暗号化し、受信者の秘密鍵で復号

問 41

参考 暗号化鍵=公開鍵、復 号鍵=秘密鍵ということ。



◆事業継続計画(BCP)

P) 問 42

災害などの不測の事態が発生した場合の対応策や、事業が中断した場合の復旧手順などに関する計画のこと。BCPの策定と運用、見直しを組織的に行っていくことを、事業継続マネジメント(BCM)という。

登覚えよう! [

情報セキュリティのCIA といえば

機密性 (Confidentiality): 機密が保たれること

- 完全性 (Integrity): 情報 が正確なこと
- 可用性 (Availability): い つでも利用できること

解答

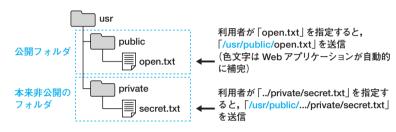
問40 <u>1</u> 問41 <u>ウ</u> 問42 <u>1</u> 問43 ア

84	4 通信を要求した PC に対し,ARP の仕組みを利用して実現できる通信可否の判定方法のうち,最も適切なものはどれか。
ア イ	PC にインストールされているソフトウェアを確認し、登録されているソフトウェアだけがインストールされている場合に通信を許可する。 PC の MAC アドレスを確認し、事前に登録されている MAC アドレスである場合だけ
	通信を許可する。 PCのOSのパッチ適用状況を確認し、最新のパッチが適用されている場合だけ通信を 許可する。
•	PC のマルウェア対策ソフトの定義ファイルを確認し、最新になっている場合だけ通信を許可する。
8 4	5 ディレクトリトラバーサル攻撃に該当するものはどれか。
ア イ	Web アプリケーションの入力データとしてデータベースへの命令文を構成するデータを入力し、想定外の SQL 文を実行させる。 Web サイトに利用者を誘導した上で、Web サイトの入力データ処理の欠陥を悪用し、
ウ	利用者のブラウザで悪意のあるスクリプトを実行させる。 管理者が意図していないパスでサーバ内のファイルを指定することによって、本来は許されないファイルを不正に閲覧する。 セッション ID によってセッションが管理されるとき、ログイン中の利用者のセッショ
	ンIDを不正に取得し、その利用者になりすましてサーバにアクセスする。
4	6 オブジェクト指向プログラミングの特徴はどれか。
ア	オブジェクトが相互にメッセージを送ることによって、協調して動作し、プログラム全 体の機能を実現する。
1	オブジェクトの外部からオブジェクトの内部のデータを直接変更できるので,自由度が 高い。
ウ エ	下位クラスは上位クラスの機能や性質を引き継ぐので、下位クラスに必要な性質は全て 上位クラスに含まれる。 個々のオブジェクトが使用するデータ(属性)は、あらかじめデータ辞書に登録してお
	ζ.
問 4 ジュー	階層構造のモジュール群から成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

ARP (Address Resolution Protocol) は、IP アドレスに対応する MAC ア ドレスを取得するプロトコルです。したがって、ARP を利用した通信可否の 判定方法としては、PCのMACアドレスを確認し、登録されたMACアドレ スかどうかで通信の可否を判定する方法が考えられます。正解はイです。

ディレクトリトラバーサル攻撃 > 初モノ! 出 45

ディレクトリトラバーサル攻撃は、管理者が意図していないパス名を指定し て、サーバ内の本来は許されていないファイルにアクセスする攻撃手法です。



- SQL インジェクション攻撃の説明です。
- X 1 クロスサイトスクリプティング攻撃の説明です。
- () ウ 正解です。
- セッションハイジャック攻撃の説明です。 XI

オブジェクト指向プログラミング 当46

オブジェクト指向プログラミングでは、プログラムは複数のオブジェクトと 呼ばれる部品で構成されます。オブジェクト同士はメッセージによって必要な 指示を送り、プログラム全体の機能を実現します。

○ ア 正解です。

- × イ オブジェクトへの指示はメッセージを送ることによって行い、オブジェ クト内部のデータを外部から直接変更することはできません。
- ×ゥ
 下位クラスは上位クラスの機能や性質を引き継ぎますが、上位クラス にはない機能や性質は、下位クラスで必要に応じて追加します。
- × I 個々のオブジェクトが使用するデータは、オブジェクトの内部に埋め 込まれます。データ辞書に登録する必要はありません。

スタブとドライバスよりは 出47

上位モジュールから下位モジュールへと順にテストを進めていく場合には、 上位モジュールから呼び出す仮の下位モジュールを用意しておく必要がありま す。この仮のモジュールをスタブといいます。正解は ウ です。

スタブの逆で、下位モジュールから上位モジュールへとテストを進めていく 場合に用意する仮の上位モジュールをドライバといいます。



♠ MACアドレス ネットワークインタフェースご とに一意に割り当てられている 48ビット長のアドレスのこと。 前半 24 ビットがベンダー ID 後半24ビットが機種IDとシ リアル番号を表す。

☆ 結合テスト

モジュール同士を組み合わせ て、正常に動作するかを確認す るテスト。上位モジュールから 順に結合している場合をトップ ダウンテスト,下位モジュール から順に結合している場合をボ トムアップテストという。

22 覚えよう!

スタブといえば

- テスト用の下位モジュール
- トップダウンテストで使う

ドライバといえば

- テスト用の上位モジュール
- ボトムアップテストで使う

解答

問44 問45 問46 問47

	4	8 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。
		ソースコードを解析し、プログラムの制御の流れと変数間のデータの流れのテストを、 主にプログラム開発者以外の第三者が実施する。
	1	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を, 主にプログラム開発者以外の第三者が実施する。
	ウ	プログラムの全ての命令について最低1回は実行することを完了の条件とするテスト
		を、主にプログラム開発者自身が実施する。
	I	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を, 主にプログラム開発者自身が実施する。
	84	ります。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
	8 5	● ソフトウェアのリバースエンジニアリングの説明はどれか。
	ア	CASEツールなどを用いて、設計情報からソースコードを自動生成する。
	イ ウ	外部から見たときの振る舞いを変えずに、ソフトウェアの内部構造を変える。 既存のソフトウェアを解析し、その仕様や構造を明らかにする。
	Ξ	既存のソフトウェアを分析し理解した上で、ソフトウェア全体を新しく構築し直す。
		問 51 から問 60 までは,マネジメント系の問題です。
L		
	8 5	ソフトウェア開発プロジェクトで行う構成管理の対象項目として、適切なものはどれか。
	ア	開発作業の進捗状況
		成果物に対するレビューの実施結果 プログラムのバージョン
		プロジェクト組織の編成

問48 ブラックボックステスト よく出る!

ブラックボックステストでは、プログラムが仕様どおりの機能を実現しているかどうかを、内部のロジックは考慮せずに検証します。内部のロジックを検証するテストは、ブラックボックステストに対してホワイトボックステストといいます。

ホワイトボックステストは、主としてプログラム開発者自身が担当しますが、 ブラックボックステストはプログラムの利用者など、主としてプログラム開発 者以外の第三者が担当します。

以上から,正解は 1です。

問49 値呼出しと参照呼出し

サブルーチンで処理する値を引数で渡す際に、値そのものを渡す方法を値呼出し (call by value)、値が格納されている変数のアドレスを渡す方法を参照呼出し (call by reference) といいます。

値呼出しでは、引数に変数を指定する場合でも、実際にはその変数に格納されている値をコピーして指定するのと同じなので、サブルーチン実行後に変数の値が変更されることはありません。以上から、正解はフです。

なお、結果呼出し(call by result)は、サブルーチンで処理された結果を呼出し元に戻る際に指定された引数にコピーする方式です。また、名前呼出し(call by name)は、呼び出されたサブルーチン自体を呼び出した場所にコピーする方式です。

問50 リバースエンジニアリング

ソフトウェア開発は、通常は設計→プログラミング→運用の順に進めます。 リバースエンジニアリングはこの順序を逆転(リバース)し、できあがったソフトウェアからソースコードを解析したり、ソースコードから仕様や構造を復元する手法です。

- × ア 設計情報からソースコードを生成するのはリバースではなく,通常の エンジニアリングの工程です。
- × **イ** リファクタリングの説明です。
- ウ 正解です。
- × **エ** リエンジニアリングの説明です。

問51 プロジェクトの構成管理

構成管理とは、開発過程でできる様々な文書やソースコードを管理することです。開発が進捗するにつれ、設計書やソースコードに多くのバージョンができるので、それらの組合せを管理するバージョン管理が重要になります。正解は ってす。



答覚えよう!

問 48

ブラックボックステスト といえば

- プログラムの外部仕様に基 づいてテスト
- 同値分割法、限界値分析などの方式がある

ホワイトボックステスト

といえば

- プログラムの内部構造に基づいてテスト
- 条件網羅, 判定条件網羅, 分岐網羅などの方式がある

問 49

参考 結果呼出しと名前の呼出しは、基本情報技術者試験では覚えておく必要はないよ。



CASE ツール 図50 CASE は Computer Aided Software Engineering の略。 ソフトウェア開発を支援する各種の開発ツールのこと。要求分析や外部設計などを支援する上流 CASE ツールと,プログラミングやテストなどを支援する下流 CASE ツールがある。

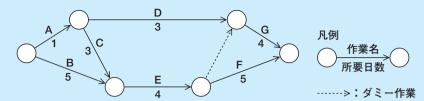
問 51

<mark>対策</mark> 構成管理=バージョン 管理と覚えておこう。



· 解答

問48 <u>イ</u> 問49 ア 問50 ウ 問51 ウ □ **52** プロジェクトのタイムマネジメントのために次のアローダイアグラムを作成した。クリティカルパスはどれか。



- \nearrow A \rightarrow C \rightarrow E \rightarrow F
- $B \to E \to F$

- $A \rightarrow D \rightarrow G$
- \blacksquare B \rightarrow E \rightarrow G

問 53 設計書の作成状況が表のとおりであるとき、3種類の設計書全ての作成を完了させるために必要な今後の工数(人時)は幾らか。

設計書	作成枚数	1枚当たりの所要工数	現在までの作成済み枚数
設計書	(枚)	(人時)	(枚)
基本設計書	80	5	80
概要設計書	300	2	200
詳細設計書	500	2	50

- プ 550
- 900
- **D** 1100
- **=** 2000

- ア WBS を構成するワークパッケージの完了数
- ✓ 個人別のプログラミングの生産性
- ウ 成果物ごとのレビュー時間
- **エ** プログラムのバージョン

□ **55** IT サービスマネジメントの管理プロセスはどれか。

ア サービスレベル管理

イ スケジュール管理

ウ 品質管理

I

11 フカ答冊

解説

問52 アローダイアグラム ***ン!*

クリティカルパスは、プロジェクトの開始から終了までの各作業のうち、時



前

間的余裕のない一連の作業経路をいいます。クリティカルパス上にある各作業 は、1日でも遅れると全体の進捗の遅れにつながってしまいます。

アローダイアグラムでは、開始から終了までの作業経路のうち、最も所要日 数の多い経路がクリティカルパスです。したがって、各選択肢の作業経路のう ち、所要日数の合計が一番多いものが正解となります。

 \times 7 1 + 3 + 4 + 5 = 13 \oplus \times 1 + 3 + 4 = 8 \oplus

 \bigcirc 5 + 4 + 5 = 14 \Box

 \times 5 + 4 + 4 = 13 \boxminus

153 工数計算

作成枚数と、現在までの作成済み枚数から、今後必要な作成枚数を求めます。

作成枚数 作成済み枚数 必要な作成枚数

基本設計書: 80 0枚 80 概要設計書: 300 200 100 枚 詳細設計書: 500 50 450 枚

1枚当たりの所要工数は、概要設計書、詳細設計書ともに2人時です(1人 時は、1人で作業すると1時間かかる工数)。したがって、今後必要な工数は 次のようになります。

100×2 + 450×2 = 200 + 900 = 1100 人時

概要設計書 詳細設計書

以上から、正解はってす。

問54 ソフトウェア品質の管理

工程の各段階でできる成果物を、次の工程に進む前に客観的に検証する手続 きをレビューといいます。選択肢の中では、成果物ごとのレビュー時間が、ソ フトウェアの品質に関わる指標になります。

× ア ワークパッケージの完了数は、進捗管理の指標です。

× d 個人別の生産性は、進捗管理や成績管理の指標です。

○ ウ 正解です。

× エ プログラムのバージョンは、構成管理の指標です。

問**55** IT サービスマネジメント

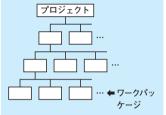
業務システムなどのITサービスを組織的に運用管理していくことをITサー ビスマネジメントといい、ITIL という国際的なガイドラインが定められてい ます。

サービスレベル管理は、サービス提供者とサービス利用者との間で合意した サービスレベルを維持・管理するプロセスで、ITIL に規定されているサービ スデリバリ(中長期的な IT サービス計画)の5つのプロセスの1つです。以 上から、正解はアです。



WBS (Work Breakdown Structure)

プロジェクト全体を階層的に細 かく分割して管理していく手 法。細分化された最小単位の作 業をワークパッケージという。



₫ ITIL

IT サービスマネジメントのガイ ドライン。現在は国際標準化機 構による国際規格「ISO/IEC 200001となっている。

サービスサポートといえば

①インシデント管理

200 覚えよう!

2問題管理

③構成管理

4変更管理

⑤リリース管理

⑥サービスデスク

サービスデリバリといえば

①サービスレベル管理

② IT サービス財務管理

③可用性管理

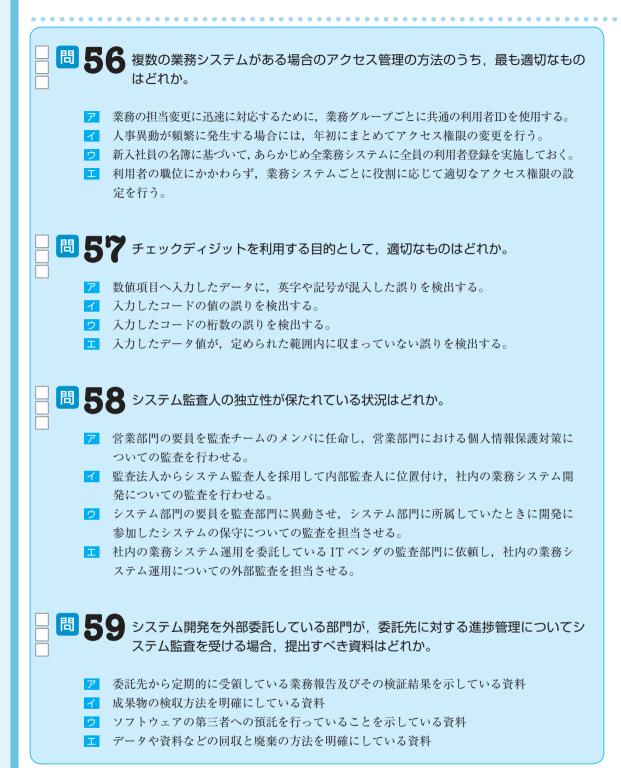
④ IT サービス継続性管理

⑤キャパシティ管理

解答

問52 問53

問54 問55



解説

問**56** アクセス管理の手法

アクセス権限は、必要な利用者に、必要な範囲のみ与えるのが原則です。し



前

たがって複数の業務システムがある場合には、人事異動のたびに業務システムでとに適切なアクセス権限を設定し直す必要があります。異動先の業務システムの利用者 ID を新たに発行するだけでなく、旧部署で使用していた利用者 ID を無効にするといった措置も重要です。

- × 7 複数の利用者が利用者IDを共有すると、不正アクセス等の問題が発生 しても誰がアクセスしたか特定できず、セキュリティ上問題があります。
- × イ 年初まで,アクセス権限が適切に設定されないまま運用しなければならず,不都合です。
- × **ウ** 新入社員が、自分の業務外の業務システムを利用できるようなるため、 セキュリティ上問題があります。
- I 正解です。

問57 チェックディジット ***ン!

チェックディジットは、コードから一定の規則に従って算出した数値を、そのコードに付加したものです。システムはコードが入力されると、そのコードからチェックディジットを計算し、入力されたものと照合します。もし一致しなければ、入力されたコードに誤りがあることがわかります。以上から、正解は 1です。

①本来のコードからチェックディジットを算出 12345 ➡ 6

②入力されたチェックディジットと照合して, 入力ミスがないかどうかを確認

問58 システム監査人の独立性 キホンパ

システム監査人の独立性とは、監査人と監査対象との間に、公正な監査を疑われるような利害関係がないことです。自分が所属している部署を自分で監査するケースや、監査対象が監査人の上司であるケース、納入したシステムのシステム監査を業者が自ら受託するケースなどは、独立性に問題があります。

- ×ア 営業部門の要員が営業部門を監査するので、独立性に問題があります。
- システム監査人は内部監査人として、他の部署から独立していると考えられるので、独立性は保たれています。
- × ウ システム監査人が以前に開発したシステムが監査対象なので、独立性 に問題があります。
- × IT ベンダが運用を受託している業務システムを、IT ベンダが自ら監査するので、独立性に問題があります。

問59 委託先に対する進捗管理

進捗管理についてのシステム監査なので、委託している業務の進捗状況を把握するために利用する文書が監査資料となります。

選択肢のなかでは、アの委託先からの業務報告とその検証結果が、委託先の業務の進捗状況を把握していることを示す文書になります。



コード番号から算出した数値を、そのコードに付加したもの。システムは、入力されたコードからチェックディジットを算出し、それを付加されたチェックディジットと照合して入力ミスがないかどうかを検出する。

問 57

参考 チェックディジットは 書籍の ISBN コードの末尾に も付いているよ。



問 58

<u>参考</u> システム監査人の独立 性には、

- ①公正かつ客観的な判断ができること(精神的な独立性)
- ②第三者から見て、信頼がおける中立的な立場を維持していること(外見上の独立性)が必要だよ。



🔷 内部監査

組織の内部にある監査部門が行う監査のこと。

解答

問56 <u>I</u> 問57 <u>イ</u> 問58 <u>イ</u> 問59 ア

8 6	システム監査において、監査証拠となるものはどれか。
ア イ ウ	システム監査チームが監査意見を取りまとめるためのミーティングの議事録 システム監査チームが監査報告書に記載した指摘事項 システム監査チームが作成した個別監査計画書
	システム監査チームが被監査部門から入手したシステム運用記録
	問 61 から問 80 までは,ストラテジ系の問題です。
8	リスクや投資価値の類似性で分けたカテゴリごとの情報化投資について、最適な資源配分を行う手法はどれか。
	3C 分析
	IT ポートフォリオ エンタープライズアーキテクチャ
	ベンチマーキング
8 6	と 情報戦略の立案時に、必ず整合性をとるべき対象はどれか。
	新しく登場した情報技術
	基幹システムの改修計画 情報システム部門の年度計画
	中長期の経営計画
6	るSPとは,どのようなサービスを提供する事業者か。
ア	顧客のサーバや通信機器を設置するために、事業者が所有する高速回線や耐震設備が 整った施設を提供するサービス
1	顧客の組織内部で行われていた総務、人事、経理、給与計算などの業務を外部の事業者
Ċ	が一括して請け負うサービス 事業者が所有するサーバの一部を顧客に貸し出し、顧客が自社のサーバとして利用する サービス
	汎用的なアプリケーションシステムの機能をネットワーク経由で複数の顧客に提供する サービス

問60 監査証拠

監査証拠とは、システム監査によって判明した事項の裏付けとなる客観的な 資料のことです。具体的には、被監査部門のサーバから抽出したデータや、入 手した文書、アンケート結果などが監査証拠となります。

選択肢の中では、

のシステム運用記録だけが、被監査部門から入手した
資料です。ほかはいずれもシステム監査チームの作成した文書なので、客観的
な証拠とはなりません。

問61 情報化投資の手法 | 初モノ!

ハイリスク/ハイリターンの投資と、ローリスク/ローリターンの投資を複数組み合わせて、全体としてリスクとリターンのバランスをとる資産運用の手法を、ボートフォリオといいます。この手法を応用して、情報化投資への資源配分を最適化する手法をITボートフォリオといいます。

- × <mark>ア 3C分析</mark>は、市場 (Customer)、競合 (Competitor)、自社 (Company) の 3 つの視点から現状を分析する経営手法です。
- 1 正解です。
- エンタープライズアーキテクチャ(EA)は、組織全体をビジネス、データ、アプリケーション、テクノロジの4つの体系(アーキテクチャ)でとに分析し、最適化を図る手法です。

問62 情報戦略の立案

情報戦略とは、経営戦略を実現するために、どのような情報システムが必要となるかを立案することです。そのためには、目先の現状に対応するのではなく、中長期の経営計画との整合性を保つことが重要となります。

- ×ア 最新の情報技術に追随する必要はありません。
- × イ, ウ 基幹システムの改修計画や、情報システム部門の年度計画は、情報戦略の策定後に立案します。
- I 正解です。

問63 ASP よく出る!

ASP(Application Service Provider)は、アプリケーションの機能をネットワーク経由で顧客に提供するサービスです。インターネットには、電子メールやスケジュール管理など、様々なサービスを提供する ASP があります。

- × プ IDC (Internet Data Center) の説明です。
- × BPO (Business Prooess Outsourcing) の説明です。
- × **ウ サーバホスティング**(レンタルサーバ)の説明です。
- 工 正解です。



登 覚えよう!

システム監査の手順

- ①監査計画の立案
- ②予備調査:アンケート調査など
- ③本調査:監査証拠を入手
- ④**評価・結論**:監査報告書を 作る
- ⑤フォローアップ:監査結果に基づき,適切な指導を行う

○ 解答

問60 三 問61 問62 三 問63

問 64 企業が保有する顧客や市場などの膨大なデータから、有用な情報や関係を見つ け出す手法はどれか。 ア データウェアハウス データディクショナリ **ウ** データフローダイアグラム データマイニング 65 サプライチェーンマネジメントの改善指標となるものはどれか。 ア 残業時間の減少率 1 販売単価下落の防止率 ウ 不良在庫の減少率 ■ 優良顧客数の増加率 問 **6** 図に示す手順で情報システムを調達する場合, b に入るものはどれか。 発注元はシステム化の目的や業務内容などを示し、 а 調達先に情報提供を依頼する。 発注元は調達対象システム、調達条件などを示し、 b 提案書の提出を依頼する。 発注元は提案書,能力などに基づいて,調達先を С 決定する。 発注元と調達先の役割や責任分担などを、文書で Ы 相互に確認する。 7 RFI 1 RFP ウ 供給者の選定 工 契約の締結 プロダクトポートフォリオマネジメント (PPM) における "花形" を説明した ものはどれか。 市場成長率、市場占有率ともに高い製品である。成長に伴う投資も必要とするので、資 金創出効果は大きいとは限らない。 → 市場成長率,市場占有率ともに低い製品である。資金創出効果は小さく、資金流出量も 少ない。 **ウ** 市場成長率は高いが、市場占有率が低い製品である。長期的な将来性を見込むことはで きるが、資金創出効果の大きさは分からない。 ■ 市場成長率は低いが、市場占有率は高い製品である。資金創出効果が大きく、企業の支 柱となる資金源である。

問64 データ分析の手法 よく出る!

大量に蓄積されたデータから、統計的・数学的手法などを用いて有用な情報や関係を見つけ出す手法をデータマイニングといいます。

- データウェアハウスは、蓄積された大量の情報を時系列的にデータベース化したものです。
- × **ゴ** データディクショナリは、DBMS が管理するデータや利用者、プログラムなどの情報を一箇所に集めて管理するためのデータの集合体です。
- データフローダイアグラム (DFD) は、業務過程をデータの流れに着目して図式化したものです。
- I 正解です。

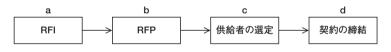
問65 サプライチェーンマネジメント

サプライチェーンマネジメント (SCM) は、生産から購買、販売、物流に至る一連の業務(サプライチェーン)を全体で最適化し、リードタイムの短縮や 在庫削減を図る手法です。

SCM の導入によって余分な在庫が削減されるので、 の不良在庫の減少率は、SCM の効果を測るための改善指標となります。

問66 情報システムの調達手順 キホン!

発注元が調達条件を示して、業者に対し提案書の提出を依頼する文書です。 このような文書を RFP (Request For Proposal) といいます。正解は です。 なお、発注元が業者に対し情報提供を依頼する文書は RFI (Request For Information) といいます。



問67 プロダクトポートフォリオマネジメント よく出る!

プロダクトポートフォリオマネジメント (PPM) は、市場成長率、市場占有率という2つの評価軸から、自社の事業を「花形」「金のなる木」「問題児」「負け犬」の4種類に分類する分析手法です。

「花形」は市場成長率,市場占有率がともに高い事業で,資金の流入をもたらしてくれますが,さらなる成長のために資金の投入も必要となるため,資金 創出効果は大きいとは限りません。

- ア 正解です。
- × イ 「負け犬」の説明です。
- × ウ 「問題児」の説明です。
- × I 「金のなる木」の説明です。



問 64

「<mark>参考</mark> データマイニングは 「データ発掘」という意味だね。



29 覚えよう!

RFP (提案依頼書)

といえば

外部業者にシステム見積や 提案を依頼

RFI(情報提供依頼書)

といえば

外部業者に情報提供を依頼

≥ 覚えよう!

プロダクトポートフォリオマネジメントといえば



解答

問64 三 問65

問66 1 問67

ゥ

8 6	3 コアコンピタンス経営を説明したものはどれか。
ア イ ウ	迅速な意思決定のために、組織の階層をできるだけ少なくした平型の組織構造によって 経営を行う。
I	他社にはまねのできない,企業独自のノウハウや技術などの強みを核とした経営を行う。
8 6	9 バランススコアカードの顧客の視点における戦略目標と業績評価指標の例はどれか。
ア イ ウ エ	製品開発力の向上が目標であるので、製品開発領域の研修受講時間を指標とする。
8 7	り ワンチップマイコンの内蔵メモリにフラッシュメモリが採用されている理由として、適切なものはどれか。
ア イ ウ エ	マイコン出荷後もソフトウェアの書換えが可能である。 マイコンの処理性能が向上する。
問 7	1 ①~③の手順に従って処理を行うものはどれか。
2 (2) 5	今後の一定期間に生産が予定されている製品の種類と数量及び部品構成表を基にして、の構成部品についての必要量を計算する。 別き当て可能な在庫量から各構成部品の正味発注量を計算する。 製造/調達リードタイムを考慮して構成部品の発注時期を決定する。
7	CAD CRP JIT I MRP

問68 コアコンピタンス経営 よく出る!

コアコンピタンス経営とは、他社にまねのできない独自のノウハウや技術を 核に据えて経営を行う経営手法です。

- × ア ナレッジマネジメントの説明です。
- ×イフラット組織の説明です。
- × ウ ベンチマーキングの説明です。
- I 正解です。

問69 バランススコアカード

バランススコアカードは、自社の経営戦略を①財務、②顧客、③内部ビジネスプロセス、④学習と成長の4つの視点から分析し、具体的な目標や施策を策定する経営管理手法です。

- ×ア 財務の視点における目標と指標です。
- ✓ 顧客の視点における目標と指標です。
- ×ゥ 学習と成長の視点における目標と指標です。
- ×

 内部ビジネスプロセスの視点における目標と指標です。

問70 ワンチップマイコンの内蔵メモリ

ワンチップマイコンは、CPU、メモリ、入出力機能などを1つのチップ上に搭載したマイクロコンピュータです。

フラッシュメモリは内容の消去や再書込みが可能なので、バージョンアップや不具合などによってマイコンの出荷後にソフトウェアの書換えが必要になった場合、フラッシュメモリであれば書換えが可能であるという利点があります。以上から、正解は 1です。

問71 生産計画の手法 よく出る!

①~③の手順で処理を行うのは、MRPと呼ばれる生産計画の手法です。 MRP (Material Requirements Plannning, 資材所要量計画) は、製品の製造に必要な資材の発注量や発注時期を、生産計画に基づいて適切に管理する手法です。

- × **7 CAD** (Computer Aided Design) は、コンピュータを利用した設計です。
- × CRP (Continuous Replenishment Program) は、チェーンの小売店で販売された数量に応じて必要在庫量を算出し、自動的に補充するシステムです。
- × JIT (Just In Time) は、必要な製品を必要な時に、必要な量だけ生産することで、各工程における在庫量を最小にする生産方式です。
- 工 正解です。



≥ 覚えよう!

コアコンピタンス経営

といえば

• 自社独自のノウハウや技術 に集中

ナレッジマネジメント といえば

- 知識を集約して共有
- フラット組織といえば
- 階層をできるだけ少なくした組織

ベンチマーキングといえば

先進企業と自社を比較する

営 覚えよう!

バランススコアカードの 4 つの視点といえば

- ・財務
- 顧客
- 内部ビジネスプロセス
- 学習と成長

≥ 覚えよう! ▮

MRP といえば

部品構成や在庫量から、必要な資材の発注量を算出

解答

問68 I 問69 1 問70 1 問71 I

生産方式	a	b	С	d
生産量	少	中	多	多
主な生産形態	受注生産	受注・見込生産	見込生産	受注・見込生産
生産品種	多	中	少	少
段取り頻度	多	中	少	極少
処理 (例)	組立て	組立て	組立て	化学的

- ア 個別生産
- プロセス生産
- ウ 連続生産
- エ ロット生産

- 問 **73** IC タグ (RFID) の特徴はどれか。
 - ア GPS を利用し、受信地の位置情報や属性情報を表示する。
 - 専用の磁気読取り装置に挿入して使用する。
 - ウ 大量の情報を扱うので、情報の記憶には外部記憶装置を使用する。
 - **エ** 汚れに強く、記録された情報を梱包の外から読むことができる。
- □ 門4 パレート図を説明したものはどれか。
 - ア 原因と結果の関連を魚の骨のような形態に整理して体系的にまとめ、結果に対してどの ような原因が関連しているかを明確にする。
 - → 時系列的に発生するデータのばらつきを折れ線グラフで表し、管理限界線を利用して客観的に管理する。
- - 条件 $4x + 8y \ge 40$ $9x + 6y \ge 54$ $x \ge 0, y \ge 0$ 目的関数 $2x + 3y \rightarrow$ 最大化
 - 夕 条件 $4x + 9y \ge 40$ $8x + 6y \ge 54$ $x \ge 0, y \ge 0$ 目的関数 $2x + 3y \rightarrow$ 最大化
- 条件 $4x + 8y \le 40$ $9x + 6y \le 54$ $x \ge 0, y \ge 0$ 目的関数 $2x + 3y \rightarrow$ 最大化

問**72** 生産方式

個別生産は、顧客の注文に応じて個別に生産する方式です。製品は顧客ごと に異なるため、生産品種が多く、生産量は少なくなります(空欄 a)。

プロセス生産(バッチ生産)は、原料を精製・混合したり、化学的に加工し たりして製品にする生産方式です。設備があれば一度に大量に生産できるので、 丁数が少ないわりに生産量は多く、生産品種は少なくなります(**空欄 d**)。

連続生産は、ひとつの品目を一定期間連続して生産し続ける生産方式です。 生産品種が少なく、生産量が多い製品の生産に向いています(空欄 c)。

□ット生産は、同じ製品をロットと呼ばれる単位で生産する方式です。異な る製品をロットごとにまとめて生産するので、生産品種は連続生産より多く、 生産量は個別生産より多くなります(空欄 b)。

以上から、正解はアです。

問**73** IC タグ(RFID)

RFID (Radio Frequency IDentification) は, 超小型の IC チップに情報を 記録し、電波による無線で情報をやり取りする技術です。この IC チップに商 品の情報を記録して商品や梱包に付けたものを IC タグといいます。

IC タグは無線で情報をやり取りするため汚れに強く、梱包の外からでも情 報を読むことができるのが特徴です。正解はエです。

問74 パレート図 ★ 出る!

パレート図は、各項目を出現頻度順に並べた棒グラフと、その累積比率を表 す折れ線グラフを組み合わせ、各項目の重要度を分析する図です。

- × ア 特性要因図(フィッシュボーン図)の説明です。
- × イ 管理図の説明です。
- × ウ ヒストグラムの説明です。
- 工 正解です。

出75 線形計画問題

製品 A, B の生産量をそれぞれ x トン, y トンとすると, 原料 P の必要量は 4x + 8y トン、原料 Q の必要量は 9x + 6y トンです。原料 P は 40 トン、原料 Qは54トンしかないので、次の条件が成り立ちます。

$4x + 8y \le 40$

 $9x + 6y \le 54$

 $x \ge 0, y \ge 0$

製品 A を x トン生産, 製品 B を y トン生産したときの利益は, 2x + 3y と 表せます。この値が最大になるようなx, yの値を求めるので、目的関数は、 $2x + 3y \rightarrow$ 最大化のようになります。

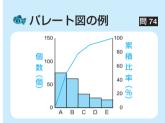
以上から、正解は \checkmark です(ちなみに、利益は製品 A を 4 トン、製品 B を 3 トン生産したときに最大になります)。

午前のカギ

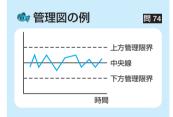
29 覚えよう!

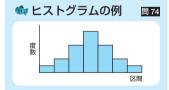
IC タグ (RFID) といえば

無線電波により非接触で情 報を読み取る









輸線形計画問題 問 75 与えられた制約条件のもとで. 最大の成果を得るための解を求 める手法。



図の損益計算書における経常利益は何百万円か。ここで、枠内の数値は明示し ていない。

単位 百万円

	一一一 日 23 1 3
	_
I. 売上高	1,585
Ⅱ. 売上原価	951
Ⅲ. 販売費及び一般管理費	160
Ⅳ. 営業外収益	80
Ⅴ. 営業外費用	120
Ⅵ. 特別利益	5
Ⅷ. 特別損失	15

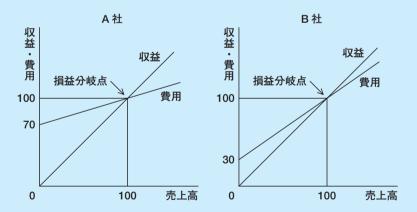
ア 424

434

9 474

= 634

図は二つの会社の損益分岐点を示したものである。A 社と B 社の損益分析に関 する記述のうち、適切なものはどれか。



- ▼ A 社, B 社ともに損益分岐点を超えた等しい売上高のとき, 固定費の少ない B 社の方 がA社よりも利益が大きい。
- ✓ A 社は B 社より変動費率が小さいので、損益分岐点を超えた等しい売上高のとき、B 社に比べて利益が大きい。
- **| 両社の損益分岐点は等しいので、等しい利益を生み出すために必要な売上高は両社とも**
- 両社は損益分岐点が等しく、固定費も等しいので、売上高が等しければ、損益も等しい。

問76 経営利益 ***ン!

経常利益は、特別損益を含まない利益なので、次のように求めます。

販売費及び 十営業外収益一営業外費用 経常利益=売上高-売上原価-般管理費

売上総利益

営業利益

したがって、経常利益は 1585 - 951 - 160 + 80 - 120 = 434 百万円となり ます。正解はイです。



問77 損益分析

損益分岐点は、売上高と総費用が一致するポイントです。売上高が損益分岐 点を上回れば利益が発生し、下回れば損失が発生します。

- × プ 売上高ゼロのときの費用が固定費です。図より、固定費はたしかに B 社のほうが少ないですが、同じ売上高で比べると、利益(収益-費用) は A 社のほうが大きくなっています。
- 1 変動費率は、売上高に占める変動費の割合です。費用から固定費を差 し引いたものが変動費になります。損益分岐点の売上高で求めると、 A 社の変動費率は (100 − 70) ÷ 100 = 0.3, B 社の変動費率は (100 − 30) ÷ 100 = 0.7 となり、A 社のほうが小さくなります。変動費率が小 さいほど、売上に占める費用の割合が小さいので、利益は大きくなり ます。
- × ウ 利益は収益 費用で求めます。2 つのグラフを比較すると、売上高が 等しい場合の利益は、A 社のほうが大きくなります。
- × 工 売上高や固定費が同じでも、変動費率が異なれば利益は異なります。



200 覚えよう!

指益計算といえば

売上総利益 -

売上高一売上原価

営業利益 -

売上総利益一販管費

※販管費:販売費及び一般管理費

経常利益 -

党業利益十党業外指益

税引前当期利益

経常利益十特別損益

当期利益 -

税引前当期利益-法人税•住民税

200 覚えよう!

損益分岐点といえば

- 売上高と総費用が等しくな る利益ゼロのポイント
- で求める。

🔷 変動費

問 77

売上高に比例して増減する費 用。材料費や運搬費など。

● 固定費

売上高に関係なく発生する一定 の費用。人件費や賃借料など。

슚 変動費率

売上高に占める変動費の割合 (変動費÷売上高)。

解答

問76 1 問77

問	7	8 著作権法に れか。	よるソフトウェアの	D保護範囲	に関する記述の	かうち,適切な	ものはど
	ア イ ウ エ	ラムは権利の対 アルゴリズムや アルゴリズムを	ンプログラムは著作 価がハードウェアの プログラム言語は, 記述した文書は著作れ ムとオブジェクトプロ	料金に含まれ 著作権法によ 権法で保護さ	れるので、保護 にって保護される されるが、プロク	されない。 る。 ブラムは保護され	ない。
問	7		り扱う商品やサーと , 記号など(識別機				るための
	ア	意匠法	✓ 商標法	ウ特	許法	工 著作権法	
問	8		法の対象となる制術 ウェアはエレベータ				
	アイウエ	エレベータの可 エレベータメー	ち時間が長くなる原見動部分の交換を早める カの出荷作業の遅延の エレベータ事故の原見	る原因となっ	た不具合た不具合		

問78 著作権法 キホン!

- ×ア 基本プログラムも著作権法の保護の対象となります。
- × プログラムは著作権で保護されますが、プログラムの作成に使用する アルゴリズムやプログラム言語は、著作権法の保護の対象外です。小 説は著作権法で保護されても、文法や日本語に著作権がないのと同様 です。
- × フルゴリズムを記述した文書は、著作物なので保護の対象となります。 同様に、プログラムも保護の対象となります。
- I 正しい記述です。ソースプログラムも、それを機械語に変換したオブ ジェクトプログラムも、著作権法によって保護されます。

問79 知的財産権 キホン!

商品名やブランド名, ロゴマーク, 会社名といった, 商品やサービスの識別 標識は, **商標法**によって保護されます。

- ×ア 意匠法は、商品の外観やデザインを保護する法律です。
- 1 正解です。
- × **工 著作権法**は、著作物(思想・感情を創作的に表現したもの)を保護する法律です。

問80 製造物責任法

製造物責任法 (PL 法) は、製造物の欠陥が原因で、他人の生命、身体または財産を侵害した場合に、製造業者が損害賠償の責任を負うことを定めた法律です。ソフトウェアは本来製造物責任法の対象とはなりませんが、ソフトウェアを組み込んだ制御装置は、製造物責任法の対象となります。

- ×ア 待ち時間が長くなる程度では、損害賠償の対象にはなりません。
- × 「損害が当該製造物についてのみ生じたとき」は、製造物責任の対象外となります。欠陥がエレベーターの別の不具合の原因となっているだけなので、製造物責任の対象とはなりまりません。
- ×ウ製造業者自身の損害は対象外となります。
- 工 正解です。人的被害の原因となった欠陥なので、製造業者の製造物責任が問われます。



登 覚えよう!

産業財産権といえば

- 特許権:発明に関する権利
- 実用新案権: 実用的なアイ ディアに関する権利
- 意匠権:製品のデザインな どに関する権利
- 商標権:商品名やロゴなどに関する権利
- ※著作権は、産業財産権には含まれな

28 覚えよう!

製造物責任法といえば

製造業者が、製造物の欠陥 による損害の責任を負う

○ 解答

問78 🔳 問79

問80

47

平成 24 年度 春 午後問題

次の問 1 から問 7 までの 7 問については、この中から 5 問を選択し、選択した問題については、答案用紙の選択欄の(選)をマークして解答してください。

なお、6 問以上マークした場合には、はじめの 5 問について採点します。

問 🧻 デ動小数点数に関する次の記述を読んで,設問 1,2 に答えよ。

(1) $\alpha \times 2^{\beta}$ の形で表記される浮動小数点数を, **図 1** に示す 32 ビット単精度浮動小数点形式 (以下, 単精度表現という) で表現する。ここで, α と β は次の条件を満たすものとする。

 α = 0, \mathbb{Z} if $1 \le |\alpha| < 2$ - $126 \le \beta \le 127$

04 00 00 00 07 00 05 04 00 00

	30 29 28 21 26 25 24 23	22 21 0	(ロット番号)
符号部	指数部 (8ビット)	仮数部 (23ビット)	

図 1 32 ビット単精度浮動小数点形式

- ① 符号部 (ビット番号 31)α の値が正のとき 0, 負のとき 1 が入る。
- 2 指数部 (ビット番号 30 ~ 23)βの値に 127 を加えた値が 2 進数で入る。
- ③ 仮数部 (ビット番号 22 ~ 0)

 $|\alpha|$ の整数部分 1 を省略し、残りの小数部分が、ビット番号 22 に小数第 1 位が来るような 2 准数で入る。

ただし、 α の値が0の場合、符号部、指数部、仮数部ともに0とする。

(2) 例えば、10 進数の0.75 を2 進数で表すと、 $(0.11)_2$ となる。これは $(1.1)_2 \times 2^{-1}$ と表記でき、単精度表現では、図2 のとおり、符号部は $(0)_2$ 、指数部は-1 に127 を加えて $(01111110)_2$ となり、仮数部は $(1.1)_2$ の小数部分が入るので、 $(100\cdots0)_2$ となる。ここで、 $00\cdots0$ は0 が連続していることを表す。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	•••	0
0	0	1	1	1	1	1	1	0	1	0	0	0	0	0		0

図 2 0.75 の単精度表現

設問]	ž	,,,,,																
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17		•••	()
0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	,	•••	()
解答群 ア 3 オ 1					2 3							$\times 2^5$ 1×2				3×2^{1} 11×2		
設問 2	ž	欠の	記述	中の			(こ入	れる	正し	۲۰۱۱	答え	を,	解答	群の中	から選	べ。	
二つの	浮動	小数 精度		: A &	≤ B (の減	算と	乗算	でそ	ゔ゚ゔ								
	Т		28												•	•••)
0	1	0	0	0	0	1	0	0	0	1	0	0	0	0		•••	()
		精度																
) 29	28	27	26	25	24	23	22	21	20	19	18	17			()
(1) 減算 ①	1	0 - B	0	0 欠の言	0	0	1 ③で	1	1	0	0	0	0	0	とから,	 Bを([()
(1) 減算 ① と ②	A - 指数する ((1. ②の	O - B 対部の で (01) ₂ の結果	0 を, 彡 値を 〒う。 「- (□ よを単	0 欠の ³ 大き a	0 手順(:い,大) 一) ま	0 ①~ ずに台 ₂)× 見する	1 ③で 含わっ 2 ⁵ =	1 行う せる。 = (1.0 その)	1 。。A	0 が(1 く 2 は	o	0 2×2 ⁵	0 でま	0	とから,	 Bを([()
(1) 減算 ① と ②	A - 指する が が が ((1.4) ((0 - B 対部の の。 にを行 の1) ₂ お果 なBの	0 を, ※ 値を 一, ○ ()	0 大き 大き 指度は(0 手順(いた) () () () () () () () () () (0 ①~ ① ② ②)× 記する	1 ③でかった。 2 ⁵ = 3。こ	1 分行う せる。 = (1.0 その;	1 。。A の)) ₂ 〉 結果 とな	0 が(1 く 2 は	0 .01)	0 2×2 ⁵	0 - - - - - - - - - - - - - - - - - - -	0		 Bを([a)
(1) 減算 ① と ② (2) 乗算 a に関す ア (A - 数 ((1.1.2 の) A × る角 (1.2.2 の) る角	0 - B: 対部の。 のの1を行 01) ₂ 対話 第 のの にを行	0 を, %を の (単果	0 大き 大き 指度は(0 手順(いた) (10 0.10 0.10 0.10 0.10 0.10 0.10 0.10	0 ①~ ① ② ②)× 記する	1 ③でかった。 2 ⁵ = 3。こ	1 分行う せる。 = (1.0 その;	1 。。A 3)) ₂ > 8 4 とな	0 が(1 くく2 は し る。	0 .01)	0 2×2 ⁵	0 - - - - - - - - - - - - - - - - - - -	0 うるこ 3。			a)
(1) 減算 ① と ② (2) 乗算 a に関す ア (A - お数る第((1.0のA× あり)。0111 る角	0 - B: かっ。 でもり。 を行りまま のの おま、 と を れる と を れる と に を れる と に を れる と に と に と に と に と に と に と に と に と に と	0 を値 うーを結 詳 詳 詳	0 欠の** 大き 精は(0 手順(:いた) :表現 (0.10	0 ① ① ② ② ② ② ② ② ② ② O O O O O O O O O O	1 ③でかった。 2 ⁵ = 2 3。 2 3。2	1 1 **行うがせる。 ==(1.4 **での。 **で、くっと、 **できる。 * * * * * * * * * * * * * * * * * * *	1 。。A の) ₂ > 結とな	0 が(1 くく2 は し る。	0 .01)	0 2 2 2 2 2 5	0 0 ここなる	0 うるこ 3。		7	a 1.1)
(1) 減算 ① ② ② ② ② 乗りす (と で り に関す こ に関す	A - 数る算((1.0のA× る角).0111 る角 132 る角 31	0 - B: xix o. xix o. x	0 を が が が が が が が が が が が が が が が が が が が	0 欠の ³	0 手順(:い方 0.10 4	0 ①~ ① (1)~ ② (2) × ② (2) × ② (2) × ② (2) × ② (2) × ② (2) × ③ (2) × ③ (2) × ③ (2) × ③ (2) × ③ (2) × ③ (2) × ⑥ (2) (2) (2) (2) (2) (2) (2) (2) (2) (2)	1 ③で かっ 2 ⁵ = 3。 こ]) ₂ >	1 (1.0) (イン・イン・イン・イン・イン・イン・イン・イン・イン・イン・イン・イン・イン・イ	1 。。A の)。2 結果 とな	0 が(1 はし る。	0 .01)	0 20	0 0 ですることなる	0 5るこ 3。 1.01 6	17	a	a 1.1	0 0
(1) 減算 ② ② (2) 乗算 aに関す で bに関す	A 指す減((2) A × あり).0111 る	0 - B: かっ。 でもり。 を行りまま のの おま、 と を れる と を れる と に を れる と に を れる と に と に と に と に と に と に と に と に と に と	0 を値 うーを結 詳 詳 詳	0 欠の** 大き 精は(0 手順(:いた)) (0.10 4	0 ① ① ② ② ② ② ② ② ② ② O O O O O O O O O O	1 ③でかった。 2 ⁵ = 2 3。 2 3。2	1 1 **行うがせる。 ==(1.4 **での。 **で、くっと、 **できる。 * * * * * * * * * * * * * * * * * * *	1 。。A の) ₂ > 結とな	0 が(1 くく2 はし る。	0 .01)	0 20	0 0 ですることなる	0 5るこ 3。 1.01 6		7	a 1.1	
(1) 減算 ① ② ② ② ② 乗算す (2) 乗算す (b に関す C に関す	A - 数る算((1.0のA× る角).0111 る角 132 る角 31	0 - B: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	0 を値うしき結 詳 詳 単 29 0	0 欠の ³	0 手順(いた) 元表現 (0.100 4	0 ① ① ② ② ② ② ② ② ② ② ② ② O O O	1 3でかった。 2 ⁵ = 25 0	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1 。。A 3)) ₂ 〉 結果とな	0 が(1 はし る。	0 b c c 21 0	0 20 0	0 0 エ エ 1 1 1 0	0 5るこ 3。 1.01 6	17	a	a 1.1	0 0

		30	29		27									18		•••	0
ウ	0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	•••	0
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17		0
I	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	•••	0
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	•••	0
才	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	•••	0
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17		0
カ	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	•••	0
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17		0
‡	0	1	0	0	0	0	1	0	1	1	0	0	0	0	0	•••	0
関す	「る角	2000年	Ι¥														
		+ 🗆 1			1 11			-	1	1101				1.11	1	-	1 1 1 1 1
ア	1.0			1	1.11			ウ	1.	1101			I	1.11	I	才	1.1111

間1 3 午後のカギ

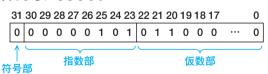
 $\alpha \times 2^{\beta}$ (α は仮数, β は指数)の形式で表した実数を、浮動小数点数といいます。浮動小数点数をコンピュータのビット列で表現する際の形式(IEEE 754 規格)に関する問題です。形式については、問題文で説明されているため、専門知識は必要ありません。

設問 1 2 進数の $(0.1)_2$ は、 $(1.0)_2 \times 2^{-1}$ と表せます。 ** (値) $_2$ は、2 進数の値を示します。

設問 2 2 進数の乗算は、計算の方法を工夫すると簡単になります。

設問 1 浮動小数点数の変換

単精度表現の符号部,指数部,仮数部は,それぞれ 次のようになります。



①符号部:0

0なので正の数です。

②指数部:00000101

 $(101)_2$ は 10 進数の 5 です。これは指数に 127 を加えた値なので,5-127=-122 が指数となります。

③仮数部:011000…0

小数部分が $(011000\cdots0)_2$ なので, 仮数部は $(1.011)_2$ です。

以上から、この浮動小数点数を $\alpha \times 2^{\beta}$ の形式で表す と、

$$(1.011)_2 \times 2^{-122} = (1011)_2 \times 2^{-125} = 11 \times 2^{-125}$$

となります。正解はオです。

仮数の小数点が1つ右に移動すると, 指数は1減るよ。



設問2 浮動小数点数の演算

空欄 ${\bf a}$: $(10000011)_2$ は 10 進数の 131 なので,浮動 小数点 ${\bf B}$ の指数部は 131-127=4,仮数部は $(1.1)_2$ です。

 $(1.1)_2 \times 2^4 = (0.11)_2 \times 2^5$

以上から、**空欄 a** は ウ です。

空欄 b: A - B を計算します。

$$((1.01)_2 - (0.11)_2) \times 2^5 = (0.1)_2 \times 2^5$$

= $(1.0)_2 \times 2^4$

以上から、**空欄 b** は **1** です。

空欄 \mathbf{c} : $(1.0)_2 \times 2^4$ を単精度表現で表します。指数部は、4+127=131。これを 2 進数で表すと、

$$\frac{100}{4} + \frac{10000000}{128} - 1 = \frac{10000011}{131}$$

となります。また、仮数部は $(1.0)_2$ の小数部分なので、 $(000\cdots)_2$ となります。

以上から、 $(1.0)_{2} \times 2^{4}$ の単精度表現は、

です。**空欄 c** は <mark>イ</mark> です。

空欄d:A×Bを計算します。計算が簡単になるように、 少し工夫しましょう。

$$\frac{(1.01)_2 \times 2^5}{A} \times \frac{(1.1)_2 \times 2^4}{B}$$
= $(1.01)_2 \times \frac{(1.0 + 0.1)_2}{\cancel{2}} \times 2^{(5+4)}$

- $= (1.01 \times 1.0 + 1.01 \times 0.1)_2 \times 2^9$
- $= (1.01 + 0.101)_2 \times 2^9$
- $= (1.111)_{2} \times 2^{9}$

以上から,**空欄 d** は エ です。

○ 解答

設問 1 才

設問2 a - ウ, b - イ, c - イ, d - エ

問2

コンパイラの最適化に関する次の記述を読んで、設問 1 \sim 3 に答えよ。

コンパイラとは、プログラム言語で記述された原始プログラムを翻訳して目的プログラムを生成するためのソフトウェアである。コンパイラの機能の一つに最適化がある。最適化では、原始プログラムを翻訳する過程で、プログラムの実行時間を短くするために原始プログラムの構造を変換する。最適化の方法の例を表1に示す。

表 1 最適化の方法の例

最適化の方法	内容
関数のインライン展開	関数を呼び出す箇所に,呼び出される関数のプログラムを展開する。
共通部分式の削除	同じ式が複数の箇所に存在し、それらの式で使用している変数の値が変更されず、式の値が変化しないとき、その式の値を作業用変数に格納する文を追加し、 複数の同じ式をその作業用変数で置き換える。
定数の畳込み	プログラム中の定数同士の計算式を、その計算結果で置き換える。
定数伝播	変数を定数で置き換える。
無用命令の削除	プログラムの実行結果に影響しない文を削除する。
ループ内不変式の移動	ループ中で値の変化しない式があるとき、その式をループの外に移動する。
ループのアンローリング	ループ中の繰返しの処理を展開する。

擬似言語の形式で記述したプログラムの一部(以下,プログラム1という)に対して,表1の 最適化の方法を複数組み合わせて最適化した例を,表2に示す。

〔プログラム 1〕

```
\blacksquare i:0, i ≤ 1, 1
  \cdot x[i] \leftarrow y[i] + m + n
  \cdot v[i] \leftarrow w[i] + m + n
```

表 2 プログラム 1 の最適化の例

最適化の方法	プログラム 1 の変換
1	■ i:0, i ≤ 1 , 1 • t ← m + n • x[i] ← y[i] + t • v[i] ← w[i] + t
2	• t ← m + n ■ i:0 , i ≤ 1, 1 • x[i] ← y[i] + t • v[i] ← w[i] + t
3	\cdot t ← m + n \cdot x[0] ← y[0] + t \cdot v[0] ← w[0] + t \cdot x[1] ← y[1] + t \cdot v[1] ← w[1] + t \cdot i ← 2

次の記述中の

表2において、最適化の方法を①、②、③の順で適用するとき、②で適用される最適化の方法 は a であり、③で適用される最適化の方法は b である。

a, bに関する解答群

ア 関数のインライン展開

ウ 定数の畳込み

オ 無用命令の削除

‡ ループのアンローリング

オ 共通部分式の削除

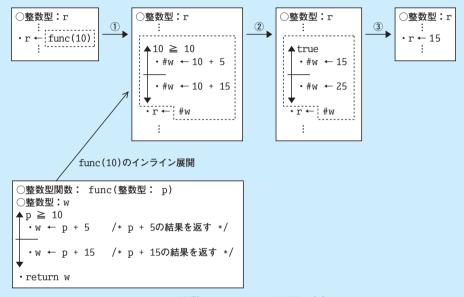
工 定数伝播

カ ループ内不変式の移動

設問 ク 次の記述中の に入れる正しい答えを、解答群の中から選べ。

関数のインライン展開の例を図1に示す。図1の関数 func をインライン展開した場合、実引 数 10 の値が仮引数 p に代入され, その値 10 が関数 func 内の変数 p の値になる (図 1 ①)。次に, c の方法を適用することによって、条件式 p ≥ 10 の判定結果や p を含んだ計算式の計算 結果がコンパイル時に確定する (\mathbf{Z} 1②)。その結果、 $\mathbf{func}(10)$ を値 15 に置き換えることがで きる (図13)。

なお、#w はコンパイラが最適化をしたときに生成した整数型の変数である。



関数のインライン展開の例 図 1

cに関する解答群

ア 共通部分式の削除

ウ 定数伝播

オ ループ内不変式の移動

イ 定数の畳込み

エ 無用命令の削除

カ ループのアンローリング

設問 ? 次の記述中の に入れる適切な答えを、解答群の中から選べ。

最適化をしないときと最適化をしたときとで浮動小数点数の演算の結果が異なる場合がある。 プログラム1に対して表2の最適化をしなかったものと,最適化をしたものとに,y[0]= 307000000.0, v[1] = 305000000.0, m = -303000000.0, n = 4.0 を与えて実行した。その結果の x[0], x[1]が次のとおりになった。ここで、同一優先順位の算術演算子は、左から順に演算する。

最適化をしないとき: x[0] = 4000004.0, x[1] = 2000004.0最適化をしたとき: x[0] = 4000000.0, x[1] = 2000000.0

この実行結果が異なる原因としては.「 d の方法の適用によって演算順序が変化したこと

で、 e が発生したからである。ここで、浮動小数点数の演算は単精度(仮数部は 23 ビット) で計算しているものとする。

- d に関する解答群
 - ア 関数のインライン展開
 - ウ 定数の骨込み
 - オ 無用命令の削除
 - **‡** ループのアンローリング
- オ 共通部分式の削除
- 定数伝播
- カループ内不変式の移動

e に関する解答群

- ア
 桁あふれ
 イ
 桁落ち
 ウ
 情報落ち

- エ 丸め誤差

コンパイラの最適化が、プログラムコードをどのように変換するのか、その具体的な方法について検 討する問題です。各設問の例が、表1の7種類の方法のどれに当てはまるのかを考えます。

設問 1 3つのプログラムの変更によって、実行時間がどのように短縮されるのかを理解しましょう。

設問2 図1の②の変換で変更された箇所に注目します。

設問3 計算順序を変えることで、浮動小数点数の有効桁数がどうなるかに注目します。

設問 1 プログラム 1 の最適化

表2の最適化の方法①~③を順番に検討します。 空欄 a には②、空欄 b には③の方法が入ります。

【最適化の方法①】

プログラム1では.

- $\cdot x[i] \leftarrow y[i] + m + n$
- $\cdot v[i] \leftarrow w[i] + m + n$

のように、式m+nが2箇所に共通しています。こ の部分は.

- \cdot t \leftarrow m + n
- $\cdot x[i] \leftarrow y[i] + t$
- $\cdot v[i] \leftarrow w[i] + t$

のように変更すれば、式m+nの計算が1回で済み ます。この操作は、表1の「共通部分式の削除」に 当たります。

【最適化の方法②】

空欄 a: 上記の式 t ← m + n は、ループの内側で実 行されていますが、tの値はループごとに変化するわ けではないので、ループの外側に出しても結果は変わ りません。

- **■** i: 0, i **≤** 1, 1
 - \cdot t \leftarrow m + n
 - $\cdot x[i] \leftarrow y[i] + t$
 - $\cdot v[i] \leftarrow w[i] + t$

 \cdot t \leftarrow m + n

■ i: 0, i **≤** 1, 1

 $\cdot x[i] \leftarrow y[i] + t$

 $\cdot v[i] \leftarrow w[i] + t$

繰返しごとに計算せずに済むので、このほうが効率 的です。この操作は,表1の「ループ内不変式の移動」

に当たります。**空欄 a** は <u>カ</u>です。

【最適化の方法③】

空欄 b:繰返し処理を逐次処理に展開すると、プログラムの行数は増えますが、繰返し開始行へのジャンプ、繰返し条件の評価、変数のインクリメントといった処理を繰返しごとに行う必要がなくなるので、処理速度は速くなります。この操作は、表1の「ループのアンローリング」に当たります。空欄 b は 幸です。

設問2 関数のインライン展開

空欄 c: 図1 ①では、インライン展開の結果、条件式 「 $p \ge 10$ 」が定数同士の計算「 $10 \ge 10$ 」に置き換わります。図1 ②では、この計算式を計算結果「true」に置き換えています。この操作は、表1 の「プログラム中の定数同士の計算式を、その計算結果で置き換える」という「定数の畳込み」に当たります。

定数同士の計算式を その計算結果で置き換える

以上から、**空欄 c** は **イ** です。

設問3 最適化による誤差の発生

最適化をしないプログラム 1 では、次のような計算が行われます。

y[0] m n $x[0] \leftarrow 307000000.0 + -303000000.0 + 4.0$

計算は左から順に行われるので.

のように進みます。いずれも, 有効桁数が同じ値同士 の計算です。

一方,最適化後のプログラム1では,最初に変数tにm+nの結果が格納されます。

-303000000.0 +4.0 $30299996.0 \rightarrow t$

この結果, y[0] + t の計算は,

のように,有効桁数が異なる値同士の計算になります。 浮動小数点数同士の計算では,指数を大きいほうにそろえるため, t の値は仮数部の下位部分が失われ.

 $307.0 \times 10^6 - 302.0 \times 10^6 = 5.0 \times 10^6$

のような結果になります。

このような誤差を情報落ちといいます。

情報落ちが発生したのは、最適化の方法①で m + n を先に計算するように変更したことが原因です。最適化の方法①は、先に検討したとおり「共通部分式の削除」に当たります。

以上から、空欄 d に入る字句は の「共通部分式 の削除」、空欄 e に入る字句は の「情報落ち」に なります。

解答欄にあるその他の誤差は、次のとおりです。

桁あふれ	演算結果が扱うことのできる数値の最大桁数 を超えてしまうことによる誤差(オーバフロー)。
桁落ち	絶対値がほぼ同じ減算で、有効桁数が減って しまうことによる誤差。
情報落ち	浮動小数点数同士の加減算で指数部を揃えた とき,指数部が小さい方の数値の仮数部の下 位部分が失われるために生じる誤差。
丸め誤差	四捨五入や切上げ,切捨てによる誤差。

誤差の種類は基礎知識として覚えて おこう。



○ 解答

設問 1 a - カ, b - 丰

設問 2 c - 1

設問3 d-1, e-ウ



社員食堂の利用記録データベースの設計と運用に関する次の記述を読んで、設 問1~4に答えよ。

A 社では、社員証の IC カード化に伴い、社員証を用いた社員食堂の精算システムを構築するこ とにした。トレーに載せた料理を精算機の前に置くと、料理皿に埋め込まれた IC チップのデータ から料金が計算され、合計金額が表示される。合計金額を確認した後に社員証をかざすと、精算 ができる。精算データはデータベースに記録され、1か月分の精算額が、まとめて翌月の給料か ら引き落とされる。

A 社のシステム部門では、精算データを記録するデータベースとして、当初、図1に示す表を 設計した。

精算表

社員番号	日付	精算額
050221	20120310	380

図1 表構成とデータの格納例

精算システムの機能に関して関係者にヒアリングした結果、給料からの引落し額の算出以外に も次に示す四つの要望が挙がった。

要望1:ある社員の、ある日の精算の明細を表示できること

要望2:ある日の売上合計額(精算額の合計)を算出できること

要望3:料理の一覧を表示できること

要望4:ある日の、ある料理の販売皿数を算出できること

そこで、四つの要望に対応できるように図2のとおり、1回の精算に対して一つの精算コード を割り当てた三つの表で構成するように設計を変更した。下線付きの項目は主キーを表す。

精算表

精算コード	社員番号	日付	精算額
03100186	050221	20120310	380

明細表

精算コード	料理コード	皿数
03100186	0001	1
03100186	0002	1

料理表

料理コード	料理名	単価
0001	ごはん	100
0002	肉じゃが	280

図2 変更後の表構成とデータの格納例

設問 図 1 に示した表構成のままでも対応できる要望として正しい答えを、解答群の中から選べ。
解答群 ア 要望 1 イ 要望 2 ウ 要望 3 エ 要望 4
設問 2 料理名が "肉じゃが" の単価に誤りがあることが判明したので,購入者に差額を返金することになった。"肉じゃが" 購入者の社員番号と購入皿数を求める。
次のSQL文の に入れる正しい答えを、解答群の中から選べ。
SELECT 精算表.社員番号, SUM(明細表.皿数) AS 購入皿数 FROM 料理表, 精算表, 明細表 WHERE
·····································
7 精算表. 精算コード = 明細表. 精算コード AND 明細表. 料理コード = (SELECT 料理表. 料理コード FROM 料理表 WHERE 料理表. 料理名 = '肉じゃが')
✓ 精算表.精算コード = 明細表.精算コード AND 明細表.料理コード = 料理表.料理コード GROUP BY 精算表.社員番号 HAVING 料理表.料理名 = '肉じゃが'
ウ 精算表. 精算コード = 明細表. 精算コード AND 明細表. 料理コード = 料理表. 料理コード AND 料理表. 料理名 = '肉じゃが'
工 精算表.精算コード = 明細表.精算コード AND 明細表.料理コード = 料理表.料理コード AND 料理表.料理名 = '肉じゃが' GROUP BY 精算表.社員番号
設問 3 次の記述中の に入れる適切な答えを、解答群の中から選べ。
A 社の健康管理部門から、精算時に料理の合計カロリーを表示する機能と、ある社員の、ある

しかし、平均カロリーを求めるには a を結合しなければならないので、多くの検索と計算の処理が必要となることが予想できた。精算時に合計カロリーを計算するので、その情報を記録しておけば、検索及び計算量は少なくなる。そこで、 b に、精算単位の合計カロリーの列を追加することにした。これによって、前述の平均カロリーを求める場合は、 b だけを参照すればよいので、処理の高速化が期待できる。

期間における1回の精算当たりの平均カロリーを求める機能の追加を要望された。このため、料

理表にカロリーの列を追加することにした。

解答群

- ア精算表
- ウ 精算表と料理表
- 才 明細表
- エ 料理表と明細表
- 1 精算表と明細表
- 精算表と明細表と料理表
- カ料理表

設問 4 設問 3 のカロリーに関する機能を追加した後、食堂利用者にカロリーを意識して料理を選んでもらうために、人気料理とそのカロリーを掲示することにした。販売皿数の多い順に、料理名、カロリー及び販売皿数を求める。正しい SQL 文を、解答群の中から選べ。

解答群

- SELECT 料理表.料理名,料理表.カロリー,COUNT(明細表.皿数) AS 販売皿数 FROM 料理表,明細表 WHERE 料理表.料理コード = 明細表.料理コード GROUP BY 料理表.料理名,料理表.カロリー ORDER BY 販売皿数 DESC
- SELECT 料理表.料理名,料理表.カロリー,COUNT(明細表.皿数) AS 販売皿数 FROM 料理表,明細表 WHERE 料理表.料理コード IN (SELECT 明細表.料理コード FROM 明細表 WHERE 明細表.皿数 IS NOT NULL) GROUP BY 料理表.料理名,料理表.カロリー ORDER BY 販売皿数 DESC
- SELECT 料理表.料理名,料理表.カロリー,SUM(明細表.皿数)AS 販売皿数 FROM 料理表,明細表 WHERE 料理表.料理コード = 明細表.料理コード GROUP BY 料理表.料理名,料理表.カロリー ORDER BY 販売皿数 DESC
- SELECT 料理表.料理名, 料理表.カロリー, SUM(明細表.皿数) AS 販売皿数 FROM 料理表, 明細表 WHERE 料理表.料理コード IN (SELECT 明細表.料理コード FROM 明細表 WHERE 明細表.皿数 IS NOT NULL) GROUP BY 料理表.料理名, 料理表.カロリー ORDER BY 販売皿数 DESC

問3 3 午後のカギ

データベースの設計と、基本的な SQL を使った問合せの問題です。複数の表を結合するときの結合 条件の指定方法や、GROUP BY 句の使い方を理解しておきましょう。

設問 1 精算表は料理についての情報を持っていません。

設問 2 3 つの表を結合するには、結合条件が 2 つ必要です。

設問3 計算に必要な項目がどの表にあるかを考えます。

設問 4 1回に2皿以上注文する社員もいるので、皿数の集計には SUM 関数を使います。

設問 1 精算表だけで対応できる要望

図1の精算表の項目は、社員番号、日付、精算額の3つだけで、料理についての情報がありません。したがって、料理に関する情報を使う要望には対応できません。

- ×要望1 明細には、社員がどの料理を何皿購入した かを示す情報が必要です。
- ○要望 2 正解です。ある日の精算額の合計は、日付 と精算額がわかれば算出できます。
- ×要望3 料理の一覧は精算表では表示できません。
- ×要望4 料理ごとの皿数の情報が必要です。

以上から, 正解は 1 です。

設問2 肉じゃがの購入皿数を集計

まず、**図2**の3つの表を結合し、その中から "肉じゃが" を購入した行だけを抜き出します。3つの表を結合するので、2つの結合条件が必要です。

精算表. 精算コード = 明細表. 精算コード AND 明細表. 料理コード = 料理表. 料理コード



結合した表から "肉じゃが" 購入者だけを抜き出します。抽出条件は次のようになります。

料理名 = '肉じゃが'

抽出した行を、社員番号ごとに集計すれば、肉じゃが購入者の皿数を求められます。社員番号ごとにグループ化するので、GROUP BY句を次のように指定します。

GROUP BY 精算表, 社員番号

社員番号ごとにグループ化

江貝田りこ	212	ループル			
社員番号	皿数	料理名		社員番号	購入皿数
XXXXXX	1	肉じゃが	SUM (皿数)	XXXXXX	3
XXXXXX	2	肉じゃが		YYYYYY	1
YYYYYY	1	肉じゃが	7	ZZZZZZ	3
ZZZZZZ	1	肉じゃが			
ZZZZZZ	1	肉じゃが	}		
ZZZZZZ	1	肉じゃが			
		┺	理名が"肉じゃ	が"の行だけ	を取り出す

以上をまとめると、SQL 文は次のようになります。

SELECT 精算表. 社員番号 , SUM(明細表. 皿数) AS 購入皿数

FROM 料理表, 精算表, 明細表

WHERE 精算表.精算コード = 明細表.精算コード AND 明細表.料理コード = 料理表.料理コード AND 料理名 = '肉じゃが'

GROUP BY 精算表, 社員番号

- ×ア 社員ごとに集計されていません。
- ※ 「HAVING 料理表.料理名 = '肉じゃが'」 句は、"肉じゃが"を購入したことのある社員 だけをグループ化する指定です。集計結果は肉 じゃがを含む全料理の購入皿数も含まれます。
- × ウ 社員ごとに集計されていません。
- I 正解です。

設問3 カロリーの表示

空欄 a:カロリーは料理表に追加するので、精算ごとのカロリー合計を求めるには、「料理表」と「明細表」を結合する必要があります。これを社員ごとに集計して平均を求めるので、「精算表」も結合します。したがって、空欄 a は です。

空欄 b: 精算単位の合計カロリーなので、1 件の精算につき 1 行のデータがある「精算表」に追加するのが適切です。空欄 b は ア です。

明細表は同じ精算コードの行が複数あるので、精算単位のデータを追加する のには向かないよ。



設問4 販売皿数の多い順に集計

「明細表」の販売皿数を料理ごとに集計し、大きい順に料理名とカロリーを表示します。SQL文は次のようになります。

SELECT 料理表.料理名,料理表.カロリー, SUM(明細表.皿数) AS 販売皿数 …①

FROM 料理表 ,明細表

- WHERE 料理表.料理コード = 明細表.料理コード …2
- GROUP BY 料理表.料理名,料理表.カロリー …3
- ORDER BY 販売皿数 DESC

...(4)

- ①料理名,カロリー,販売皿数を表示
- ②料理表と明細表を結合
- ③料理名ごとにグループ化
- ④販売皿数の大きい順(DESC)に並べ替え

以上から、正解は ウ です。

アとイは、集計関数が COUNT (明細表. 皿数) になっているため、販売皿数ではなく精算数が集計されます。

また、イ、エは、結合条件が間違っています。



84

データ転送時のフロー制御に関する次の記述を読んで、設問に答えよ。

端末 A から端末 B にデータを転送する。端末 A では、データを 1k バイト単位に分割し、分割したデータをそれぞれ一つずつのパケットに格納して送信する。パケット一つの大きさは 1.2k バイトである。端末 B では、受信したパケットを受信バッファに格納し、受信処理として、パケットの整合性の検査とパケットからのデータ抽出を行う。その後、パケットを正しく受信したことを知らせるために、端末 A に ACK を送信する。受信処理後は直ちに受信バッファの再利用が可能となる。端末 B の受信バッファの大きさは 1.2k バイトである。端末 A では、端末 B の受信バッファの大きさは判明しているが、端末 B での各処理に掛かる時間は分からない。

端末 A は,送信したパケットに対応する端末 B からの ACK を受信することで,端末 B の受信バッファに空きができたことを検知し、次のパケットを送信する。

端末 A で、パケット 1 個の送信に掛かる時間は $10 \le 1$ り秒,ACK 受信に掛かる時間は $0.5 \le 1$ 秒,ACK の受信を完了してから次のパケットが送信可能になるまでに掛かる時間は $0.5 \le 1$ 秒である。端末 A と端末 B との間の通信の様子と、端末 A での各処理に掛かる時間を、**図 1** に示す。

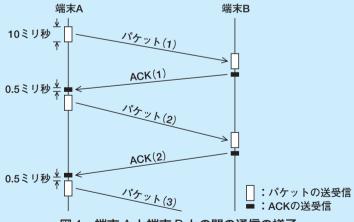


図 1 端末 A と端末 B との間の通信の様子

パケット 1 個の送信を完了してから、対応する ACK の受信を開始するまでに掛かる時間は一定で、その時間が 100 ミリ秒であったとすると、データ 1M バイトを送信し、次のパケットが送信可能となるまでの時間(以下、データ 1M バイト当たりの転送時間という)は a 秒である。ここで、1M バイトは 1.000k バイトとし、パケットや ACK は確実に相手に届くものとする。

データの転送に掛かる時間を短縮するために、端末Bの受信バッファを2.4kバイトに拡大した。 端末Aは、端末Bの受信バッファに空きがあることが確かなときは、送信したパケットに対応 する ACK の受信を待たずに次のパケットを送信することができる。すなわち、"送信済みのパケッ ト数 - 受信済みの ACK 数"が1以下であれば、端末 A は次のパケットを送信できる。パケット 1個の送信を完了してから次のパケットが送信可能になるまでに掛かる時間は0.5ミリ秒であり、 その他に掛かる時間は受信バッファの大きさが 1.2k バイトのときと同じである。

パケットの送信と ACK の受信。及びパケットの受信と ACK の送信は、並行して行うことがで きる。このときの通信の様子を、図2に示す。

このとき、データ 1M バイト当たりの転送時間は b 一秒である。

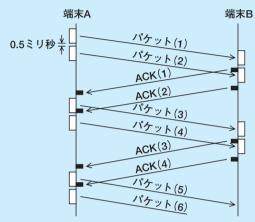


図2 受信バッファを拡大したときの通信の様子

受信バッファを更に拡大することで、データの転送に掛かる時間を短縮することを考える。受 信バッファの大きさを (1.2×n) k バイトとすると, "送信済みのパケット数 - 受信済みの ACK 数" が c ときは、端末 A は端末 B の受信バッファにパケット 1 個分以上の空きがあることが分 かるので、次のパケットを送信することができる。

各処理に掛かる時間が図1及び図2のとおりであり、パケット1個の送信を完了してから、対 応する ACK の受信を開始するまでに掛かる時間は一定で、その時間が 100 ミリ秒であったとする と、データ 1M バイト当たりの転送時間は d 秒まで短くすることができ、このときの最小 の受信バッファの大きさは e kバイトである。

設問 本文中の に入れる正しい答えを、解答群の中から選べ。

a, b, d に関する解答群

7 8.3

50

1 8.8 **=** 55.5 **ウ** 10

92.5

10.5

100

46.3

111

cに関する解答群

ア n 以下の

1 n 以上の

ウ n 未満の

I nより大きい

e に関する解答群

7 9

10

D 10.8

I 11

12

カ 13.2

= 14.4

間4 乳午後のカキ

ネットワークを介したデータの送受信で、データ送信が高速すぎて受信側が処理しきれないことのないように、通信速度を制御することをフロー制御といいます。この問題では、送信側の送ったデータに対し、受信側が ACK 信号を返すという単純なフロー制御が行われています。

また、インターネット標準の TCP などで、ACK を待たずに連続して送信できるパケットのまとまりをウィンドウといいます。ウィンドウのサイズは、受信側の受信バッファの大きさによって決まります。 問題文のプロトコルは TCP などの実際のプロトコルとは異なるため、こうした基礎知識がなくても解答は可能ですが、実際のプロトコルをイメージできると問題の意味も理解しやすくなります。

空欄 a: データは 1 パケット当たり 1k バイトなので、 1M (= 1,000k) バイト送信するには、1,000 パケット送信する必要があります。

問題文より、端末 A が 1 個のパケットの送信を開始してから、次のパケットを送信するまでに掛かる時間は、次の時間の合計です。

- ①送信に掛かる時間:10 ミリ秒
- ② ACK の受信開始までの時間:100 ミリ秒
- ③ ACK 受信に掛かる時間: 0.5 ミリ秒
- ④次のパケットが送信可能になるまでの時間: 0.5 ミリ秒

① \sim ④の合計は、10 + 100 + 0.5 + 0.5 = 111 ミリ秒になります。パケットは 1.000 個あるので、

111×1,000 = 111,000 ミリ秒= 111 秒

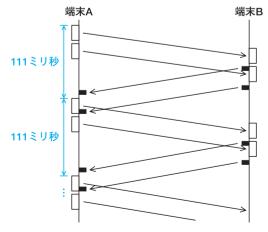
となります。空欄 a は です。

空欄 b:端末 B の受信バッファが 2 倍になったので、2 個のパケットを連続して送信できます。最初の 2 個のパケットを送信してから、次のパケットが送信可能になるまでのタイムテーブルは、次のようになります。

経過時間	動作	
0	①パケット(1)送信開始	
10	②パケット(1)送信完了	
10.5	③パケット(2)送信開始	
20.5	④パケット(2)送信完了	

110	⑤ ACK (1) 受信開始(②の 100 ミリ
	秒後)
110.5	⑥ ACK (1) 受信完了
111	⑦パケット (3) 送信開始
120.5	⑧ ACK (2) 受信開始(④の 100 ミリ
	秒後)
121	⑨パケット (3) 送信完了
	① ACK (2) 受信完了

上記の表から、2個のパケットを連続して送信してから、次のパケット(3)が送信可能になるまでに掛かる時間は、111ミリ秒であることがわかります。



以上から、1,000個分のパケットを送信するのに掛

かる時間は.

111× (1,000÷2) = 55,500 ミリ秒= 55.5 秒

となります。空欄 b は ‡ です。

空欄 c: 送信済みのパケット数は受信バッファに入ったパケットの数, 受信済みの ACK 数は, 受信バッファから出たパケットの数を表します。したがって「送信済みのパケット数ー受信済みの ACK 数」は, 現在受信バッファに残っているパケットの個数を表します。

送信済みの	受信バッファ	受信済みの
パケット数、		ACK数

受信バッファの大きさがパケット n 個分とすると、 受信バッファにパケット 1 個分以上の空きがあるの は、受信バッファに残っているパケットが n 個より少 ないとき、すなわち「送信済みのパケット数ー受信済 みの ACK 数 | が n 未満のときです。

以上から、空欄 c は ウ です。

空欄 d:端末 B の受信バッファが十分に大きければ、すべてのパケットを連続して送信できます。転送時間はこのとき最短になります。

1M バイトのデータ送信に必要なパケットは 1,000 個, パケット 1 個の送信に掛かる時間は, 間の時間を含めて 10.5 ミリ秒なので, 転送時間は次のようになります。

10.5ミリ秒×1,000個=10,500ミリ秒=10.5秒

以上から、正解は エ です。

空欄 e: 受信バッファに残っているパケット数 (送信済みのパケット数 – 受信済みの ACK 数) が n 未満である間は、連続してパケットを送信できます。

最初のパケットの送信開始から、最初の ACK の受信開始まで、110ミリ秒掛かります。その間に送信できるパケットは、次のようになります。

経過 時間	動作	受信バッファ のパケット数
0	①パケット(1)送信	1
10.5	②パケット(2)送信	2
21	③パケット(3)送信	3
31.5	④パケット(4)送信	4

42	⑤パケット(5)送信	5
52.5	⑥パケット(6)送信	6
63	⑦パケット(7)送信	7
73.5	⑧パケット(8)送信	8
84	⑨パケット(9)送信	9
94.5	⑩パケット(10)送信	10
105	⑪パケット(11)送信	11
110	⑩ ACK(1)受信	10

②で ACK (1) を受信すると、受信バッファに 1 つ空きができるので、次のパケット (12) を送信できます。以降は、次のように連続してパケットを送信できます。

経過 時間	動作	受信バッファ のパケット数
115.5	⑬パケット(12)送信	11
120.5	¹ ACK(2)受信	10
126	⑮パケット(13)送信	11
131	¹⁶ ACK(3)受信	10
:	:	:

以上から、端末 B の受信バッファがパケット 11 個分あれば、連続してパケットを送信できます。パケット 1 個の大きさは 1.2k バイトなので、受信バッファの大きさは次のようになります。

1.2×11 = 13.2k バイト

以上から、**空欄e**は<u>カ</u>です。

○ 解答 ○
 設問 a - □, b - ‡, c - □,
 d - エ, e - ヵ



受験者数の集計リスト作成に関する次の記述を読んで、設問1~3に答えよ。

C大学では、今年度と前年度の受験者データを用いて、出身校ごとの受験者数を集計した出身 校別受験者数リスト(以下、出身校リストという)を作成する。

〔受験者データ及び出身校リストの説明〕

(1) 今年度と前年度の受験者データは、それぞれ今年度受験者ファイルと前年度受験者ファイル の二つの順ファイルに保存されている。これらのレコード様式を**図1**に示す。

受験番号 姓 名 学校コード 高校卒業年度

図 1 今年度受験者ファイル及び前年度受験者ファイルのレコード様式

- (2) 今年度受験者ファイル及び前年度受験者ファイルのレコードは、どちらも学校コードの昇順に並んでいる。
- (3) 図2に示す例のように、出身校リストの印字項目は、順位、学校名、今年度受験者数、前年度受験者数である。印字の順序は、今年度受験者数の降順とし、今年度受験者数が同数の場合は、前年度受験者数の降順、それも同数の場合は、学校コードの昇順とする。順位は、今年度受験者数の多い方から1、2、3、…と採番した番号である。今年度受験者数が同じ場合は同順位とし、次の順位は、同順位の学校数分だけ加算した順位とする。学校名は、学校名ファイルから得る。学校名ファイルのレコード様式を図3に示す。

出身校別受験者数リスト				
順位 学校名		今年度受験者数	前年度受験者数	
1	東南高校	82	62	
2	東西高校	80	96	
2	西北高校	80	68	
4	南西高校	73	64	
5	北北学園	60	70	
5	南南学園	60	0	
	その他計	2666	2723	
	合計	3101	3083	

図2 出身校リストの例

学校コード 学校名

図3 学校名ファイルのレコード様式

(4) **図2**に示す例のように、出身校リストには、1位から指定した順位(以下、指定順位という)までを印字し、指定順位よりも下位の学校の受験者数は、それらを合計して"その他計"として印字する。

〔出身校リスト作成処理の説明〕

図4に、出身校リスト作成処理の流れと各ファイルのレコード様式を示す。

- (1) 集計処理では、今年度受験者ファイルと前年度受験者ファイルから、それぞれ学校ごとに受験者数を集計したファイル1及びファイル2を作成する。
- (2) 突合せ処理では、学校コードをキーとして、ファイル1とファイル2の突合せを行い、ファイル3を作成する。この突合せ処理の中では、整列は行わない。
- (3) 整列処理では、ファイル3を整列し、ファイル4を作成する。
- (4) 順位付け処理では、ファイル4の各レコードの順位付けを行い、ファイル5を作成する。
- (5) リスト作成処理では、ファイル5から、出身校リストを作成する。学校ごとの明細行の印字は、ファイル5の1レコードの内容から1行ずつ、指定順位まで印字する。このとき、学校名は、学校コードをキーにして学校名ファイルから読み込む。指定順位よりも下位のレコードは、それらの受験者数を合計して"その他計"として印字する。

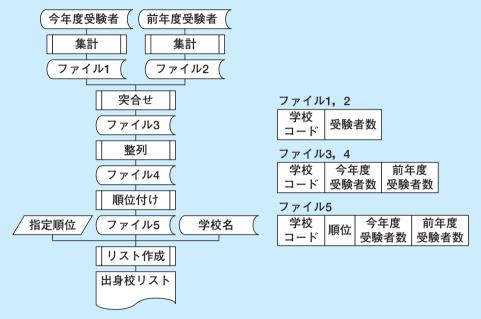


図 4 出身校リスト作成処理の流れと各ファイルのレコード様式

表1は、図4中の突合せ処理における条件に合致するレコードの有無と出力レコードの各項目の内容である。表1中の に入れる正しい答えを、解答群の中から選べ。

表 1 突合せ処理での出力レコードの各項目の内容

レコードの有無		学校コード	今年度受験者数	前年度爲驗妻数	
ファイル1	ファイル 2	子仅一 1	7千及文級有数	削十及文数有数	
あり	あり				
あり	なし			a	
なし	あり		b		

注記 網掛けの部分は表示していない。

a. bに関する解答群

ア	0	0
1	0	ファイル1の当該項目
ゥ	0	ファイル2の当該項目
I	ファイル1の当該項目	0
才	ファイル1の当該項目	ファイル2の当該項目
力	ファイル2の当該項目	0
+	ファイル2の当該項目	ファイル 1 の当該項目

設問 2 図4中の整列処理に最低限必要な整列キー項目及び整列順(昇順又は降順)の 組の並びとして正しい答えを、解答群の中から選べ。ここで、キー項目及び整 列順の組は、(「整列キー項目」:「整列順」)で表す。また、コンマで区切られた組の並びは、 左の方が整列の優先度が高いことを表す。

解答群

- ア (学校コード;昇順), (今年度受験者数;昇順), (前年度受験者数;昇順)
- ✓ (学校コード;昇順),(前年度受験者数;降順),(今年度受験者数;降順)
- 뉯 (今年度受験者数;降順),(前年度受験者数;降順),(学校コード;降順)
- □ (今年度受験者数;降順),(前年度受験者数;降順),(学校コード;昇順)
- 才 (今年度受験者数;降順),(前年度受験者数;降順)
- カ (今年度受験者数;昇順),(学校コード;昇順)

設問 3 次の記述中の に入れる正しい答えを、解答群の中から選べ。

〔出身校リストへの印字項目追加の説明〕

図5に示す例のような、出身校リストに前年度順位を追加した出身校別受験者数リスト2(以下、出身校リスト2という)を作成することになり、処理の追加と変更を行うことになった。前年度に受験者がいなかった高校の前年度順位は空欄とする。

山白松叫巫野土粉リフリの

図6に出身校リスト2作成処理の流れと主なファイルのレコード様式を示す。

四身仪別党験省致リスト2							
順位	学校名	今年度受験者数	前年度順位	前年度受験者数			
1	東南高校	82	8	62			
2	東西高校	80	1	96			
2	西北高校	80	4	68			
4	南西高校	73	5	64			
5	北北学園	60	3	70			
5	南南学園	60		0			
	その他計	2666		2723			
合計		3101		3083			

図5 出身校リスト2の例

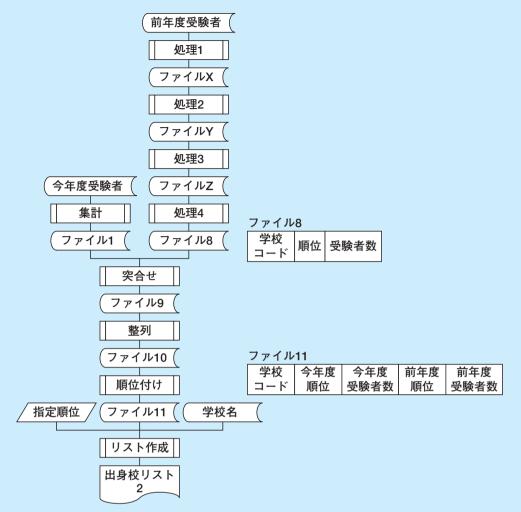


図 6 出身校リスト2作成処理の流れと主なファイルのレコード様式

(出身校リスト2作成処理の説明)

- (1) 図6の処理は、図4の同名の処理に、入出力のレコード様式への前年度順位の追加に対する変更を加えたものを用いる。
- (3) 変更になる突合せ処理の出力であるファイル 9 及びそれを整列した出力であるファイル 10 の レコード項目は、 e である。

c, dに関する解答群

- ア 学校コードの降順への整列
- イ 学校コードの昇順への整列
- ウ 学校ごとの人数の集計
- エ 受験者数の降順への整列
- オ 受験者数の昇順への整列
- カ順位付け

+ リスト作成

e に関する解答群

- ア 学校コード,今年度受験者数,前年度受験者数
- ✓ 学校コード, 今年度受験者数, 前年度順位
- ウ 学校コード, 今年度受験者数, 前年度順位, 前年度受験者数
- 学校コード, 今年度順位, 今年度受験者数, 前年度受験者数
- 才 学校コード、今年度順位、今年度受験者数、前年度順位、前年度受験者数
- カ 学校コード、今年度順位、前年度受験者数
- 学 学校コード、今年度順位、前年度順位
- ク 学校コード, 今年度順位, 前年度順位, 前年度受験者数

問5 等午後のカギ

既存のデータから、目的の出力を得るためのソフトウェア設計の問題です。特に、突合せ処理は午後 問題でひんぱんに出題されるので理解しておきましょう。

設問 1 簡単な例を思い浮かべて考えると、わかりやすくなります。

設問 2 整列項目については問題文に説明があります。

設問3 突合せ処理に入力するファイルは、あらかじめ共通のキーで整列しておく必要があります。

設問 1 突合せ処理

空欄 a, b: 突合せ処理は、共通のキー項目のある 2 つのファイルを集計して、新しいファイルを作る処理です。ここでは、ファイル 1 (今年度の出身校別受験者数) とファイル 2 (前年度の出身校別受験者数) から、新しいファイル 3 を作成します。

次のような簡単な例で考えてみましょう。

ファイル1

ノアイルロ						
学校コード	受験者数					
0001	11					
0002	8					
:	:					

ファイル2

学校コード	受験者数	
0001	13	
0003	7	
:	:	



		•	
	学校コード	今年度 受験者数	前年度 受験者数
1	0001	11	13
2	0002	8	0
3	0003	0	7

- ①ファイル 1, ファイル 2 ともにレコードがある学校 (0001) は, 今年度, 前年度ともに受験者がいます。ファイル 3 のレコードには, 今年度受験者数にファイル 1 の当該項目, 前年度受験者数にファイル 2 の当該項目が入ります (学校コードはどちらでもかまいません)。
- ②ファイル1にレコードがあって、ファイル2にレコードがない学校(0002)は、今年度は受験者がいて、前年度は受験者がいないケースです。したがってファイル3のレコードは、学校コードと今年度受験者数にファイル1の当該項目を入れ、前年度受験者数を0にします。
- ③ファイル2にレコードがあって,ファイル1にレコードがない学校(0003)は,今年度は受験者がおらず,前年度は受験者がいたケースです。したがってファイル3のレコードは,学校コードと前年度受験者数にファイル2の当該項目を入れ,今年度受験者数を0にします。

以上を表にまとめると、次のようになります。

レコート ファイル 1	べの有無 ファイル 2	学校 コード	今年度 受験者数	前年度 受験者数
あり	あり	どちらでも よい	ファイル 1 の当該項目	ファイル 2 の当該項目
あり	なし	ファイル 1 の当該項目	ファイル 1 の当該項目	0
なし	あり	ファイル 2 の当該項目	0	ファイル 2 の当該項目

以上から,**空欄 a** には <mark>エ,空欄 b</mark> には <mark>カ</mark> が入り ます。

設問2 整列処理

問題文によれば、出身校リストの印字の順序は次のようになっています。

印字の順序は、①今年度受験者数の降順とし、今年度受験者数が同数の場合は、②前年度受験者数の降順、それも同数の場合は、③学校コードの昇順とする。

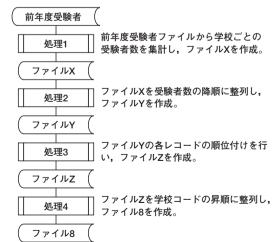
以上から、整列キー項目は、

(今年度受験者数;降順),(前年度受験者数;降順),(学校コード;昇順)

となります。正解はエです。

設問3 印字項目の追加

空欄 c, d: ファイル 8 に順位の項目が追加されていることから,前年度受験者数の順位は,突合せ処理の前の段階で順位付けを行うことがわかります。処理 1 から処理 4 までの流れは,次のようになります。



処理 4 で、ファイル Z を学校コードの昇順に整列するのは、今年度受験者数と突合せ処理を行うために、ファイルを同じキーで整列しておく必要があるからです。

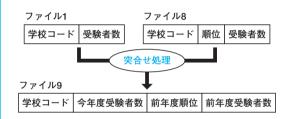
突合せ処理で処理するファイルは、同 じキー項目で整列しておくのが鉄則。



空欄 c は処理 2 の内容なので、 **工**の「**受験者数の 降順への整列**」が入ります。

また,**空欄** \mathbf{d} は処理4の内容なので, \mathbf{d} の「**学校コードの昇順への整列**」が入ります。

空欄 e: ファイル1のレコード項目は、学校コード、受験者数の2つ。ファイル8のレコード項目は、学校コード、順位、受験者数の3つです。これらを突合せした結果は、次のようになります。



以上から、ファイル 9、10 のレコード項目は、学校 コード、今年度受験者数、前年度順位、前年度受験者 数の 4 つになります。正解は っです。







設計工程での進捗管理に関する次の記述を読んで、設問1~3に答えよ。

チーム X は、あるシステム開発プロジェクトにおけるユーザインタフェース設計(以下, UI 設 計という)を担当している。このプロジェクトでは、週単位でプロジェクトの進捗状況を把握し、 計画値と実績値の比較分析によって、スケジュール遅延などに早期対応を行っている。

(設計工程での進捗管理の説明)

- (1) 就業規則は、次のとおりである。
 - ① 1日の勤務時間は8時間である。
 - ② 週のうちで勤務が認められている日は、月曜日から金曜日までの5日間である。土曜日 と日曜日は休日であり、勤務は禁止されている。
- (2) 設計工程の開始前に、各メンバの UI 設計に要する工数を見積もり、設計工程の各週に分配す る。分配した工数を計画工数といい、単位は時間である。
- (3) 週単位で進捗状況を把握するための指標の一つとして、計画進捗率を用いる。計画進捗率は、 次の計算式で求め、小数第2位を四捨五入する。

"計画工数の累積"とは、計画工数を設計の開始時点からその週まで累積した値である。

〔チームの状況〕

(1) チーム X のメンバは A, B, C, D の 4 名であり、メンバ別の設計分担、計画工数及び計画進 排率は**表** 1 のとおりである。

表 1 設計工程におけるメンバ別の設計分担、計画工数及び計画	進捗率
--------------------------------	-----

		第1週		第2週		第3週		計画工数	
設計分担	メンバ	計画 工数 (時間)	計画 進捗率 (%)	計画 工数 (時間)	計画 進捗率 (%)	計画 工数 (時間)	計画 進捗率 (%)	前岡工数 の総合計 (時間)	
UI 設計 1	Α	0	0.0	25	50.0	25	100.0	50	
UI 設計 2	В	20	20.0	40	60.0	40	100.0	100	
UI 設計 3	С	20	20.0	40	60.0	40	100.0	100	
UI 設計 4	D	20	20.0	40	60.0	40	100.0	100	
合訂	†	60	17.1	145	58.6	145	100.0	350	

- (2) 設計工程の期間は3週間であり、現在、第2週が終了したところである。
- (3) 設計工程での第1週及び第2週における,各メンバが実際に作業した時間数(以下,実績工 数という)は、表2のとおりである。第1週では、他業務の都合で実績工数が計画工数と異 なるメンバがいたが、第2週は全メンバが計画工数どおりの工数を UI 設計に充てた。

表 2 各メンバの実績工数及び残り工数

単位 時間

設計分担 メンバ	42.45	計画工数の	第1週の週末時点		第2週の週末時点	
	総合計	実績工数の累積	残り工数	実績工数の累積	残り工数	
UI 設計 1	Α	50	0	50	25	20
UI 設計 2	В	100	25	75	65	35
UI 設計 3	С	100	20	80	60	45
UI 設計 4	D	100	15	85	55	40
合計		350	60	290	205	140

- ① "計画工数の総合計"は、表1の右端の欄の値に等しい。
- ② "実績工数の累積"は、各週末時点で、各週の実績工数をその週まで累積した値である。
- ③ 残り工数は、各週末時点で、各メンバの今後の作業に要する工数を見積もりし直したもので ある。

設問 1 次の計算式による指標を実績進捗率といい、各週末時点で算出する。ここで、 実績進捗率は、小数第2位を四捨五入した値である。計画進捗率と実績進捗率 との対比によるチーム X の進捗判定に関する次の記述中の に入れる適切な答え を. 解答群の中から選べ。

実績進捗率 = 実績工数の累積 実績工数の累積 + 残り工数 × 100

実績進捗率と計画進捗率が等しい場合は、計画どおりの進捗であり、実績進捗率が上回ってい る場合は、計画よりも進んでいる状況である。いずれの場合も、設計工程の進捗に問題がないと 判定する。一方、実績進捗率が計画進捗率を下回っている場合は、スケジュール遅延などの問題 が懸念される。

第1週の週末時点では、UI 設計作業に着手しているメンバの中で、実績進捗率が計画進捗率を 下回っているメンバは a 。第2週の週末時点では, b の2名であり,メンバ個別に 判定した場合には、やや進捗遅れの懸念がある。次に、表2の最下行にある合計値によるチーム X全体の実績進捗率を用いて、 c と判定した。

a に関する解答群

ア B である イ C である ゥ D である

エ いない

b に関する解答群

ア AとB ✓ A と C ウ AとD I BとC

カ CとD オ BとD

cに関する解答群

- 週ごとに実績進捗率が計画進捗率を下回る傾向にあり、このままではスケジュール遅延
- ▼ 実績進捗率が、第1週の週末時点では計画進捗率と同じであり、第2週の週末時点では

計画進捗率を上回っており、チーム X 全体では進捗に問題はない

- 支績進捗率が、第1週の週末時点では計画進捗率と同じであるが、第2週の週末時点では計画進捗率を下回っており、このままではスケジュール遅延が懸念される
- 工 実績進捗率が、第1週の週末時点は計画進捗率を下回っているが、第2週の週末時点では計画進捗率を上回っており、チーム X 全体では進捗が回復しているので問題はない

設問 2 次の計算式による指標を計画実績工数比といい、各週末時点で算出する。ここで、計画実績工数比は、小数第3位を四捨五入した値である。計画実績工数比を用いた今後のスケジュール分析に関する次の記述中の に入れる適切な答えを、解答群の中から選べ。

計画実績工数比 = 実績工数の累積+残り工数 計画工数の総合計

計画実績工数比は、プロジェクトメンバの増減がないとすると、今後のスケジュールの予測に利用できる。計画実績工数比が 1.00 の場合は、今後も計画どおりのスケジュールで進捗可能と判定できる。1.00 よりも小さい場合は、スケジュール遅延の懸念はないと判定できる。1.00 よりも大きい場合は、今後のスケジュールに遅延が生じる可能性があると判定できる。

チーム X 全体の計画実績工数比は,第1週及び第2週の週末時点のそれぞれの値が d であり,チーム X 全体では計画した期間内に設計工程を終えることが可能であると判定した。次に,メンバ個別の計画実績工数比を週ごとに比較してみると,第2週の週末時点の値が第1週の週末時点の値よりも増加しているメンバ e 名については,スケジュール遅延が懸念されると判定した。

dに関する解答群

ア 0.99 及び 0.95

1 0.99 及び 1.00

ウ 1.00 及び 0.99

エ 1.00 及び 1.00

オ 1.01 及び 0.99

カ 1.01 及び 1.00

e に関する解答群

ア 1

2

ウ 3

設問 3 次の文は、スケジュール遅延の懸念に対する解決策について記述したものである。次の記述中の に入れる適切な答えを、解答群の中から選べ。

第2週の週末時点での残り工数が第3週の計画工数を下回っているメンバのうち、その時点での実績進捗率が最も高い f に、スケジュール遅延の懸念があるメンバの UI 設計の一部を分担させることによって、スケジュール遅延を回避し、計画した期間内に全メンバの UI 設計を完了させることにした。

解答群

ア A

1 B

ウ C

I D

… 間6 3 午後のカギ ……

プロジェクトマネジメントの進捗管理に関する問題です。計画進捗率,実績進捗率,計画実績工数比といった値が出てきますが,すべて問題文に説明があるので,前提となる知識はほとんど必要ありません。

- 設問 1 メンバごとの計画進捗率と実績進捗率を比較します。実績工数の累積+残り工数は、計画工数の総合計と等しいとは限らないことに注意しましょう。
- 設問 2 表 2 から、計画実績工数比を計算します。
- 設問3 表1,表2から,第3週の計画工数と第2週週末時点の残り工数をメンバごとに比較します。 残り工数のほうが少ないメンバのうち,実績進捗率(設問1で算出済み)の高いほうが正解です。

設問 1 計画進捗率と実績進捗率

空欄 a: 各メンバの各週末ごとの計画進捗率は表 1 に記載されており,第1週の週末時点では次のようになっています。

A: 0.0% B: 20.0% C: 20.0%

D: 20.0%

一方.実績進捗率は**表2**より次のように計算します。

A: $\frac{0}{0+50} \times 100 = 0.0\%$

B: $\frac{25}{25+75} \times 100 = 25.0\%$

C: $\frac{20}{20+80} \times 100 = 20.0\%$

D: $\frac{15}{15+85}$ × 100 = 15.0%

以上から,第1週の週末時点で実績進捗率が計画進 捗率を下回っているメンバは,Dです。空欄aは です。

空欄 b:表 1 より, 第 2 週の週末時点の計画進捗率は次のとおりです。

A: 50.0% B: 60.0% C: 60.0% D: 60.0%

一方,実績進捗率は表2より次のように計算します。

A: $\frac{25}{25+20}$ × 100 = 55.6% (小数第2位を四捨五入。 以下同じ)

B: $\frac{65}{65+35} \times 100 = 65.0\%$

C: $\frac{60}{60+45}$ × 100 = **57.1%**

D: $\frac{55}{55+40}$ × 100 = **57.9%**

以上から,第2週の週末時点で実績進捗率が計画進 捗率を下回っているメンバは, $C \ge D$ です。 $\mathbf{2}$ **々** は $\mathbf{5}$ です。

空欄 c: チーム全体の計画進捗率は、**表 1** より、第 1 週が 17.1%、第 2 週が 58.6%です。

一方,実績進捗率は,**表 2** より次のように計算し ます。

第1週: $\frac{60}{60+290}$ ×100=17.1%

第 2 週: $\frac{205}{205 + 140} \times 100 = 59.4\%$

以上から、実績進捗率は第1週の週末時点では計画 進捗率と同じですが、第2週の週末時点では計画進捗 率を上回っています。したがって、空欄 c には、 イ が入ります。

設問2 計画実績工数比

プロジェクトの進捗につれ、当初計画した工数と実際に必要な工数にズレが生じます。計画実績工数比は、このズレの度合いを示します。

計画工数の総合計 350 に対し、実績工数の累積+残

り工数は、**表 2** より、第 1 週目が 60 + 290 = 350、第 2 週目が 205 + 140 = 345 です。したがって、計画実績工数比は次のようになります。

第 1 週:
$$\frac{60 + 290}{350} = 1.00$$

以上から,**空欄 d** は <mark>ウ</mark> です。

空欄 e:表 1,表 2 より,メンバごとの計画実績工数 比を求めます。

第1週	第2週
A: $\frac{0+50}{50}$ = 1.00	$\frac{25+20}{50} = 0.90$
B: $\frac{25+75}{100}$ = 1.00	$\frac{65+35}{100} = 1.00$
C: $\frac{20+80}{100} = 1.00$	$\frac{60+45}{100} = 1.05$
D: $\frac{15+85}{100}$ = 1.00	$\frac{55+40}{100} = 0.95$

以上から、第2週の週末時点の計画実績工数比が、

第1週より増加しているメンバはCのみです。**空欄e**は \mathbb{Z} です。

設問3

表1、表2より、各メンバの第3週の計画工数及び第2週週末時点の残り工数は、次のとおりです。

	第3週の計画工数	第2週週末時点の 残り工数
Α	25	20
В	40	35
С	40	45
D	40	40

以上から、残り工数が計画工数を下回っているのは、メンバ A と B の 2 名です。実績進捗率は、設問 1 (空欄 b の解説) より A が 55.6%, B が 65.0% なので、B のほうが実績進捗率が高いです。空欄 f は 1 です。

○ 解答

設問 1 a ー <mark>ウ</mark> , b ー カ , c ー イ

設問2 dーウ, eーア

設問 3 f - < ✓



正味現在価値(Net Present Value)による投資採算性の評価に関する次の記述を読んで、設問 $1\sim3$ に答えよ。

日用品メーカの D 社では、業務効率の向上及び 2016 年に予定されている新生産設備の稼働への対応を目的として、新たに生産管理システムの導入を検討している。生産管理システムの導入における投資採算性の評価を、企画課で行うことになった。企画課で試算したシステム導入に掛かる投資と効果は次のとおりである。

(1) 投資

生産管理システムの導入の方法として,次の2案を検討した。

- ① 業務効率向上のための投資と新生産設備の稼働対応の投資の両方を2012年に行う(以下, 一括投資という)場合,7億5千万円が必要となる。
- ② 業務効率向上のための投資を 2012 年に, 新生産設備の稼働対応の投資を 2015 年に行う(以下, 分割投資という)場合,2012 年に 5 億円,2015 年に 3 億円が必要となる。
- (2) 効果

(1) のどちらの場合でも、生産管理システムの導入によって、2013年からは年間 2 億円、2016年からは年間 3 億円のコスト削減が見込まれる。

D社では、投資採算性の評価に正味現在価値(以下、NPVという)を用いている。NPVは、キャッシュの現在価値という概念を使っている。現在価値の考え方を用いると、現在の 100 万円は 1 年後の 100 万円よりも価値が高い。なぜなら、現在の 100 万円を 1 年間預金すると利子がついて 1 年後には 100 万円よりも多くなるからである。利子率を 100 万円は現在の約 100 万円と同じ価値ということになる。これがキャッシュの現在価値という概念である。

なお、NPV の計算では、利子率を割引率と呼ぶこともある。

設問 NPV を使った投資採算性の評価に関する次の記述中の に入れる正しい答えを、解答群の中から選べ。

NPV を計算するときは、各年に増減するキャッシュの現在価値を計算する必要がある。割引率を r で複利計算し、n 年後のキャッシュを C と表すとき、その現在価値は、 a と表せる。 なお、投資年については、n=0 で計算する。

将来の各年におけるキャッシュの増減に対して、その現在価値を計算し、次に各年の現在価値を合計することで NPV が求められる。NPV が正のときは、その投資案は採算性があり、NPV が大きいほど採算性が高いと評価される。NPV が負のときは、その投資案は採算性がなく、投資するべきでないと評価される。NPV が正の投資案が二つ以上あるときは、NPV の最も大きな投資案を選択するべきである。

解答群

$$\frac{C}{(1+r)^n}$$

$$\frac{C}{(1-r)^n}$$

$$\frac{C}{r^n}$$

$$C \times (1 + r)^n$$

$$T \quad C \times (1-r)^n$$

設問 2 生産管理システムの投資採算性の評価に関する次の記述中の に入れる適切な答えを、解答群の中から選べ。

企画課では、次の手順で生産管理システム導入の投資採算性の評価を行うことにした。一括投資する場合の生産管理システム導入のNPVを計算する表1と、分割投資する場合のNPVを計算する表2を作成した。

- (1) 2012 年から 2020 年までの期間を対象に NPV を計算する。
- (2) 各年のコスト削減額から減価償却費を差し引いて、各年の課税対象効果を求める。
- (3) 減価償却費は、投資した翌年から4年間の定額償却(残存価額は0円)として求める。
- (4) 法人税率を40%とし、各年の課税対象効果から法人税として控除される法人税額を求める。
- (5) 各年のコスト削減額から法人税額を差し引いて税引き後効果を求め、更に税引き後効果から 各年の投資額を差し引いて各年の投資後効果(キャッシュ増減額)を求める。
- (6) 割引率を 0.08 とし、各年の投資後効果の現在価値を計算する。
- (7) (6) で計算した各年の現在価値を合計し、投資案の NPV を計算する。

表 1 一括投資する場合の生産管理システム導入の NPV 計算

単位 百万円

年	2012	2013	2014	2015	2016	2017	2018	2019	2020
投資額	750								
減価償却費	0	187.5	187.5	187.5	187.5	0	0	0	0
コスト削減額	0	200	200	200	300	300	300	300	300
法人税額	0	5	5	5	45	120	120	120	120
税引き後効果	0	195	195	195	255	180	180	180	180
投資後効果	— 750								
割引率	0.08	b	0.08	0.08	0.08	0.08	0.08	0.08	0.08
現在価値	— 750								
NPV	378.1								

注記 網掛けの部分は表示していない。 現在価値は小数第2位を四捨五入している。

表 2 分割投資する場合の生産管理システム導入の NPV 計算

単位 百万円

年	2012	2013	2014	2015	2016	2017	2018	2019	2020
投資額	500			300					
減価償却費	0	125	125		200	75	75	75	0
コスト削減額	0	200	200		300	300	300	300	300
法人税額	0	30	30	C	40	90	90	90	120
税引き後効果	0	170	170		260	210	210	210	180
投資後効果	– 500								
割引率	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08
現在価値	— 500	157.4	145.7	-103.2	191.1	142.9	132.3	122.5	97.2
NPV									

注記 網掛けの部分は表示していない。 現在価値は小数第2位を四捨五入している。

企画課では,表1と表2の結果から d と判断した。

b に関する解答群

ア	— 195
	0.08
	— 210.6

_	
	— 195
	0.08
	— 195

I	— 195
	0.08
	0

オ	195
	0.08
	0

カ	195
	0.08
	180.6

丰	195
	0.08
	195

ク	195
	0.08
	210.6

cに関する解答群

_	
ア	125
	200
	30
	170

1	125
	300
	70
	230

ウ	200
	200
	0
	16

I	200
	300
	40
	260

d に関する解答群

- ア 一括投資、分割投資のどちらの投資案にしてもよい
- ✓ 一括投資,分割投資のどちらの投資案もするべきでない
- ウ 一括投資するべきである
- エ 分割投資するべきである

設問 3 生産管理システムの NPV の再計算に関する次の記述中の に入れる適切な答えを、解答群の中から選べ。

なお、解答は重複して選んでもよい。

企画課で作成した表 1,表 2 を社内の関係部署に説明したところ、生産課から、生産管理システムに新たな機能を追加したいとの要望があった。新たな機能の追加には、2012年に5千万円の追加投資が必要となる。その場合、生産管理システムの機能が向上し、追加効果は年間で5百万円の見込みである。

機能を追加した場合の NPV を企画課で計算した結果,一括投資の場合の NPV は 3 億 6,190 万円,分割投資の場合の NPV は 3 億 7,000 万円であった。企画課では,これらの結果と**表 1**,**表 2** から,生産課の要望については e と判断した。

また、経理課から、D社では情報システムの減価償却期間を4年間ではなく5年間で行っているので、表1、表2もそのように修正するべきであるとのコメントがあった。減価償却期間を5年に変更すると、表1の各年の減価償却費は表3のように変わる。

表3 減価償却期間を5年に変更したときの表1の各年の減価償却費

単位 百万円

年	2012	2013	2014	2015	2016	2017	2018	2019	2020
減価償却費	0	150	150	150	150	150	0	0	0

表3によると、2013年から2016年までの各年の減価償却費は表1に比べて下がり、2017年の減価償却費は表1に比べて上がり、2018年以降の減価償却費は表1と同じであることが分かる。したがって、2013年から2016年までの税引き後効果は f ことになり、2017年の税引き後効果は g ことになる。2020年までの減価償却費の総額は変わらないものの、現在価値の考え方を考慮すると、NPV は h と考えられる。これは、表2についても同様である。

企画課では、経理課のコメントに従って表1と表2を修正して、再度投資採算性を評価した。

e に関する解答群

- ア 一括投資、分割投資のどちらの場合も受け入れるべきではない
- → 一括投資、分割投資のどちらの場合も受け入れるべきである
- ウ 一括投資の場合だけ受け入れるべきである

f~hの解答群

ア 表1と変わらない イ 表1に比べて上がる ウ 表1に比べて下がる

問って、午後のカギ

正味現在価値(NPV)によって、投資採算性を評価する問題です。NPV については問題文にも説明があります。解答には細かい数値の計算が必要なので、計算間違いに注意してください。

設問 1 キャッシュの現在価値の計算方法を求めます。ここが正答できないと、後の設問に影響します。

設問 2 各欄の計算方法は、問題文(1)~(7)に説明があります。最後に表 1 と表 2 の NPV を比較して、どちらの投資採算性が高いかを評価します。

設問3 減価償却費が下がると法人税が上がります。法人税が上がると税引き後効果は下がり、現在価値も下がります。

設問 1 現在価値を求める式

割引率を 0.05 とすれば、現在の 100 万円は、

1 年後:100 × (1 + 0.05) = 105 万円 2 年後:105 × (1 + 0.05) = 110.25 万円

:

n 年後:100 × (1 + 0.05)ⁿ 万円

となります。現在のキャッシュをP, 割引率をr, n 年後のキャッシュをC とすると.

$$C = P \times (1 + r)^{-n} : P = \frac{C}{(1 + r)^{n}}$$

以上から、正解は ア です。

設問2 投資案の検討

空欄 b∶表 1 における 2013 年の投資後効果,割引率, 現在価値を求めます。

・投資後効果

(5) より、投資後効果は税引き後効果から各年の投資額を差し引いたものです。2013年の税引き後効果は195、投資額は0なので、次のようになります。

投資後効果= 195 - 0 = 195

・割引率

(6) より、割引率は 0.08 です。

・現在価値

現在価値は、 $C/(1+r)^n$ で求めます。C=195百万円,r=0.08,n=1年後より,

以上から、**空欄 b** には、上から 195, 0.08, 180.6 が入ります。**空欄 b** は <mark>カ</mark>です。

空欄 c:表2における2015年の減価償却費,コスト削減額,法人税額,税引き後効果を求めます。

・減価償却費

減価償却費は、投資した翌年から 4 年間の定額償却となります。2012 年の投資額 500 百万円は、2013 ~16 年に 125 百万円ずつ償却されます。また、2015 年の投資額 300 百万円は、2016 ~ 19 年にかけて 75 百万円ずつ償却されます。

以上から,2015年の減価償却費は125百万円です。

・コスト削減額

コスト削減額は、2013年から200百万円、2016年から300百万円になります。したがって2015年のコスト削減額は200百万円です。

・法人税額

コスト削減額-減価償却費が課税対象となります。 2015年の課税対象額は 200-125=75百万円。税率は 40%なので,法人税額は $75\times40\%=30$ 百万円になります。

・税引き後効果

コスト削減額-法人税額が税引き後効果となります。2015年の税引き後効果は、200-30=170百万円です。

以上から、空欄 ${\bf c}$ には上から 125, 200, 30, 170 が入ります。空欄 ${\bf c}$ は ${\bf r}$ です。

空欄 d:表1と表2の NPV を比較します。表1の NPV は378.1 がすでに入っているので,ここでは表2の NPV を求めます。

NPV は、各年の現在価値の合計で求められます。

- 500 + 157.4 + 145.7 - 103.2 + 191.1 + 142.9 + 132.3 + 122.5 + 97.2 = 385.9 百万円

以上から、NPV は**表 1** (一括投資) が 3 億 7,810 万円、**表 2** (分割投資) が 3 億 8,590 万円です。NPV の大きなほうの投資案を選択するので、分割投資を選択すべきということがわかります。正解は **エ**です。

設問3 NPVの再計算

空欄 e:機能を追加した場合の NPV は、3 億 6,190 万円 (一括投資) または 3 億 7,000 万円 (分割投資) で、いずれも追加前の NPV より小さくなります。よって、一括投資、分割投資のどちらの場合も受け入れるべきではありません。正解は アです。

空欄 f: 税引き後効果は、コスト削減額-法人税額で求めます。また、法人税額は、(コスト削減額-減価償却費)×法人税率で求めます。したがって減価償却費が下がると、法人税額は上がり、税引き後効果は下がります。空欄 f は っです。

空欄 g: 逆に減価償却費が上がると, 法人税額は下がり, 税引き後効果は上がります。空欄 g は 1 です。

空欄 h: 税引き後効果の減少分(2013 ~ 16 年)は、1年に付き (187.5 - 150) × 40% = 15 百万円です。したがって NPV の減少分は、次のように計算できます。

一方、税引き後効果の増加分(2017年)は $150 \times 40\% = 60$ 百万円なので、NPV の増加分は次のように計算できます。

$$\frac{60}{(1.08)^5}$$
 ...2

①を次のように変形すると,

$$\frac{15 (1.08)^{4} + 15 (1.08)^{3} + 15 (1.08)^{2} + 15 (1.08)}{(1.08)^{5}}$$

となります。この値は②より大きいので、NPVの減少分は増加分を上回り、NPV は $\mathbf{表}$ $\mathbf{1}$ に比べて下がります。

以上から. **空欄 h** は **ウ** です。

キャッシュの現在価値は先にいくほど 低くなるよ。



○ 解答

設問 1 a - ア

設問 2 b - D, c - P, d - I

設問3 e-ア, f-ウ, g-イ,

h 一 💆

次の問8は必須問題です。必ず解答してください。

8

次のプログラムの説明及びプログラムを読んで、設問 $1 \sim 3$ に答えよ。

整数型関数 BitTest は、8 ビットのデータ中の指定したビット位置にあるビットの値を検査して、結果を返す。整数型関数 BitCount は、8 ビットのデータ中にある1のビットの個数を返す。なお、本間において、演算子 "&"、" | " は、二つの8 ビット論理型データの対応するビット位置のビット同士について、それぞれ論理積、論理和を求め、8 ビット論理型で結果を得るものとする。また、" ~ "B という表記は、8 ビット論理型定数を表す。

〔プログラム 1 の説明〕

整数型関数 BitTest を,次のとおりに宣言する。

○整数型関数:BitTest (8ビット論理型:Data, 8ビット論理型:Mask)

検査される8ビットのデータは入力用の引数Dataに、検査をするビット位置の情報は入力用の引数Maskに、それぞれ格納されている。Mask中のビットの値が1であるビット位置に対応したData中のビットを検査して、次の返却値を返す。ここで、Mask中には1のビットが1個以上あるものとする。

返却値 0:検査した全てのビットが0

1:検査したビット中に0と1が混在

2:検査した全てのビットが1

例えば、図1の例1では、Maskのビット番号 $7\sim5$ の3ビットが1であるので、Dataのビット番号 $7\sim5$ の3ビットの値を検査し、0と1が混在しているので返却値1を返す。例2では、Maskのビット番号4と0の2ビットが1であるので、Dataのビット番号4と0の2ビットの値を検査し、どちらも1であるので返却値2を返す。

図 1 BitTestの実行例

〔プログラム 1〕

```
○整数型関数:BitTest (8ビット論理型:Data, 8ビット論理型:Mask)
```

・return RC /* RCを返却値として返す */

〔プログラム2,3の説明〕

整数型関数 BitCount を,次のとおりに宣言する。

○整数型関数:BitCount (8ビット論理型:Data)

検査される8ビットのデータは入力用の引数Dataに格納されている。

このためのプログラムとして、基本的なアルゴリズムを用いたプログラム2と、処理効率を重視したプログラム3を作成した。

プログラム 2, 3 中の各行には、ある処理系を想定して、プログラムの各行を 1 回実行するときの処理量 $(1, 2, \cdots)$ を示してある。選択処理と繰返し処理の終端行の処理量は、それぞれの開始行の処理量に含まれるものとする。

なお、演算子"ー"は、両オペランドを8ビット符号なし整数とみなして、減算を行うものとする。

〔プログラム2〕

(処理量)

- ○整数型関数:BitCount (8ビット論理型:Data)
- ○8ビット論理型:Work
- ○整数型:Count, Loop
- 1 ·Work ← Data
- 1 · Count \leftarrow 0
- 4 Loop: 0, Loop < 8, 1
- 3 **▲** Work の最下位ビットが 1
- 1 | Count \leftarrow Count + 1
- 1 ・Work を右へ1ビット論理シフトする
- 2 return Count
- /* Count を返却値として返す */

〔プログラム3〕

(処理量)

- ○整数型関数:BitCount (8ビット論理型:Data)
- ○8ビット論理型:Work
- ○整数型:Count
- 1 Work ← Data
- 1 Count \leftarrow 0
- 2 Work 中に 1 のビットがある
- 1 \cdot Count \leftarrow Count + 1
- 3 Work ← Work & (Work − 1)



設問 プログラム 1 中の に入れる正しい答えを, 解答群の中から選べ。

解答群

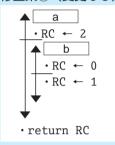
7 (Data & Mask) = "00000000"B ✓ (Data & Mask) = Data

(Data | Mask) = "00000000"B (Data & Mask) = Mask

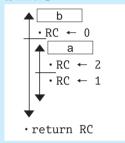
次の記述中の 入れる正しい答えを、解答群の中から選べ。

プログラム1は、Mask中に1のビットが1個以上あることを前提としている。ここで、この前 提を取り除いて、Mask中の1のビットが0個の場合は返却値0を返すようにしたい。そのために、 プログラム1の処理部分について、次の修正案①~③を考えた。ここで、修正案①は、プログラ ム1のままで何も変更しない。また、 a と b には、設問1の正しい答えが入ってい るものとする。

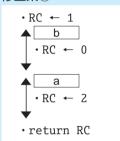
修正案(1) (変更なし)



修正案②



修正案③



これらの修正案のうち、正しく動作するのは c である。

解答群

ア 修正案①

工 修正案①及び②

✓ 修正案②

オ 修正案①及び③

ウ 修正案③

カ 修正案②及び③

次の記述中の |に入れる正しい答えを、解答群の中から選べ。

プログラム 2, 3 の処理効率について考えてみる。表 1 にプログラム 2, 3 の処理量の比較結果 を示す。

表 1 プログラム 2, 3 の処理量の比較

	最小	最大
プログラム 2	72	d
プログラム 3	е	54

プログラム 3 では、 α の行での変数 Work の更新において効率の良いアルゴリズムが使われて いる。例えば、プログラム 3 で引数 Data の内容が "01101010"B であったとき、繰返し処理に おいて α の行の 2 回目の実行が終了した時点で変数 Work の内容は、" f "B になっている。このようなビット変換の処理によって、繰返し処理の繰返し回数は、検査されるデータ中の1のビットの個数と同じになる。

d に関する解答群

\overline{z}	90
ノ	80

4 88

ウ 104

I 112

e に関する解答群

10

D 20

I 22

fに関する解答群

ア	0.0	00	00	11

00001010

01010000

01100000

10100000

□8 ∜ 午後のカギ

ビット列中のビットを検査するアルゴリズムに関する問題です。

- 設問 1 論理積や論理和演算を利用すると、特定のビットを 0 にしたり、 1 にしたりできます。これらは、午前問題でも出題される基礎知識です。
- **設問 2** Maskの内容が "00000000" の場合に、返却値が 0 になるものを選びます。設問 1 を正しく 解答している必要があります。
- 設問 3 プログラム 2, 3 の処理量は, Data の内容が "00000000" のとき最小, "11111111" のとき 最大になります。

設問 1 関数 BitTest の完成

空欄 a:空欄には、選択処理の条件式が入ります。



条件式が真のとき,返却値 RC に 2 を設定します。問題文より,返却値 2 は「**検査した全てのビットが 1**」であることを意味するので,**空欄 a** には,Data 中の検査ビットがすべて 1 であることを確認する式が入ると考えられます。

たとえば、引数 Mask の内容が "11110000" だとして、引数 Data の内容が "1111XXXX" (X は 0 でも 1 でも 可) であることは、どうすれば確認できるで

しょうか?

DataとMaskの論理積(AND)を求めると、次のようになります。

1111XXXX ← Data <u>AND 11110000</u> ← Mask 11110000

上のように、Dataの検査ビットがすべて1の場合、 検査ビットは両方とも1、それ以外はMaskが0となるため、結果はMaskと同じ値になります。したがって空欄aに入る条件式は、次のように記述できます。

(Data & Mask) = Mask

以上から,**空欄 a** は <mark>ウ</mark> です。

空欄 b:空欄には、選択処理の条件式が入ります。

$$\begin{array}{c|c}
 & b \\
 & RC \leftarrow 0 \\
\hline
 & RC \leftarrow 1
\end{array}$$

条件式が真のとき、返却値RCに0を設定します。 返却値0は「検査した全てのビットが0」であることを 意味するので、空欄bには、Data中の検査ビットがす べて0であることを確認する式が入ると考えられます。

たとえば、引数 Mask の内容が "11110000", 引数 Data の内容が "0000XXXX"(X は 0 でも 1 でも可) として、Data と Mask の論理積 (AND) を求めると、次のようになります。

上のように、Dataの検査ビットがすべて0の場合、DataかMaskのどちらかのビットが必ず0になるので、結果はすべてのビットが0になります。したがって空欄bに入る条件式は、次のように記述できます。

(Data & Mask) = "00000000"B

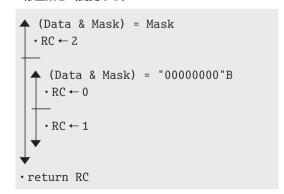
以上から、**空欄 b** は ア です。

設問2 プログラムの修正

 ${\tt Mask}$ の内容が "00000000" だった場合に、返却値 ${\tt 0}$ を返すようにプログラムを修正します。

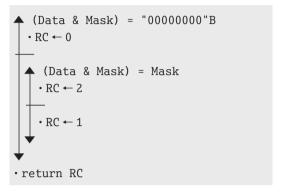
修正案①~③を順に検討しましょう。

・修正案① (変更なし)



Mask の内容が "00000000" だった場合, Data & Mask の値は 0 になります。すなわち, (Data & Mask) = Mask は真になり, 返却値 RC に 2 が設定されます。したがって, この修正案は正しく動作しません。

・修正案②



Maskの内容が"00000000"だった場合,(Data & Mask) = "00000000"B は真になるので,返却値RCには0が設定されます。この修正案は正しく動作します。

・修正案③

Maskの内容が"00000000"だった場合,(Data & Mask) = "00000000"B は真になるので,返却値RCには0が設定されます。

しかし、この修正案ではその後、条件式 (Data & Mask) = Mask も評価されます。Maskの内容が "00000000" だった場合、(Data & Mask) = Maskも真になるので、返却値RCは2に上書きされてしまいます。したがって、この修正案は正しく動作しません。

以上から,正しく動作するのは修正案2のみです。 正解は1です。

設問3 プログラムのトレース

プログラム 2, 3 ともに, 処理量が最小になるのは Data の内容が "00000000" のとき, 処理量が最大になるのは Data の内容が "11111111" のときです。

空欄 d: Data の内容が "11111111" のとき,プログラム 2 の処理量は次のようになります。

処理量(最大)

1
1
4×9
3 × 8
1 × 8
1 × 8
2
80

ループ内の処理は全部で8回繰り返されます。条件 式「Workの最下位ビットが1」は8回とも真になる ので、「Count ← Count + 1」も8回実行されます。 以上から、プログラム2の最大の処理量は80です。 空欄 d は 7です。

繰返しは条件が偽になるまで繰り返すので、条件式の評価は9回実行されることに注意。



空欄 e: Data の内容が "00000000" のとき,プログラム 3 の処理量は次のようになります。

処理量(最小)

•Work ← Data		1
• Count ← 0		1
■ Work 中に 1 のビットがある	;	2 X 1
•Count ← Count + 1 •Work ← Work & (Work - 1)		1×0
·Work ← Work & (Work — 1)		3×0
•		
·return Count		2
	合計	6

Dataの内容が "00000000" のときは、繰返し条件[Work中に1のビットがある] が偽になるため,ルー

プ内の処理は一度も実行されません。以上から、プログラム3の最小の処理量は6になります。**空欄e**はアです。

空欄 f: Work - 1 の結果は、Work 中の 1 のビット のうち、最も下位にあるものを 0 にし、以降のビット をすべて 1 にします。

Work

例: $01000000 - 1 \rightarrow 001111111$

したがって、Work & (Work -1) は、Work 中の最も下位にある 1 ビットを 0 にします。

引数 Data の内容が "01101010" のとき、「Work ← Work & (Work − 1)」は全部で4回実行されます。それぞれの処理は、次のようになります。

Work Work - 1

- ① 01101010 & 01101001 \rightarrow 01101000
- ② $01101000 & 01100111 \rightarrow 01100000$
- $301100000 & 01011111 \rightarrow 01000000$
- (4) 01000000 & 001111111 \rightarrow 00000000

° 解答 °

設問 1 a ー ウ , b ー ア

設問 2 c - 1

設問3 dーア, eーア, fーオ

次の問9から問13までの5問については、この中から1問を選択し、選択した問題については、答案用紙の選択欄の(選)をマークして解答してください。 なお、2問以上マークした場合には、はじめの1問について採点します。



次の C プログラムの説明及びプログラムを読んで、設問 1, 2 に答えよ。

〔プログラム 1 の説明〕

S社では、通常勤務時間帯(8時30分~18時)中の各社員の予定を管理している。 関数 allmem_free_slot は、通常勤務時間帯に会議を計画する場合に、出席して欲しい社員(以下、出席社員という)全員の予定が入っていない時間帯を調べるプログラムである。

(1) S社では、各社員の予定を30分単位の時間帯ごとに図1に示すスケジュール表で管理している。

時間帯番号:	0	1	•••	17	18
	8:30~9:00の 予定	9:00~9:30の 予定		17:00~17:30の 予定	17:30~18:00の 予定

図1 スケジュール表

- ① スケジュール表は大きさ19の整数型配列で、各要素の添字の値は時間帯番号に対応している。
- ② スケジュール表の各要素には、予定が入っている場合には0以外の値が、予定が入っていない場合には0が入る。
- (2) 会議時間は30分以上であり、開始時刻と終了時刻は時間帯の区切りと一致する。
- (3) 関数 allmem_free_slot の引数は次のとおりである。ここで、引数の値に誤りはないものとする。

num 出席社員数

time_tbl 出席社員全員のスケジュール (スケジュール表へのポインタの配列)

slot num 会議時間 (30 分単位の時間帯数)

free 会議開催の候補時間帯 (SLOT 型の配列)

(4) 構造体を使った SLOT 型の定義は次のとおりである。

```
typedef struct {
    int s_start; /* 開始時間帯番号 */
    int s_num; /* 時間帯数 */
} SLOT;
```

(5) 候補時間帯 free には、出席社員全員の予定が入っていない、会議時間以上の連続する時間帯 (以下、可能時間帯という)を全て格納する。最後に格納した要素の次の要素の開始時間帯番号には、-1を格納する。例えば、出席社員5名(社員A~E)の予定が**図2**のとおりであった場合、会議時間が1時間(slot_num = 2)のとき、可能時間帯の一覧は表1になる。

時刻: 8:	30 9:	00 9:	30 10	:00 10	:30 11	:00 11	:30 12	:00 12	:30 13	:00 13	:30
時間帯番号:	0	1	2	3	4	5	6	7	8	9	
社員A	0	0	0	0	×	×	0	×	×	0	
社員B	0	0	0	×	×	0	0	×	×	0	
社員C	0	0	0	×	0	0	0	×	×	0	
社員D	0	0	0	×	×	0	0	×	×	0	
社員E	0	0	0	×	×	0	0	×	×	0	
		-		-	-			-	-		-

時刻:13:30 14:00 14:30 15:00 15:30 16:00 16:30 17:00 17:30 18:00

時間帯番号:	10	11	12	13	14	15	16	17	18
社員A	×	0	0	×	0	×	0	0	0
社員B	×	0	0	\circ	0	×	0	0	0
社員C	×	0	0	×	0	×	0	0	0
社員D	0	0	0	×	0	×	0	0	0
社員E	X	0	0	0	0	0	0	0	0

注記 "○"は予定が入っていないことを表し、"X"は予定が入っていることを表す。

図2 出席社員全員のスケジュールの例

表 1 可能時間帯の一覧の例

可能時間帯	開始時間帯番号	時間帯数
8:30 ~ 10:00	0	3
14:00 ~ 15:00	11	2
16:30 ~ 18:00	16	3

(6) 関数 allmem_free_slot で使用している関数 search_free_slot の仕様は次のとおりである。

機能:スケジュール表から、予定が入っていない、時間帯数以上の連続した時間帯を全て探し出し、空き時間帯に格納する。最後に格納した空き時間帯の要素の次の要素の開始時間帯番号には、-1を格納する。

引数:sch スケジュール表

slot num 時間帯数

free 空き時間帯 (SLOT 型の配列)

〔プログラム 1〕

```
#define NUM_DAY 19 /* 1日の時間帯数 */

typedef struct {
    int s_start; /* 開始時間帯番号 */
    int s_num; /* 時間帯数 */
} SLOT;

void allmem_free_slot(int, int *[], int, SLOT []);
void search_free_slot(int [], int, SLOT []);
```

```
void allmem_free_slot(int num, int *time_tbl[], int slot_num,
                    SLOT free[]) {
   int sch[NUM_DAY], i, j;
   for (i = 0; i < NUM_DAY; i++) {
       sch[i] = a;
       for (j = 1; j < num; j++) {
           if (time_tbl[j][i] != 0) {
               sch[i] = 1;
           }
       }
   }
   search_free_slot(sch, slot_num, free);
}
void search_free_slot(int sch[], int slot_num, SLOT free[]) {
   int cnt = 0, free_num = 0, i;
   for (i = 0; i < NUM_DAY; i++) {</pre>
       if (sch[i] != 0){
           if (cnt >= slot_num) {
               free[free_num].s_start = ___
               free[free_num].s_num = cnt;
               free_num++;
       } else {
           cnt++;
       }
   if (
           d ) {
       free[free_num].s_start = NUM_DAY - cnt;
       free[free_num].s_num = cnt;
       free_num++;
   free[free_num].s_start = -1;
}
                        に入れる正しい答えを、解答群の中から選べ。
         プログラム 1 中の
aに関する解答群
  7 0
                                                   time_tbl[0][0]
                             1
  time_tbl[0][i]
bに関する解答群
  ァ i

✓ i - cnt

    i - cnt + 1
```

cに関する解答群

dに関する解答群

設問 2 出席社員が増え、出席社員全員での会議の計画ができない(可能時間帯がない)場合がでてきた。そこで、出席社員を出席が必須の社員(以下、必須出席社員という)とそれ以外の社員(以下、任意出席社員という)に分け、必須出席社員全員が出席可能な時間帯で会議の計画をすることにした。さらに、任意出席社員について順位付けを行い、順位の高い方から最多の社員が出席可能な時間帯に絞込みを行うことにした。この条件で会議が計画できる時間帯を調べるプログラムを作成した。作成したプログラム中の に入れる正しい答えを、解答群の中から選べ。ここで、 a には、設問1の正しい答えが入っているものとする。

〔プログラム2の説明〕

(1) 関数 maxmem_free_slot の引数は次のとおりである。ここで、引数の値に誤りはないものとする。

num 出席社員数

must_num 必須出席社員数

time tbl 出席社員全員のスケジュール(スケジュール表へのポインタの配列)

slot_num 会議時間(30分単位の時間帯数)

free 会議開催の候補時間帯 (SLOT 型の配列)

- ① 必須出席社員のスケジュール表へのポインタは、time_tbl の先頭から人数分が格納されている。
- ② 任意出席社員のスケジュール表へのポインタは、time_tbl の必須出席社員のスケジュールに続けて、順位の高い方から順に格納されている。
- (2) 候補時間帯 free には、必須出席社員全員が出席可能な時間帯の中で、任意出席社員の順位の高い方から最多の社員が出席可能な時間帯を全て格納する。最後に格納した要素の次の要素の開始時間帯番号には、-1を格納する。

〔プログラム2〕

```
#define NUM_DAY 19 /* 1日の時間帯数 */

typedef struct {
   int s_start; /* 開始時間帯番号 */
   int s_num; /* 時間帯数 */
} SLOT;
```

```
void maxmem_free_slot(int, int, int *[], int, SLOT []);
void search_free_slot(int [], int, SLOT []);
void maxmem_free_slot(int num, int must_num, int *time_tbl[],
                    int slot_num, SLOT free[]) {
   int sch[NUM_DAY], i, j, k;
   SLOT wk_free[2][NUM_DAY + 1];
   /* 必須出席社員全員が出席可能な時間帯を探し出す */
   for (i = 0; i < NUM_DAY; i++) {
       sch[i] = a;
               e ; j++) {
       for (
           if (time_tbl[j][i] != 0) {
               sch[i] = 1;
           }
       }
   search_free_slot(sch, slot_num, wk_free[0]);
   /* 任意出席社員の順位の高い方から
      最多の社員が出席可能な時間帯に絞り込む */
   for (j = must_num; j < num; j++) {</pre>
       for (i = 0; i < NUM_DAY; i++) {
           if (time_tbl[j][i] != 0) {
               sch[i] = 1;
           }
       search_free_slot(sch, slot_num, wk_free[1]);
       if ( f ) {
           break:
       }
       k = 0;
       do {
           wk_free[0][k] = wk_free[1][k];
       } while (wk_free[1][k++].s_start != -1);
   }
   /* 候補時間帯を格納する */
   k = 0:
   do {
       free[k] = wk_free[0][k];
   } while (wk_free[0][k++].s_start != -1);
}
e に関する解答群
  j = 1; j < must_num</pre>
                                            j = must_num; j < num</pre>
  j = 1; j < num - must_num</pre>
```

間9 3年後のカギ

二次元配列を検索する C プログラムの問題です。

設問 1 空欄 a は間違いやすい問題です。内側の for 文で,変数 j の初期値が 1 になっていることに注意しましょう。

設問 2 プログラムは、はじめに必須出席社員だけの候補時間帯リストを作り、出席者を徐々に増やして、候補時間帯リストを絞り込んでいきます。処理の流れを理解すれば、解答はおのずと分かります。

設問 1 プログラムの完成

空欄 a:空欄の前後は次のとおりです。

```
for (i = 0; i < NUM_DAY; i++) {
    sch[i] = a ;
    for (j = 1; j < num; j++) {
        if (time_tbl[j][i] != 0) {
            sch[i] = 1;
        }
    }
}</pre>
search_free_slot(sch, slot_num, free);
```

空欄 a には,配列 sch の要素に代入する式が入ります。

配列 sch には、出席社員全員のスケジュールが空いている時間帯には 0 が、出席社員のうち 1 人でも予定が入っている時間帯には 1 が設定されます。

たとえば、出席社員が5名の場合、各出席社員のスケジュール表 $time_tbl[0] \sim time_tbl[4]$ の時間帯番号iが順番に調べられ、その時間帯に1人でも予定が入っている場合は、sch[i]に1が設定されます。

	0	1	2	3	4	← i
<pre>time_tbl[0]</pre>	0	0	0	0	1	•••
<pre>time_tbl[1]</pre>	0	0	0	1	1	•••
<pre>time_tbl[2]</pre>	0	0	0	1	0	•••
<pre>time_tbl[3]</pre>	0	0	0	1	1	•••
<pre>time_tbl[4]</pre>	0	0	0	1	1	•••
	+	+	+	+	+	
sch	0	0	0	1	1	•••

この処理は二重ループで実行しますが、内側の for 文は次のようになっています。

```
for (j = 1; j < num; j++) {
    if (time_tbl[j][i] != 0) {
       sch[i] = 1;
    }
}</pre>
```

変数 j の初期値が 1 になっているよ!



内側の for 文は、 $time_tbl[0]$ のスケジュール を調べていません。 $time_tbl[0]$ の各要素を、配列 sch の各要素の初期値として設定しておけば、内側の for 文の繰返しを 1 回分省略できるからです。

したがって、**空欄 a** を含む代入文は、次のようになります。

```
sch[i] = time_tbl[0][i];
```

以上から,空欄 a には \bot が入ります。

空欄 b, c:空欄の前後は次のとおりです。

配列 s ch の各要素を先頭から順に調べ,予定が空いている(s ch [i] = 0)場合は,変数 cnt v c l 増やします。したがって変数 cnt には,連続する空き時間帯の数が入ります。

予定が入っている(sch[i] != 0)場合には、その前までの変数 cntの値を調べ、それが会議時間 slot_num以上なら、会議開催の候補時間として、候補時間帯 free に登録します。空欄 b はこの登録処理の一部です。

```
free[free_num].s_start = b ;
free[free_num].s_num = cnt;
```

free.s_startには、候補時間帯の開始時間帯番号が入ります。たとば cnt が 3 の場合は、現在の時間帯 i の 3 つ前が開始時間帯になります。

```
s_start i sch 1 0 0 0 1 ...
```

したがって、free[free_num].s_startには、 i - cnt の値を代入すればよいとわかります。空欄 b はイです。

空欄 c は、候補時間帯 free の登録が終わった後に 実行しなければならない処理です。次の候補時間帯を 検索するために、変数 cnt の値をいったんリセット する必要があります。

```
cnt = 0:
```

以上から、**空欄 c** は ア です。

空欄 d:空欄の前後は次のとおりです。

```
if ( __d__) {
   free[free_num].s_start=NUM_DAY-cnt;
   free[free_num].s_num =cnt;
   free_num++;
}
```

この if 文は,配列 sch の要素を調べる for 文を抜けた後に実行されます。

for 文の内側では、sch[i]が0以外の値になったとき、その前までの連続した空き時間帯をfreeに登録していました。

ところが、schの最後の要素が0だった場合、それまでの空き時間は登録されず、for文を抜けてしまいます。

その場合には、このif文のブロックで候補時間帯を追加します。ただし、候補時間帯を追加するのは、cntの値が会議時間以上の場合に限ります。したがって空欄dに入る条件式は、

```
cnt >= slot_num
```

となります。空欄 d は ウ です。

設問2 プログラムの修正

空欄 e:空欄の前後は次のとおりです。

```
for (i = 0; i < NUM_DAY; i++) {
    sch[i] = time_tbl[0][i]; ←プログラム1
    for ( e ; j++) {
        if (time_tbl[j][i] != 0) {
            sch[i] = 1;
        }
    }
}
```

search free slot(sch, slot num, wk free[0]);

配列 s ch の各要素は、必須出席社員全員が空いている時間帯であれば 0 になります。time_tbl の先頭から must_num 個分が、必須出席社員のスケジュール表なので、内側の for 文では、

```
time_tbl[1][i] から
time_tbl[must_num-1][i]
```

までの要素を調べます。したがって、**空欄 e** を含む for 文は.

```
for (j = 1; j < must_num; j++) {
となります。空欄eはアです。
```

空欄 f:空欄の前後は次のとおりです。

プログラムの前半で、必須出席社員全員が出席可能な候補時間帯リストは、 $wk_free[0]$ に格納されます。

次に、任意出席社員のスケジュール表を1人分追加して、配列 sch の内容を更新し、必須出席社員全員+任意出席社員1人が出席可能な候補時間帯リストをwk_free[1]に格納します。

 $wk_free[1]$ は $wk_free[0]$ より出席者の多い 候補時間帯リストなので、こちらを新しい $wk_free[0]$ にします。

この処理を、wk_free[1]に候補時間帯がなくなるまで繰り返すと、wk_free[0]に最も出席者の多い候補時間帯リストが格納されます。

空欄fを含むif文は、wk_free[1]に候補時間帯がなくなった場合に、繰返しから抜ける処理です。wk_free[1]に候補時間帯がない場合、wk_free[1]には、関数search_free_slotによって先頭の要素のs_startに-1が格納されます。したがって、if文の条件式は次のようになります。

```
if (wk_free[1][0].s_start == -1) {
    break;
}
```

以上から,**空欄f**は<u>オ</u>です。

```
    解答
    設問 l aーエ, bーイ, cーア, dーウ
    設問 2 eーア, fーオ
```

問 $oldsymbol{10}$ 次の COBOL プログラムの説明及びプログラムを読んで,設問 1,2 に答えよ。

(プログラムの説明)

ある遊園地では、ICカードを利用した精算システムを導入した。入園時に、入園者から入園料を受け取ると、入園者番号を記録したICカードを渡し、入園者番号と年齢層の情報を入園者ファイルに格納する。各施設では、提示されたICカードから入園者番号を読み取り、施設利用実績をイベントファイルに格納する。退園時に、入園者からICカードが返却されたときに、施設利用料金を一括で徴収する。

このプログラムは、入園者ファイル及びイベントファイルに格納されたデータを読み込み、入

園者に関する情報を集計して表示する。

(1) 入園者ファイルは、図1に示すレコード様式の、入園者番号を主キーとする索引ファイルで ある。ファイルは、営業日ごとに作成する。

> 入園者番号 年齢層コード 4桁 1桁

図1 入園者ファイルのレコード様式

- ① 入園者番号は、営業日ごとに一意となる、1から始まる4桁の数字である。
- ② 年齢層コードには、 $4 \sim 12$ 歳は 1、 $13 \sim 18$ 歳は 2、 $19 \sim 29$ 歳は 3、 $30 \sim 49$ 歳は 4、50 歳 以上は5を格納する。4歳未満は入園料及び遊戯施設利用料金が無料となるので、ICカード は発行せず、入園者ファイルにも格納しない。
- (2) イベントファイルは、**図2**に示すレコード様式の順ファイルである。入園者の IC カード利用 状況(以下、イベントという)を格納する。ファイルは、営業日ごとに作成する。

入園者番号	イベントコード	時刻	金額
4桁	2桁	6桁	6桁
. 113	- 113	0 113	0 113

図2 イベントファイルのレコード様式

① イベントコードには、発生したイベントに応じて、表1に示すイベントコードを格納する。

表 1 イベントとイベントコード

イベント	イベントコード	補足
入園	01	IC カード発行
退園	02	IC カード返却
観覧車	03	遊戱施設利用
ゴーカート	04	遊戱施設利用
ジェットコースター	05	遊戱施設利用
メリーゴーランド	06	遊戱施設利用
ウォータースライダー	07	遊戱施設利用
お化け屋敷	08	遊戱施設利用
レストラン	09	飲食
ショップ	10	購入

② 時刻には、イベントが発生した時刻の時、分、秒を24時間表記で格納する。時系列で格納 するので、レコードは時刻の昇順になっている。

なお、この遊園地は10時に開園し、全ての入園者は20時になるまでに退園する。20時台

の時刻が格納されることはない。

- ③ 金額には、該当するイベントを利用した際の金額を格納する。 なお、入退園 (イベントコード 01, 02) の場合は、0 を格納する。
- ④ 一旦退園した入園者が、その日のうちに同じ入園者番号で再入園することはない。
- (3) 集計結果の表示例を図3に示す。

図3 集計結果の表示例

- ① TOTAL GUESTS には、入園者数を表示する。1日の入園者数が5,000人を超えることはない。
- ② STAY GUESTS には、左端に表示した時間帯の滞在者数を、年齢層別に棒グラフで表示する。 例えば、10 時ちょうどに入園し、11 時ちょうどに退園した者は、10 時台と 11 時台の滞在者として数える。
- ③ 棒グラフを構成する各記号は、**表2**に示すとおり滞在者の年齢層と対応している。記号一つは100人を表し、100人未満の端数は切り捨てる。

表 2	・ 年齢	層と対	여応ਰ	トる言	근문

年齢層	記号		
4~12歳	@		
13~18歳	*		
19~29歳	+		
30 ~ 49 歳	=		
50 歳以上	-		

(プログラム)

(行番号)

- 1 DATA DIVISION.
- 2 FILE SECTION.
- 3 FD GUEST.
- 4 01 G-REC.

5 02 G-NO PIC X(4).

6 02 G-AGE

PIC 9(1).

7 FD EVENT.

8 01 E-REC.

```
02 E-CODE PIC 9(4).
    02 E-NO
10
11
       88 E-ENTER
                        VALUE 1.
       88 E-EXIT
                        VALUE 2.
12
13
       88 E-FAC
                        VALUE 3 THRU 8.
14
       88 E-MEAL
                        VALUE 9.
15
       88 E-SHOP
                        VALUE 10.
    02 E-TIME.
16
                        PIC 9(2).
17
       03 E-HH
18
       03 E-MMSS
                       PIC 9(4).
                PIC 9(6).
   02 E-SALES
19
20 WORKING-STORAGE SECTION.
21 77 READ-FLAG
                 PIC X(1) VALUE SPACE.
      88 EOF
                    VALUE "E".
22
                  PIC 9(2) VALUE 10.
23 77 CR-HH
24 77 I
                  PIC 9(2).
25 77 J
                   PIC 9(2).
                 PIC 9(2).
26 77 MARK-CNT
27 77 MARK-POS
                   PIC 9(2).
28 77 TOTAL-GUEST PIC 9(4) VALUE 0.
29 77 TOTAL-GUEST-X REDEFINES TOTAL-GUEST PIC X(4).
30 01 STAY-GUEST.
31
    02 STAY-CNT PIC 9(4) OCCURS 5 VALUE ZERO.
32 01 PRT-GUEST.
    02 PRT-TIME
33
                     OCCURS 10.
       03 PRT-AGE PIC 9(4) OCCURS 5 VALUE ZERO.
35 01 HD-SCALE PIC X(50) VALUE
36
      "----+----1----+----5".
37 01 LHEADER.
               PIC 9(2).
38
     02 LHD-HH
     02
39
                  PIC X(7) VALUE ":00-59".
    02 LHD-MARK PIC X(50).
40
41 01 AGE-MARK.
    02 KIDS-M PIC X(50) VALUE ALL "@".
42
    02 STUDENT-M PIC X(50) VALUE ALL "*".
43
    02 YOUNG-M PIC X(50) VALUE ALL "+".
02 ADULT-M PIC X(50) VALUE ALL "=".
44
45
46
     02 OLD-M
                   PIC X(50) VALUE ALL "-".
47 01 REDEFINES AGE-MARK.
      02 MARK
48
                   PIC X(50) OCCURS 5.
49 PROCEDURE DIVISION.
50 MAIN-PROC.
51
      OPEN INPUT GUEST EVENT.
52
     PERFORM UNTIL EOF
53
        READ EVENT
54
           AT END
55
              SET EOF TO TRUE
            NOT AT END
56
              IF CR-HH NOT = E-HH THEN
```

```
58
                     PERFORM VARYING CR-HH FROM CR-HH BY 1
  59
                            UNTIL CR-HH = E-HH
  60
                                       а
                     END-PERFORM
  61
  62
                  END-IF
                  PERFORM SUM-PROC
  63
            END-READ
  64
  65
         END-PERFORM.
         CLOSE GUEST EVENT.
  66
  67
         PERFORM PRT-PROC.
         STOP RUN.
  68
  69 SUM-PROC.
  70
        MOVE E-NO TO G-NO.
  71
         READ GUEST.
        EVALUATE TRUE
  72.
  73
            WHEN E-ENTER
  74
               ADD 1 TO b
  75
            WHEN E-EXIT
  76
               SUBTRACT 1 FROM STAY-CNT(G-AGE)
  77
         END-EVALUATE.
  78 PRT-PROC.
  79
         DISPLAY "TOTAL GUESTS " TOTAL-GUEST-X.
  80
         DISPLAY "STAY GUESTS".
                     " HD-SCALE.
  81
         DISPLAY "
         PERFORM VARYING I FROM 1 BY 1 UNTIL I > 10
  82
  83
            MOVE 1 TO MARK-POS
  84
            COMPUTE LHD-HH = I + 9
  85
            MOVE SPACE TO LHD-MARK
           PERFORM VARYING J FROM 1 BY 1 UNTIL J > 5
  86
  87
               COMPUTE MARK-CNT = PRT-AGE(I, J) / 100
               IF MARK-CNT > 0 THEN
  88
  89
                  MOVE MARK(J)(1:MARK-CNT) TO LHD-MARK(MARK-POS:)
  90
  91
               END-IF
  92.
            END-PERFORM
  93
            DISPLAY LHEADER
  94
        END-PERFORM.
設問 1
        プログラム中のに入れる正しい答えを、解答群の中から選べ。
a に関する解答群
  MOVE STAY-CNT(CR-HH) TO PRT-AGE(CR-HH - 8, E-HH)

✓ MOVE STAY-CNT(CR-HH - 8) TO PRT-TIME(CR-HH)

  MOVE STAY-GUEST TO PRT-TIME(CR-HH - 8)
  MOVE ZERO TO PRT-AGE(CR-HH - 8, CR-HH)
```

b~dに関する解答群

- CR-HH
- 1 I
- ゥ J
- PRT-AGE(CR-HH 9, G-AGE)
- PRT-AGE(G-AGE, CR-HH 9)
- ⇒ STAY-CNT (CR-HH)
- # STAY-CNT(G-AGE)
- TOTAL-GUEST

e に関する解答群

- ADD J TO MARK-CNT
- ✓ ADD J TO MARK-POS
- D ADD MARK-CNT TO MARK-POS
- ADD MARK-POS TO MARK-CNT

各遊戯施設の延べ利用者数を集計し、図4に示すグラフを追加表示するように プログラムを変更した。表3中の に入れる正しい答えを、解答群の 中から選べ。

〔グラフの説明〕

- (1) 左端に表示したイベントコードに対応する遊戯施設を利用した延べ人数を表示する。記号# 一つは100人を表し、100人未満の端数は切り捨てる。一つの遊戯施設の延べ利用者数が 5.000 人を超えることはない。
- (2) 延べ利用者数が100人未満の遊戯施設の行は表示しない。

```
RECREATIONAL FACILITIES
   ----+----1-----2------3------4------5
03: ###############
04: ######
08: #########
```

図 4 遊戯施設利用状況の表示例

表3 プログラムの変更

処置	変更内容		
行番号48と49の間	01 .		
に追加	02 FAC-CNT PIC 9(4) OCCURS 6 VALUE ZERO.		
	01 FAC-HEADER.		
	02 FAC-NO PIC 9(2).		
	02 PIC X(2) VALUE ": ".		
	01 FAC-MARK PIC X(50) VALUE ALL "#".		
行番号76と77の間	WHEN E-FAC		
に追加	f		
行番号 94 の後ろに	DISPLAY "RECREATIONAL FACILITIES".		
追加	DISPLAY " " HD-SCALE.		
	PERFORM g		
	COMPUTE MARK-CNT = FAC-CNT(FAC-NO - 2) / 100		
	IF MARK-CNT NOT = 0 THEN		
	DISPLAY FAC-HEADER FAC-MARK(1:MARK-CNT)		
	END-IF		
	END-PERFORM.		

fに関する解答群

- ADD 1 TO FAC-CNT(E-CODE)
- ADD 1 TO FAC-CNT(E-CODE 2)
- ADD G-AGE TO FAC-CNT(E-CODE)
- ADD G-NO TO FAC-CNT (E-CODE)
- ★ ADD STAY-CNT(G-AGE) TO FAC-CNT(E-CODE)

gに関する解答群

- 7 6 TIMES
- UNTIL FAC-NO > 6 AND MARK-CNT > 50
- UNTIL FAC-NO > 6 OR MARK-CNT > 50
- VARYING FAC-NO FROM 1 BY 1 UNTIL FAC-NO > 6
- ▼ VARYING FAC-NO FROM 3 BY 1 UNTIL FAC-NO > 8

間10 気 午後のカギ

COBOL プログラムによる集計処理の問題です。文法的に難しいところはありませんが、集計結果を 格納する二次元の表の使い方などは、基本知識として理解しておきましょう。

設問 1 空欄 a は、時間帯でとの入園者数を集計するのに必要な処理です。空欄 b \sim d を先に考えた ほうがわかりやすいでしょう。

設問 2 遊戯施設ごとの利用者数は、新たに追加された FAC-CNT(1) ~ FAC-CNT(6) に集計します。

設問 1 プログラムの完成

空欄 a:空欄の前後は、次のとおりです。

57 IF CR-HH NOT = E-HH THEN

58 PERFORM VARYING CR-HH FROM CR-HH
BY 1

В

59 UNTIL CR-HH = E-HH
60 a

61 END-PERFORM

62 END-TF

63 PERFORM SUM-PROC

このブロックは、イベントファイルを先頭から1レコードずつ読むたびに実行されます。CR-HHは、集計対象となる時間帯(初期値10)で、E-HHはイベントファイルから読み込んだレコードに記録されている時間です。

行番号 57 の IF 文の条件が真になるのは、読み込んだレコードの記録時間が、集計対象の時間帯と異なっていた場合です。レコードは記録時間の昇順に並んでいるので、その場合は集計対象の時間帯を切り替えなければなりません。空欄 a には、その際に必要な処理が入ります。

空欄 a を解答するには、読み込んだレコードがどのように集計されるかを理解しておく必要があります。 WORKING-STORAGE 節で定義されている集計用の項目には、以下のものがあります。

① TOTAL-GUEST:1日の総入園者数を格納

TOTAL-GUEST

② STAY-GUEST: 園内に滞在する入園者数を年齢層 別に格納

STAY-GUEST STAY-CNT(1) STAY-CNT(2) ··· STAY-CNT(5)

③ PRT-GUEST: 時間帯ごとの入園者数を年齢層別に 格納

PRT-GUEST

	<u>'</u>		, ·
PRT-TIME(1)	PRT-AGE(1,1)	PRT-AGE(1,2)	 PRT-AGE(1,5)
PRT-TIME(2)	PRT-AGE(2,1)	PRT-AGE(2,2)	 PRT-AGE(2,5)
:	:	:	:
PRT-TIME(10)	PRT-AGE (10, 1)	PRT-AGE (10, 2)	 PRT-AGE (10,5)

② STAY-GUEST の値は入園者数一退園者数で、その時点で園内に滞在している入園者の数になります。

また、③ PRT-GUEST は、各時間帯に入園した入園者数ではなく、その時間帯に園内に滞在した入園者数(前の時間帯に入園した人を含む)であることに注意しましょう。

イベントファイルから読み込んだレコードのイベントコードが01(入園)だった場合には、これらの項目の該当する部分に、それぞれ1が加算されます(行番号74)。

空欄 a は、空欄 b \sim d を先に埋めたほうがわかりやすいかも。



PRT-GUEST は二次元の表で、時間帯ごとに PRT-TIME (1) ~ PRT-TIME (10) に分かれているため、集計対象の時間帯を変更した場合は、集計先の PRT-TIME も変更します。たとえば、集計対象を 10 時台から 11 時台に変更すると、集計先は PRT-TIME (1)から PRT-TIME (2)に変わります。先ほど説明したように、PRT-GUEST はその時間帯に園内に滞在している入園者数なので、その時点で園内に滞在している入園者数を、11 時台に繰り入れる必要があります。

PRT-TIME(2) 10時台から引き続き 11時台に入園した 滞在している入園者 入園者

空欄 a で実行するのはこの処理です。前の時間帯 (CR-HH) から引き続き滞在している入園者の数は、STAY-GUESTに格納されています。また、次の時間 帯の入園者数は PRT-TIME (CR-HH - 8) に格納するので、空欄 a の処理は次のようになります。

MOVE STAY-GUEST TO PRT-TIME(CR-HH - 8)

以上から,**空欄 a** は <mark>ウ</mark> です。

空欄 $b \sim d$: 行番号 74 は、先に説明した 3 つの集計 先に、それぞれ 1 を加算する処理です。

74 ADD 1 TO b c d

- ① TOTAL-GUEST (1日の総入園者数) に1を加算します。
- ② STAY-GUEST 内の該当する年齢層に 1 を加算します。 STAY-GUEST は一次元の表で、各要素は STAY-CNT という名前で定義されています。また、

年齢層コードは入園者ファイルのレコードから G-AGE として読み込まれているので、加算先は STAY-CNT(G-AGE) となります。

 ③ PRT-GUEST の該当する時間帯,年齢層の要素に1 を加算します。PRT-GUEST は二次元の表で、各要素は PRT-AGE という名前で定義されています。また,時間帯に該当する行は CR-HH - 9,年齢層コードは G-AGE なので,加算先は PRT-AGE(CR-HH - 9, G-AGE) となります。

ADD 1 TO TOTAL-GUEST

STAY-CNT(G-AGE)

PRT-AGE(CR-HH - 9, G-AGE)

以上から,**空欄 b \sim d** には $oldsymbol{ I I }$, $oldsymbol{ b }$ が順不同で入ります。

空欄 e:空欄の前後は次のとおりです。

85 MOVE SPACE TO LHD-MARK

86 PERFORM VARYING J FROM 1 BY 1

UNTIL J > 5

87 COMPUTE MARK-CNT = PRT-AGE(I, J)

/ 100

88 IF MARK-CNT > 0 THEN

MOVE MARK(J)(1:MARK-CNT) TO

LHD-MARK (MARK-POS:)

e

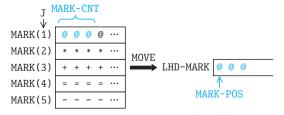
92 END-PERFORM

LHD-MARKには、ある時間帯の入園者数を表す年齢層別の棒グラフが入ります。

行番号87で、年齢層ごとの記号の数を算出し、MARK-CNTに格納します。次に、

MOVE MARK(J)(1:MARK-CNT) TO LHD-MARK(MARK-POS:)

で、二次元の表 MARK にある該当する記号列を MARK-CNT 個分切り取り、LHD-MARK に転記します。



LHD-MARK に転記する位置は、MARK-POS に設定

さています。1つの年齢層の記号を転記したら、次の年齢層の記号を追加するために、MARK-POSの値を更新します。空欄 e に入るのはこの処理です。

次のMARK-POSの値は、現在のMARK-POSにMARK-CNTの値を加算したものになるので、次のようになります。

ADD MARK-CNT TO MARK-POS

以上から、**空欄 e** は **ウ** です。

設問2 プログラムの変更

空欄 f:遊戯施設の延べ利用者数も集計するように、 プログラムを変更します。空欄の前後は次のようにな ります。

EVALUATE TRUE
:
WHEN E-FAC
f
END-EVALUATE.

遊戯施設利用を表すイベントコードは、3~8です。 空欄 f には、イベントファイルから読み込んだレコードのイベントコード (E-CODE) が、この範囲内の数値 (E-FAC) だった場合に実行する処理が入ります。

遊戯施設は 6 種類あり、それぞれの利用者数は新たに追加された FAC-CNT(1) \sim FAC-CNT(6) に集計します。したがって、空欄 \mathbf{f} に入る処理は、

ADD 1 TO FAC-CNT(E-CODE - 2)

となります。**空欄 f** は **イ** です。

空欄 g:空欄の前後は次のとおりです。

PERFORM 9

COMPUTE MARK-CNT = FAC-CNT(FAC-NO - 2)/100

IF MARK-CNT NOT = 0 THEN

DISPLAY FAC-HEADER

FAC-MARK (1: MARK-CNT)

END-IF
END-PERFORM.

集計した遊戯施設の利用者数を、棒グラフで表示します。6種類の遊戯施設ごとに、PERFORM文で繰り

返します。棒グラフを構成する"#"記号の個数を、

FAC-CNT(FAC-NO - 2) / 100

で計算しているので、FAC-NO は $3 \sim 8$ の遊戯施設の コードを表すことがわかります。FAC-NO の初期値は 3 で、8 を超えるまで繰返しごとに 1 加算するので、2 空欄 2 は次のようになります。

PERFORM VARYING FAC-NO FROM 3 BY 1 UNTIL FAC-NO > 8 以上から、**空欄 q** は オ です。

解答

設問 1 a - ウ,

b, c, d — 工, 丰, 夕 (順不同),

e — 💆

設問2 f - 1, g - 1

11

次の Java プログラムの説明及びプログラムを読んで,設問 1,2 に答えよ。 (Java プログラムで使用する API の説明は,この冊子の末尾を参照してくださ

い。)

(プログラムの説明)

試験の成績を管理するプログラムである。

- (1) クラス ScoreManager は、後述するクラス ValueSortedMap を継承したクラスで、試験の成績を管理する。点数の高い順に学籍番号(英数字で構成される固定長の文字列)と点数を出力するメソッド print をもつ。
- (2) クラス ValueSortedMap は、キーと値の対応付けを値の昇順、又はコンストラクタで指定したコンパレータに従った順に保持する。キーと値の対応付けをキーの昇順、又はコンパレータに従った順に保持するクラス TreeMap を利用している。主なメソッドは次のとおりである。public V put(K key, V value)

keyにvalueを対応付けて登録する。key又はvalueがnullのときは例外NullPointerExceptionを投げる。keyが既に他の値と対応付けられていれば、その値をvalueで置き換え、置き換えられる前の値を返す。keyに値が対応付けられていなければ、nullを返す。

public V get(K key)

key に対応付けられた値を返す。key と値の対応付けがなければ, null を返す。

public V remove(K key)

keyと値の対応付けを削除し、対応付けられていた値を返す。keyと値の対応付けがなければ、nullを返す。

public Iterator(K> iterator()

対応付けられた値の昇順,又はコンストラクタで指定したコンパレータに従った順に, キーを返すための反復子を返す。

フィールド map は、キーと値の対応付けを保持する。

フィールド reverseMap は、値にキーを対応付けて保持する。異なるキーに同じ値が対応付けられることがあるので、値に対応付けられるのはキーのリストである。

(3) クラス ScoreManagerTester は、成績管理プログラムのテストプログラムである。メソッド main の実行結果を、図 1 に示す。

```
数学の成績
FE0002:90点
FE0005:90点
FE0003:85点
FE0001:85点
null
```

図1 メソッド main の実行結果

〔プログラム 1〕

```
import java.util.Comparator;
public class ScoreManager extends ValueSortedMap< a > {
   private final String subject;
   public ScoreManager(String subject) {
      super(new Comparator<Integer>() {
            public int compare(Integer a, Integer b) {
               return b.compareTo(a);
            }
         });
      this.subject = subject;
   }
   public void print() {
      System.out.println(subject + "の成績");
      for (String name : this) {
         System.out.printf(" %s: %d点%n", name, get(name));
      }
   }
}
```

〔プログラム2〕

```
import java.util.ArrayList;
import java.util.Comparator;
import java.util.HashMap;
import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.NoSuchElementException;
import java.util.TreeMap;

public class ValueSortedMap<K, V> implements Iterable<K> {
    Map<K, V> map = new HashMap<K, V>();
    Map<V, List<K>> reverseMap;
```

```
public ValueSortedMap() {
   reverseMap = new TreeMap<V, List<K>>();
public ValueSortedMap(Comparator<? super V> c) {
   reverseMap = new TreeMap<V, List<K>>(c);
public V put(K key, V value) {
   if (| b |) {
      throw new NullPointerException();
   V old = remove(key);
   map.put(key, value);
   List<K> keys = reverseMap.get(value);
   if ( c ) {
      keys = new ArrayList<K>();
     reverseMap.put(value, keys);
   keys.add(key);
   return old:
public V get(K key) {
   return map.get(key);
public V remove(K key) {
   V value = d (key);
   if (value != null) {
      List<K> keys = reverseMap.get(value);
      keys.remove(key);
      if (keys.isEmpty()) {
            e (value);
      }
  return value;
public int size() {
   return map.size();
public Iterator(K> iterator() {
   return new Iterator<K>() {
      Iterator<V> vi = reverseMap.keySet().iterator();
      Iterator<K> ki = new ArrayList<K>().iterator();
      public boolean hasNext() {
         return vi.hasNext() || ki.hasNext();
      }
```

```
public K next() {
            if (hasNext()) {
              if (!ki.hasNext()) {
                 ki = reverseMap.get(vi.next()).iterator();
              return ki.next();
            }
            throw new NoSuchElementException();
         }
         public void remove() {
            throw new UnsupportedOperationException();
      };
   }
}
(プログラム3)
public class ScoreManagerTester {
   public static void main(String[] args) {
      ScoreManager sm = new ScoreManager("数学");
      try {
         sm.put("FE0001", 70);
         sm.put("FE0002", 90);
         sm.put("FE0003", 85);
         sm.put("FE0004", 95);
         sm.put("FE0005", 90);
         sm.put("FE0001", 85);
         sm.remove("FE0004");
         sm.put(null, 90);
         sm.put("FE0004", 90);
      } catch (NullPointerException e) {}
      sm.print();
      System.out.println(sm.get("FE0004"));
   }
}
                             に入れる正しい答えを、解答群の中から選べ。
設問
         プログラム中の
a に関する解答群
                                         Integer, String
     Integer
                                          String, Integer
      String
b に関する解答群
  🗾 key != null && value != null 🔀 key != null 📙 value != null
```

- b key == null && value == null
- key == null || value == null

- cに関する解答群
 - !keys.isEmpty()
 - bevs == null

- ✓ keys != null
- keys.isEmpty()

- d. eに関する解答群
 - Map.get
 - Map.remove
 - remove
 - reverseMap.remove

- ✓ map.get
- map.remove
- πeverseMap.get

設問 2 点数が同じ場合には、学籍番号の文字列としての自然順序付けに従って出力するように変更する。クラス ValueSortedMap で使用しているクラスやインタフェースの変更だけで実現する場合、変更内容として適切なものを、解答群の中から選べ。

解答群

- ArrayList を TreeSet に変更する。
- ✓ List と ArrayList を Set に変更する。
- ウ List を Set に変更し、ArrayList を TreeSet に変更する。
- List を TreeSet に変更し、ArrayList を Set に変更する。
- オ List を TreeSet に変更する。

間11 気 午後のカギ

コレクションクラスを使った Java プログラムを題材にした問題です。Map や List などのコレクションクラスの基本的な使い方については、ジェネリックスの概念も含めてマスターしておきましょう。 なお、問題文中のコレクションクラスについては、巻末の「Java プログラムで使用する API の説明」 に解説があります(試験本番では、問題冊子の巻末に掲載されます)。

設問 1 reverseMap がデータをどのように登録しているかを考えます。

設問 2 ArrayList と TreeSet は、インタフェースが異なる実装です。

設問 1 プログラムの完成

空欄 a:空欄は、クラス ScoreManager の定義の一部です。

クラスScoreManagerは、クラスValueSortedMap

を継承しています。クラス ValueSortedMap は、キーと値の対応付けを整列して保持するクラスです。

public class ValueSortedMap<K, V>implements Iterable<K> { ープログラム2

キーと値それぞれのデータ型は、後で決めることが できるように、ジェネリックスという機能を使って定 義されています。キーの型が K、値の型が V になりま す。これらは,後で実際のデータ型に置き換えます。 この置き換えを行っているのが,上記のクラス ScoreManagerの定義部分です。

クラス ScoreManager は、学籍番号と点数の組を、点数の高い順に保持するクラスで、学籍番号がキー、点数が値に当たります。学籍番号は文字列データ、点数は整数データなので、次のようになります。

public class ScoreManager

```
extends ValueSortedMap<String, <u>Integer</u>> {
 キー(学籍番号) 値(点数)の
 のデータ型 データ型
```

以上から,**空欄**aは<u>エ</u>です。

空欄 b:空欄の前後は次のとおりです。

```
if ( b ) {
   throw new NullPointerException();
}
```

空欄は、クラス ValueSortedMap のメソッド put の一部で、例外 NullPointerException を 投げるときの条件が入ります。

問題文(2)のメソッド put の説明には,

key 又は value が null のときは例外 NullPointerException を投げる。

とあります。したがって**空欄 b** には,「key 又は value が null」を表す条件式

```
key == null || value == null
```

が入ります。空欄bはエです。

空欄 c:引き続き、クラスValueSortedMapのメソッド put の一部です。

```
List<K> keys = reverseMap.get(value);
if ( c ) {
   keys = new ArrayList<K>();
   reverseMap.put(value, keys);
}
keys.add(key);
```

クラス Value Sorted Mapは、map、reverse Mapという 2 つのフィールドをもっています。 mapには keyをキー、valueを値とする組を登録し、revers Mapには valueをキー、keyを値とする組

を登録します。ただし、同じ value に対応する key は複数あると考えられるので、reverseMap の値は List 型にします。



プログラムの上の部分は、まず、

keys = reverseMap.get(value);

で、reverseMapから登録する value に対応する keys を検索します。対応する keys がない場合、新しい value と keys の組を reverseMap に追加します。対応する keys がすでに登録されている場合は、その末尾に新しい key を追加します。

```
・対応するkeysがない場合
                              ・対応するkeysが存在する場合
    (FE0005, 40)
                               (FE0005, 10)
                                     keys
        value
                            value
                 kevs
                              30 FE0002
         30 FE0002
         20 FE0001, FE0003
                              20 FE0001, FE0003
                                                 既存の
         10 FE0004
                              10 FE0004, FE000
                                                 keys
に追加
新しい組
         40 FE0005
を追加
```

空欄 c は,「対応する keys がない場合」を意味する条件式です。

get メソッドは、対応する値が登録されていない場合は null を返すので、空欄 c には

```
kevs == null
```

が入ります。空欄 c は ウ です。

空欄 d,e:空欄は,クラスValueSortedMapのメソッド removeの一部です。

メソッド remove は、指定された key のデータを 削除します。 データはフィールド map に登録されています。そこでまず、指定した key のデータを map のメソッド remove で削除します。空欄 d は次のようになります。

V value = map.remove(key);

以上から,**空欄 d** は エ です。

続いて、フィールド reverseMap からもデータを 削除します。そのためには、value に対応する keys を検索し、見つかった keys から key を削除します。

keys.remove(key);

この結果、keysの中身が空になった場合(keys.isEmpty()が真)には、対応するvalueもreverseMapから削除してよいので、

reverseMap.remove(value);

を実行します。**空欄 e** は **‡** です。

設問2 プログラムの変更

現在のValueSortedMapは、reverseMapの値をArrayList型で実装しています。ArrayListは配列を使った単純なリストなので、データは登録した順に登録され、順序付けされていません。

クラス ScoreManager もこれを継承するため、点数が同じ学籍番号は、reverseMap に順序付けされずに登録されます。



ArrayList を順序付けされたコレクションクラスに変更すれば、この問題は解決します。Java APIには、TreeSet という順序付けされたコレクションクラスが用意されているので、これを使うのがよいでしょう。

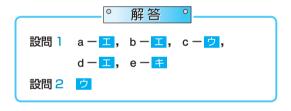
ただし、ArrayList がインタフェース List の実装なのに対し、TreeSet はインタフェース Set の実装です。そのため、reverseMap の定義も、従来の

Map<V, List<K>> reverseMap;

から,

Map<V, Set<K>> reverseMap;

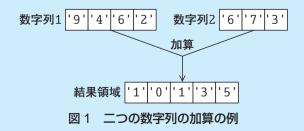
に変更する必要があります。以上から、設問2は 「ListをSetに変更し、ArrayListをTreeSet に変更する。」が正解です。



12 次のアセンブラプログラムの説明及びプログラムを読んで、設問 1 ~ 3 に答えよ。

〔プログラム 1 の説明〕

主プログラムから渡された二つの数字列(数字列 1 と数字列 2)をそれぞれ正の 10 進数とみなして加算し、結果を数字列で返す副プログラム ADDC である。二つの数字列の加算の例を**図 1** に示す。



(1) 汎用レジスタ GR1 ~ GR5 には、それぞれ次の内容が設定されて、主プログラムから渡される。

GR1:数字列1の先頭アドレス GR2:数字列2の先頭アドレス GR3:結果領域の先頭アドレス GR4:数字列1の長さ(文字数)

GR5:数字列2の長さ(文字数)

- (2) 数字は1語に1文字格納する。
- (3) 結果の数字列は結果領域に格納し、長さ(文字数)は GRO に設定して、主プログラムに返す。
- (4) 数字列1と数字列2の長さ(文字数)は、それぞれ1以上とする。
- (5) 数字列1と数字列2の左端の文字は"0"でないものとする。
- (6) 副プログラムから戻るとき,汎用レジスタ GR1~ GR7 の内容は元に戻す。

〔プログラム 1〕

```
(行番号)
   ADDC
            START
1
            RPUSH
2
 3
            CPA
                    GR4, GR5
            JPI.
                    CONT
 4
 5
            PUSH
                    0. GR4
 6
            LD
                    GR4, GR5
                                      GR4 ≥ GR5 となるように
 7
            P0P
                    GR5
                                      数字列1と数字列2のポインタ
 8
            PUSH
                    0, GR1
                                       を入替え
9
                    GR1, GR2
            T.D
10
            P<sub>0</sub>P
                    GR2
                    GR1, GR4
11
    CONT
            ADDA
                    GR2, GR5
12
            ADDA
                    GR0, =0
                                   ; 桁上げフラグを初期化
            LD
13
                                   ; スタックデータの終わりの印
14
            PUSH
                    0
15
   L00P1
            LAD
                    GR2, -1, GR2
16
            LD
                    GR6, 0, GR2
                    GR6, =#000F
17
            AND
   LOOP2
            LAD
                    GR1, -1, GR1
18
19
            ADDA
                    GR6, 0, GR1
                                   ; 1 桁加算
20
            ADDA
                    GR6, GR0
                                   ; 桁上げフラグを加算
                    GR6, = '9'
21
            CPA
                                   : 桁上げ?
            JPL
                    CARRY
22
23
            LD
                    GR0, =0
                                   ; 桁上げフラグをクリア
24
            JUMP
                    NEXT
25
   CARRY
            LD
                    GR0, =1
                                   ; 桁上げフラグを設定
26
                     а
27
    NEXT
            PUSH
                    0, GR6
                                   ; 加算結果1桁を保存
                     b
28
29
             JZE
                    END1
30
                      С
31
            JPL
                    L00P1
32
            LD
                    GR6.=0
33
            JUMP
                    LOOP2
```

```
END1
34
         LD
               GRO, GRO
                          ; 左端を桁上げ?
35
         JZE
               NOCARRY
               GR1, = '1'
36
         LD
                          ; 左端に '1' を追加
   L00P3
               GR1, 0, GR3
37
         ST
               GR3.1.GR3
38
         LAD
   NOCARRY POP
                          ; 加算結果1桁取出し
39
               GR1
               GR1, GR1
40
         LD
41
         JZE
               END2
42
         ADDA
               GR0.=1
43
         JUMP
               L00P3
44
   END2
         RPOP
45
         RET
46
         END
                          に入れる正しい答えを、解答群の中から選べ。
       プログラム 1 中の
解答群
  SUBA GR4, = 1
                          SUBA GR4, =9
                                               SUBA GR4, =10
  T
     SUBA
          GR5, =1
                          SUBA
                               GR5, =9
                                               SUBA
                                                    GR5, =10
     SUBA
          GR6, =1
                          SUBA
                               GR6, =9
                                               SUBA
                                                    GR6, =10
  丰
                                            ケ
                       に入れる正しい答えを、解答群の中から選べ。
       次の記述中の
 図1の数字列1と数字列2を引数として副プログラム ADDC を実行した場合, 行番号25のLD
命令は d 回実行される。
解答群
  ア 0
             1
                         ウ 2
       副プログラム ADDC を使用して, n 個(n≥1)の数字列を入力し,総和を求め
       て数字列で出力するプログラム SUMC を作成した。n = 5 の場合の例を図2に
示す。
 プログラム 2 中の
                    に入れる正しい答えを、解答群の中から選べ。
 なお、入力される数字列、総和は256桁以内に収まるものとする。
                      数字列1 '9' '1' '3' '4'
                             8' 7'
                      数字列2 '9'
                      数字列3 '5'
                             '7
                      数字列4 '2'
                      数字列5 '3'
                             6
                       総和
                           11
                             |'0'|'2'|'1'|'6'
                    図2 数字列1~5の総和の例
```

(プログラム2)

SUMC START LAD GR1, BUF1 ; 数字列1のポインタを設定 ΤN BUF1, LEN1 ; 最初の数字列を入力し数字列1とする : 数字列1の長さを設定 T.D GR4, LEN1 ; BUF3 を総和の領域として設定 LAD GR3, BUF3 GR2, BUF2 ; 数字列2のポインタを設定 LAD LOOP IN BUF2, LEN2 ; 次の数字列を入力し数字列2とする T.D GR5, LEN2 ; 数字列2の長さを設定 TMT FIN1 ; 入力終了 CALL ADDC е **PUSH** 0, GR1 総和の領域のポインタと LD GR1, GR3 数字列1のポインタを入替え P0P GR3 JUMP LOOP FIN1 f ; 総和の領域を判定 CPA GR1, GR3 JZE FIN2 ST GR4, LEN1 BUF1, LEN1 OUT JUMP FIN3 ST FIN2 GR4, LEN3 OUT BUF3, LEN3 FIN3 RET 256 BUF1 DS ; 数字列1又は総和 BUF2 DS 256 ; 数字列 2 BUF3 256 DS ; 総和又は数字列1 LEN1 DS 1 LEN2 DS 1 LEN3 DS 1 END

解答群

 7
 LAD
 GR3, BUF1
 ✓
 LAD
 GR3, BUF2

 □
 LAD
 GR3, BUF3
 □
 LD
 GR3, GR0

 □
 LD
 GR4, GR0
 □
 LD
 GR5, GR0



12 年後のカギ

数字列同士の加算を行うプログラムです。桁上がりを考慮しながら数字を 1 桁ずつ加算していく手順は、小学校の算数で習いますが、この手順をプログラムで表現します。

設問 1 空欄 a は桁上がりが生じたときの処理、空欄 b, c は、残りの桁を数える処理です。

設問 2 桁上がりが生じる回数を数えます。

設問3 空欄eは,数字列1の長さをADDCに渡す処理ですが,ADDCの仕様がヒントになります。空

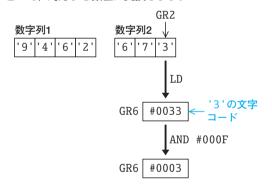
欄 f は、GR1 が指す領域が、BUF1 か BUF3 かを判別する処理です。

設問 **1** プログラム 1 の完成

空欄 a: プログラム1の動きを追ってみましょう。

15 LOOP1 LAD GR2,-1,GR2 16 LD GR6,0,GR2 17 AND GR6,=#000F

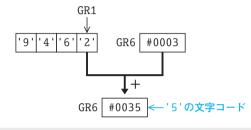
プログラムは、数字列 2 の加算する桁のアドレスを GR2 に設定し、その桁の数字(文字コード)を GR6 に転送します。数字 '0' \sim '9' に対応する文字コードは $\# 0030 \sim \# 0039$ なので、# 000F との論理積を とって、対応する数値に変換します。



18 LOOP2 LAD GR1,-1,GR1 19 ADDA GR6,0,GR1 20 ADDA GR6,GR0

続いて、数字列1の加算する桁のアドレスをGR1に設定し、その桁の数字をGR6の数値に加算します。すると、2つの数字列の1桁分を加算した結果が、文字コードでGR6に格納されます。

また、ひとつ前の桁で生じた桁上げフラグ(GRO)も加算します。



加算結果で桁上がりが生じると, GR6 の内容が '9'の文字コードを超えてしまいます。その場合は桁上げの処理(ラベル CARRY)に分岐します。

桁上げが生じない場合は、GR0 を 0 にして、次の処理に進みます。

25 CARRY LD GR0,=1
26 a

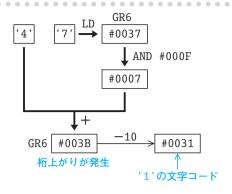
加算結果で桁上がりが生じた場合の処理を行います。行番号 25 で,GR0 に 1 を設定します。GR0 の内容は,次の桁同士を加算する際にいっしょに加算されます。

次に、GR6の文字コードを修正します。たとえば '4' + '7' を処理すると、GR6の内容は '4'の文字 コード #0034 に 7 を加えた #003B になっています。

これを正しい文字コードに修正するには、GR6から10を引きます。#003Bの場合は、10を引くと#0031になり、'1'の文字コードになります。空欄aには、この処理を実行する次のような命令が入ります。

SUBA GR6, =10

以上から、**空欄 a** は ケ です。



空欄 b, c:空欄の前後は次のとおりです。

27 NEXT	PUSH	0,GR6	
28		b	
29	JZE	END1	
30		С]
31	JPL	L00P1	
32	LD	GR6,=0	
33	JUMP	LOOP2	

加算結果 1 桁をスタックに保存したら,まだ加算する桁があるかどうかを調べます。数字列 1 と数字列 2 の桁数は,それぞれ GR4 と GR5 に格納されています。数字列 1 のほうが桁数が多いので,数字列 1 に加算する桁が残っていれば,加算処理を続行します。

空欄bでは、GR4の値を1減らします。値が0なら、これ以上加算する桁がないので、ラベルEND1に分岐します。

空欄bの内容は次のようになります。

SUBA GR4, =1

続いて、数字列 2 に加算する桁が残っているかどうかを調べるため、GR5 の値も1 減らします。値が0でなければ L00P1 に分岐し、0 の場合は GR6 に0 をセットして、L00P2 に分岐します。

空欄cの内容は次のようになります。

SUBA GR5, =1

以上から、空欄 b は r 、空欄 c は r です。

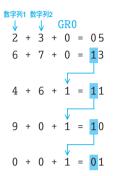
設問2 桁上がりの発生回数

空欄 d: 各桁同士の加算は、簡単に言えば

数字列1+数字列2+桁上げフラグ

で行われます(行番号 $19 \sim 20$)。行番号 25 は,各桁同士の加算で桁上がりが生じたとき,桁上げフラグ (GR0) に 1 を設定する処理です。桁上げフラグの内容は,次の桁の加算に使われます。

図1の例では、次のように3回の桁上がりが発生します。



以上から, 行番号 25 は 3 回実行されます。**空欄 d** は <mark>エ</mark>です。

設問3 プログラム2の完成

空欄 e: プログラム SUMC は、次のような手順で入力 した数字列の総和を求めます。

- ①入力装置から BUF1 に入力した数字列を,数字列 1 とする。
- ②入力装置から BUF2 に入力した数字列を,数字列 2 とする。
- ③副プログラム ADDC を呼び出す。
- ④③の結果の数字列(BUF3 または BUF1 に格納)を,新しい数字列1とする。
- ⑤②~④を,入力する数字列が終わるまで繰り返す。

上記の手順④で、ADDC の結果を数字列 1 に設定しますが、このとき、数字列 1 の長さ(文字数)を GR4 に設定する処理が必要です。 空欄 e には、この処理を実行する命令が入ります。

【プログラム 1 の説明】の(3) によれば、結果の数字列の長さは「GRO に設定して、主プログラムに返す」とあります。したがって、GRO の内容を GR4 に転記すればよいことがわかります。

LD GR4, GR0

以上から,**空欄e**は オです。

空欄 f:空欄の前後は次のとおりです。

FIN1		f	
	CPA	GR1, GR3	←総和は BUF3?
	JZE	FIN2	←YESならFIN2へ
	ST	GR4, LEN1	
	OUT	BUF1, LEN1	← BUF1を出力
	JUMP	FIN3	
FIN2	ST	GR3, LEN3	
	OUT	BUF3, LEN3	← BUF3 を出力
FIN3	RET		

すべての数字列の加算が終わると、プログラムは入力された数字列の総和を、OUT 命令で出力します。OUT 命令は、

OUT 出力領域,出力文字長領域

のように指定し、文字列データを出力装置に出力します。

入力された数字列の総和は、BUF1 または BUF3 に格納されています。そのアドレスは GR1 に格納されていますが、BUF1 のアドレスか、BUF3 のアドレスかはわかりません。OUT 命令は出力領域をラベルで指定しなければならないので、総和が格納されているの

が BUF1 か BUF3 かを判別する必要があります。 そこで、GR3 に BUF3 のアドレスを入れ、

CPA GR1, GR3

を実行して、GR1 の内容と比較します。もし、GR1 = GR3 なら、GR1 は BUF3 を指しているので、BUF3 を 出力します。さもなければ、GR1 は BUF1 を指していると判断できるので、BUF1 を出力します。

以上から、**空欄 f** は、GR3 に BUF3 のアドレスを転記する命令が入ります。

LAD GR3, BUF3

空欄 f は ウ です。



$oxed{ extit{B}}$ $oxed{1}$ 次の表計算,ワークシート及びマクロの説明を読んで,設問 1 \sim 3 に答えよ。

〔表計算の説明〕

表計算ソフトを用いて、ある大学学部所有の図書を管理するプロトタイプシステムを作成した。 プロトタイプシステムは"図書情報"と"利用者情報"の二つのワークシートから成る。

〔ワークシート:図書情報〕

学部で所有している図書のうち200冊を抽出し、図1のワークシート"図書情報"を作成した。

	Α	В	С	D	E	F	G	Н	I
1	図書ID	図書名	利用者ID	貸出日	返却期日	延滞日数		日付	11-06-15
2	1	プログラミング入門							
3	2	ネットワーク	4	11-05-08	11-06-05	10			
4	3	情報科学入門	4	11-05-08	11-06-05	10			
5	4	データベース							
6	5	統計入門	37	11-06-13	11-06-20	0			
7	6	情報セキュリティ	2	11-05-28	11-06-25	0			
:	:	:	:	:	:	:			
201	200	論理学応用	2	11-05-28	11-06-25	0			
202			•						
203	図書ID	170	利用者ID	40					

注記 行203については設問2で説明する。

図 1 ワークシート "図書情報"

- (1) セル B2 ~ B201 には、該当する図書名を入力する。セル A2 ~ A201 には、1 から始まる連番で図書 ID を入力する。
- (2) セル $C2 \sim C201$ には、図書が貸出し中であれば、借りた利用者の利用者 ID を入力する。そうでなければ空値である。
- (3) セル $D2 \sim D201$ には、図書が貸出し中であれば、貸し出した日付を入力する。そうでなければ空値である。
- (4) セル E2 ~ E201 には、返却期日を算出して格納する。貸出し中でなければ空値である。
- (5) セル $F2 \sim F201$ には、返却期日を過ぎても返却されていない図書に関して、延滞日数を求める式を入力する。この式は、返却期日を過ぎていなければ0に、貸出し中でなければ空値になる。
- (6) セル I1 には、本日の日付を求める式を入力する。
- (7) 日付の表記は yy-mm-dd で表示されるが、表計算ソフトの内部では 1970 年 1 月 1 日からの経 過日数を整数値で保持している。計算にはこの値を利用する。

〔ワークシート: 利用者情報〕

利用対象者を 50 名抽出し、利用者ごとの情報から**図 2** に示すワークシート "利用者情報 "を作成した。

	Α	В	С	D	Е	F	G	Н	I
1	利用者ID	氏名	属性	残り 貸出冊数	延滞状態		属性	貸出 上限冊数	貸出日数
2	1	情報一郎	学部生	4	*		教員	20	28
3	2	情報二郎	教員	8			大学院生	10	28
4	3	情報三郎	大学院生	10			学部生	5	14
5	4	情報四郎	教員	17	*		その他	3	7
:	:	:	:	:	:				
51	50	情報花子	その他	3					

図2 ワークシート"利用者情報"

- (1) セル B2 ~ B51 には、氏名を入力する。セル A2 ~ A51 には、1 から始まる連番で利用者 ID を入力する。
- (2) セル C2 ~ C51 には、属性(教員、大学院生、学部生、その他)を入力する。
- (3) セル $D2 \sim D51$ には、貸出上限までの現在の残り冊数(以下、残り貸出冊数という)を求める式を入力する。
- (4) セル $E2 \sim E51$ には、延滞状態の図書が1 冊でもあれば "*" に、延滞状態の図書が無いときは空値になる式を入力する。
- (5) セル $G2 \sim G5$ には属性の名称を、セル $H2 \sim H5$ とセル $I2 \sim I5$ には、それぞれの属性に応じた貸出上限冊数と貸出日数を入力する。

設問]	ワークシート"図書情報"及び"利用者情報"に関する次の記述中の	
	に入れる正しい答えを、解答群の中から選べ。	

返却期日を算出するために、次の式をワークシート"図書情報"のセルE2に入力し、セルE3~

E201 に複写する。

IF $(C2 \neq null, a , null)$

延滞日数を算出するために、次の式をワークシート"図書情報"のセルF2に入力し、セルF3~ F201 に複写する。

IF $(C2 \neq null, b, null)$

残り貸出冊数を算出するために、次の式をワークシート"利用者情報"のセル D2 に入力し、セ ル D3 ~ D51 に複写する。

С

延滞状態を表示するために、次の式をワークシート"利用者情報"のセル E2 に入力し、セル E3~E51に複写する。

d

a に関する解答群

- ア D2 + 垂直照合 (C2, 利用者情報!G\$2~ I\$5, 3, 0)
- ✓ D2 +垂直照合(C2,利用者情報!A\$2 ~ C\$51, 1, 0)
- ウ D2 + 垂直照合 (C2, 利用者情報!A\$2 ~ C\$51, 3, 0)
- D2 +垂直照合(垂直照合(C2,利用者情報!A\$2~C\$51,1,0),利用者情報!G\$2~ I\$5, 3, 0)
- **D2** + 垂直照合 (垂直照合 (C2、利用者情報!A\$2 ~ C\$51, 3, 0)、利用者情報!G\$2 ~ I\$5, 3, 0)
- D2 + 水平照合 (C2, 利用者情報!A\$2 ~ C\$51, 1, 0)
- ▶ D2 + 水平照合 (垂直照合 (C2, 利用者情報!A\$2 ~ C\$51, 1, 0), 利用者情報!G\$2 ~ I\$5, 3, 0)
- D2 + 水平照合(垂直照合(C2、利用者情報!A\$2 ~ C\$51, 3, 0)、利用者情報!G\$2 ~ I\$5, 3, 0)

bに関する解答群

E2 - I\$1

- **✓** I\$1 − E2
- $rac{1}{2}$ IF (E2 \geq I\$1, 0, I\$1 E2) IF (E2 \geq I\$1, E2, I\$1)
- **7** IF $(E2 \ge I\$1, I\$1, E2)$ **7** IF $(E2 \ge I\$1, I\$1 E2, 0)$

cに関する解答群

- **ア** 垂直照合 (C2, G\$2 ~ I\$5, 2, 0)
- 垂直照合(C2, G\$2 ~ I\$5, 2, 0) +条件付個数(図書情報!C\$2 ~ C\$201, = A2)
- ウ 垂直照合(C2, G\$2 ~ I\$5, 2, 0) 条件付個数(図書情報!C\$2 ~ C\$201, = A2)
- 重直照合(C2, G\$2 ~ I\$5, 3, 0)
- **才** 垂直照合 (C2, G\$2 ~ I\$5, 3, 0) +条件付個数 (図書情報!C\$2 ~ C\$201, = A2)

dに関する解答群

- \nearrow IF (D2 > 0, '*', null)
- IF (D2 = 0, ' * ', null)

- ウ IF (合計 (図書情報!C\$2 ~ C\$201) = C2, '*', null)
- IF (合計(図書情報!C\$2 ~ C\$201) ≠ C2, '*', null)
- J IF (合計 (図書情報!F\$2 ~ F\$201) = F2, '∗', null)
- 力 IF (合計 (図書情報!F\$2 ~ F\$201) ≠ F2, '*', null)
- IF (条件付合計 (図書情報!C\$2 ~ C\$201, = A2, 図書情報!F\$2 ~ F\$201) > 0, '*', null)
- ☑ IF (条件付合計(図書情報!C\$2 ~ C\$201, = A2, 図書情報!F\$2 ~ F\$201) = 0, '*',
 null)

設問 2 利用者が図書を借りる際、その情報を登録するマクロ Borrowing をワークシート "図書情報" に格納した。 ワークシート "図書情報" のセル B203 に図書 ID を、セル D203 に利用者 ID を入力して、マクロ Borrowing を実行すると、次に示す三つの条件を満たすとき、貸出登録される。

- (1) 対象図書が貸出し中でない。
- (2) 対象利用者の残り貸出冊数が1以上である。
- (3) 対象利用者が現在借りている全ての図書が延滞状態でない。

に入れる正しい答えを、解答群の中から選べ。

(マクロ: Borrowing)

○マクロ:Borrowing

♦ e

/* 貸出し可能か? */

- ・相対 (A1, B203, 2) ← D203
- ·相対(A1, B203, 3) ← I1
- ·B203 ← null
- D203 ← null

e に関する解答群

- 論理積(相対(A1, B203, 2) = null, 相対(利用者情報!A1, D203, 3) > 0,
 相対(利用者情報!A1, D203, 4) ≠ '*')
- ☆ 論理積(相対(A1, B203, 2) = null, 相対(利用者情報!A1, D203, 3) > 0, 相対(利用者情報!A1, D203, 4) = '*')

- 論理積(相対(A1, B203, 2)≠ null, 相対(利用者情報!A1, D203, 3)>0, 相対(利用者情報!A1, D203, 4)='*')

- ➡ 論理積(相対(A1, B203, 2)≠ null, 相対(利用者情報!A1, D203, 3)=0. 相対(利用者情報!A1, D203, 4)≠ '*')
- 論理積(相対(A1, B203, 2)≠ null, 相対(利用者情報!A1, D203, 3)=0, 相対(利用者情報!A1, D203, 4)='*')

設問 2 全利用者の過去の貸出履歴に基づき図書を推薦する機能について検討するため に、ワークシート "貸出履歴" 及びマクロ RecommendBooks を作成した。

(ワークシー	· 卜:貸出履歴〕
--------	-----------

	Α	В	С	D	E	•••	AW	AX	AY	AZ
1	図書ID\利用者ID	1	2	3	4	•••	48	49	50	推薦度
2	1	0	0	0	0	•••	1	0	0	0.63
3	2	0	0	0	1	•••	0	1	0	0
4	3	1	1	0	1	•••	0	0	0	0
5	4	0	0	0	0	•••	0	0	1	0.56
:	:		:	:	:			:	:.	:
201	200	1	0	0	0	•••	0	0	0	0
202	類似度	0.05	0	0.06	0.04	•••	0.07	0.055	0.065	
203										
204	利用者ID	2								

図3 ワークシート "貸出履歴"

- (1) 図書 ID をセル A2 ~ A201 に、利用者 ID をセル B1 ~ AY1 に入力する。セル B2 ~ AY201 には、利用者が図書を借りたことがあるならば1を、無ければ0を入力する。
- (2) セル B204 に利用者 ID を入力してマクロ Recommend Books を実行すると、セル AZ2 ~ AZ201 にその利用者に対する個々の図書の推薦度の値を表示する。
- (3) 利用者 ID に i を指定したとき、各図書の推薦度は次の方法で算出する。
 - ① 指定した利用者と残りの全ての利用者間の類似度を数値で表現し、セル B202 ~ AY202 に 求める。利用者 ID が i, j の利用者間の類似度は、次式で定義する s_{ii} で表現する。

$$s_{ij} = \begin{cases} \frac{\sum\limits_{k=1}^{200} \left(x_{ki} \times x_{kj}\right)}{200} & (i \neq j) \\ 0 & (i = j) \end{cases}$$

ここで、 x_{ki} は、利用者 ID i の利用者が、図書 ID k の図書を借りたことがあるならば 1, そうでなければ0である。

② 利用者 ID i の図書 ID k に対する推薦度 r_{ki} は類似度 s_{ij} を用いて表現し、セル AZ2 \sim AZ201 に表示する。

$$r_{ki} = \begin{cases} 0 & (x_{ki} = 1 又$$
は図書ID k の図書が貸出し中の場合)
$$\sum_{j=1}^{50} (s_{ij} \times x_{kj}) & (それ以外) \end{cases}$$

マクロ RecommendBooks 中の に入れる正しい答えを、解答群の中から選べ。

[マクロ: RecommendBooks] ○マクロ: RecommendBooks ○数値型:J, K, NumBook, NumUser · NumBook ← 200 ·NumUser ← 50 ■ J:1, J ≤ NumUser, 1 ·相対(A1, 201, J) ← 0 ♠ B204 ≠ J \blacksquare K:1, K \leq NumBook, 1 |・相対(A1, 201, J)← 相対(A1, 201, J)+「 ·相対(A1, 201, J) ← 相対(A1, 201, J) / NumBook \blacksquare K:1, K \leq NumBook, 1 ・相対 (A1, K, 51) ← 0 g ■J:1, J ≦ NumUser, 1 ・相対 (A1, K, 51) ← 相対 (A1, K, 51) 十 f. h に関する解答群 村対 (A1, B204, K)* 相対 (A1, J, K) ✓ 相対(A1, B204, K)* 相対(A1, K, J) 相対 (A1, J, K) * 相対 (A1, J, K) ■ 相対(A1, J, K)* 相対(A1, K, 51) 力 相対 (A1, K, B204) * 相対 (A1, K, J) ■ 相対 (A1, K, J) * 相対 (A1, 201, J) 夕 相対 (A1, K, J) * 相対 (A1, 201, K) g に関する解答群 → 論理積(相対(A1, K, B204) = 0, 相対(図書情報!A1, K, 2) ≠ null) 論理積(相対(A1, K, B204) = 1, 相対(図書情報!A1, K, 2) ≠ null) ➡ 論理和(相対(A1, K, B204) = 1, 相対(図書情報!A1, K, 2) = null) 論理和(相対(A1, K, B204) = 1, 相対(図書情報!A1, K, 2) ≠ null)

13 等 午後のカギ

表計算問題では、垂直照合関数、条件付個数関数、条件付合計といった関数の使い方に関する問題がほぼ毎回出題されます。これらはエクセルなどの実際の表計算ソフトでも、よく使われている関数です。また、マクロの問題では、相対表現を使ったセルの指定方法がとくに出題されやすいポイントです。詳しくは巻末の「表計算ソフトの機能・用語」を参照してください。

設問 1 垂直照合はエクセルの VLOOKUP, 条件付個数は COUNTIF, 条件付合計は SUMIF 関数と ほぼ同じ機能です。主な関数の使い方をマスターしておきましょう。

設問 2 相対表現によるセルの指定方法がポイントです。

設問3 問題文の数式をマクロで表します。空欄が数式のどの部分に当たるのかを考えます。

設問 1 ワークシートの完成

空欄 a: ワークシート"図書情報"のセル E2 に、返 却期日を求める計算式を入力します。ただし、図書が貸出中でなければセルには何も表示しないので、IF 関数を使って

IF(C2 ≠ null, ______, null)

↑

貸出中なら さもなけ
返却期日を れば空白
表示

とします。

返却期日は、貸出日+貸出日数で計算できます。ただし、貸出日数は利用者の属性によって異なります。利用者でとの属性と、属性別の貸出日数は、図2のワークシート"利用者情報"に入力されています。

そこでまず,ワークシート"利用者情報"を参照して,利用者の属性を求めます。このような処理には,垂直照合関数を使います。

垂直照合(式,セル範囲,列の位置,検索の指定) ① ② ③ ④

- ①式:属性を求める利用者 ID を指定します。利用者 ID は、**図 1** のワークシート "図書情報" のセル C2 に入力されています。
- ②セル範囲:利用者 ID に対応する属性を検索する範囲を指定します。利用者 ID に対応する属性を含むセル範囲は、ワークシート "利用者情報"のセルA2~C51です。なお、この範囲は複写しても変化しないので、行方向に複写したときずれないように行番号を固定します。
- ③列の位置:属性は、指定したセル範囲の左端から数

えて3列目に入力されているので,「3」を指定します。

④検索の指定:①で指定した利用者 ID と一致する行 を検索するので、「0」を指定します。

	Α	В	С	
1	利用者ID	氏名	属性	
2	1	情報一郎	学部生	
3	2	情報二郎	教員	利用者情報!
:	÷	÷	:	A\$2~C\$51
51	50	情報花子	その他	
列位	置 1	2	3	

完成した計算式は次のようになります。

垂直照合(C2,利用者情報! A\$2~C\$51,3,0)

この計算式で求めた属性に対応する貸出日数を求めるため、さらに垂直照合関数を使います。

- ①式:上記の垂直照合関数を指定します。
- ②セル範囲:属性に対応する貸出日数を検索する範囲を指定します。属性に対応する貸出日数を含むセル範囲は、ワークシート"利用者情報"のセル G2 ~ I5 です。この範囲は複写しても変化しないので、行方向に複写したときずれないように行番号を固定します。
- ③**列の位置**:貸出日数は,指定したセル範囲の左端から数えて3列目に入力されているので,「3」を指定します。
- ④検索の指定:①で指定した属性と一致する行を検索 するので,「0」を指定します。

	G	Н	I				
1	属性	貸出上限冊数	貸出日数				
2	教員	20	28				
3	大学院生	10	28				
4	学部生	5	14				
5	その他	3	7				

→ 利用者情報! G\$2~I\$5

列位置 1

2

以上から、完成した計算式は次のようになります。

垂直照合(垂直照合(C2,利用者情報! A\$2~ C\$51,3,0),利用者情報! G\$2~ I\$5,3,0)

この値(貸出日数)を、セル D2 に入力されている 貸出日に加算すれば、返却期日が求められます。

D2 十垂直照合 (垂直照合 (C2, 利用者情報! A\$2 ~ C\$51, 3, 0), 利用者情報! G\$2 ~ I\$5, 3, 0)

以上から、**空欄 a** は オ です。

空欄 b: 図 1 のワークシート "図書情報" のセルF2に、延滞日数を求める計算式を入力します。延滞日数は、今日の日付-返却期日で求めます。ただし、まだ返却期日を過ぎていなければ 0 日になります。このような計算式は、IF 関数を使って、次のように表せます。

IF (返却期日≧今日の日付, 0, 今日の日付一返却期日)

今日の日付はセル I1, 返却期日はセル E2 に入力されているので、

IF $(E2 \ge I1, 0, I1 - E2)$

となります。さらに、行方向に複写してもずれないように、複写しても変わらないセル範囲の行番号を固定します。

IF $(E2 \ge I\$1, 0, I\$1 - E2)$

以上から、**空欄 b** は <mark>ウ</mark> です。

条件式は「E2 > I \$1」としても結果は同じだよ。



空欄 c: 図 2 のワークシート "利用者情報" のセル D2 に、残り貸出冊数を求める計算式を入力します。

残り貸出冊数は、貸出上限冊数-現在の貸出冊数で 求めます。貸出上限冊数は、利用者の属性によって異 なるので、垂直照合関数で次のように求めます。

垂直照合(式,セル範囲,列の位置,検索の指定)









- ①**式**:利用者の属性が入力されているセル C2 を指定します。
- ②セル範囲:属性に対応する貸出上限冊数は、ワークシート"利用者情報"のセル G2~ I5 に入力されています。この範囲は複写しても変化しないので、行方向に複写したときずれないように行番号を固定します。
- ③**列の位置**:貸出上限冊数は、上記のセル範囲の左端 から数えて2列目に入力されているので、[2] を指 定します。
- ④検索の指定:①で指定した属性と一致する行を検索するので,「0」を指定します。

以上から、貸出上限冊数は次のように求められます。

垂直照合(C2, G\$2~I\$5, 2, 0)

現在の貸出冊数は、図1のワークシート"図書情報"の中から、利用者が現在借りている図書をすべて数えます。このような計算には、条件付個数という関数が利用できます。

条件付個数(セル範囲、検索条件の記述)





- ①セル範囲: ワークシート "図書情報"の利用者 ID の欄(C2~C201)を指定します。この範囲は複写しても変化しないので、行方向に複写したときずれないように行番号を固定します。
- ②検索条件の記述:①で指定した範囲中に、セル A2 と一致する利用者 ID がいくつあるかを求めるので、 検索条件は「= A2」とします。

以上から、貸出冊数は次のように求められます。

条件付個数 (図書情報! C\$2~C\$201, = A2)

残り貸出冊数は、貸出上限冊数-現在の貸出冊数なので、次のようになります。

垂直照合 (C2, G\$2~I\$5, 2, 0) 一条件付個数 (図書情報! C\$2~C\$201, = A2)

以上から,**空欄c**は ウです。

空欄 d: 図2のワークシート"利用者情報"のセル

E2 には、利用者が借りている図書が延滞状態のとき に"*"記号を表示する計算式を入力します。

利用者の借りている図書の中に、延滞状態のものが あるかどうかは、図1のワークシート"図書情報" から利用者の利用者 ID がある行をすべて検索し、そ の延滞日数を合計します。合計が 0 より大きければ、 少なくとも1冊は延滞状態であることがわかります。

指定した利用者の延滞日数を合計するには、条件付 合計という関数を使います。

条件付合計(検索範囲,検索条件,合計範囲)

1

(2)

- ①検索範囲:利用者 ID は C 列に入力されているので 検索範囲はワークシート"図書情報"のC2~ C201 とします。行方向に複写したときずれないよ うに、行番号を固定しておきます。
- ②検索条件:①で指定した範囲から、セル A2 と一致 する利用者 ID を抜き出すので、検索条件は「= A2| とします。
- ③合計範囲:条件に合致した場合には、ワークシート "図書情報"の対応する延滞日数を合計に含めます。 したがって、延滞日数が入力されている F2~ F201 を合計範囲とします。行方向に複写したとき ずれないように、行番号を固定しておきます。

以上から、延滞日数の合計を求める計算式は次のよ うになります。

条件付合計(図書情報! C\$2~ C\$201, = A2, 図 書情報! F\$2~F\$201)

図2のセル E2 は、上の計算式が0より大きければ "*". さもなければ空白 (null) を表示します。IF 関数を使い、次のように指定します。

IF (条件付合計 (図書情報! C\$2~C\$201, = A2, 図書情報! F\$2~F\$201) > 0, '*', null)

以上から、**空欄 d** は ‡ です。

設問2 貸出しが可能な条件

マクロ Borrowing の空欄 e には、貸出し可能か どうかを調べる条件式が入ります。問題文によれば, 貸出し可能となる条件は、次の3つです。

- ①対象図書が貸出し中でない
- ②対象利用者の残り貸出冊数が1以上である
- ③対象利用者が現在借りている全ての図書が延滞状態 でない

以上3つの条件を表す式を順に検討します。

①対象図書が貸出し中でない

この条件は、**図1**のセル B203 に入力された対象図 書の、利用者 ID のセルが空白かどうかで判断できま す。

対象図書の利用者 ID が入力されているセルは、セ ル A1 を起点にすると.

セル A1 から B203 行目の 2 列目

	Α		В	С
1	図書ID		図書ID 図書名	
		B203	••	
51		170	<u> 2</u> →	null

です。これは、マクロの相対表現を使うと、「相対 (A1, B203, 2) | と表せます。このセルが空白(null)で あれば、貸出し中でないと判断できるので、条件式は 次のようになります。

相対 (A1, B203, 2) = null

②対象利用者の残り貸出冊数が 1 以上である

図2のワークシート"利用者情報"の残り貸出し 冊数欄を参照します。

対象利用者の残り貸出し冊数の欄は、ワークシート "利用者情報"の A1 を起点にすると、

利用者情報! A1 から D203 行目の 3 列目

	Α		В	С	D
1	利用者ID		氏名	属性	残り 貸出冊数
		D203	:	:	:
41		40	3 -		1以上

となります。この値が1以上であればいいので、条件 式は次のようになります。

相対(利用者情報! A1, D203, 3) > 0

③現在借りている全ての図書が延滞状態でない

ワークシート "利用者情報" の延滞状態の欄を参照 します。この欄は、ワークシート "利用者情報" の A1 を起点にすると、

利用者情報! A1 から D203 行目の 4 列目

となります。この値が "*" でなければいいので、条件式は次のようになります。

相対 (利用者情報! A1, D203, 4) ≠ '*'

貸出し可能の条件は、以上①~③の条件がすべて真のときなので、論理積関数を使って、次のように指定します。

論理積(相対(A1, B203, 2) = null, 相対(利用者情報! A1, D203, 3) > 0, 相対(利用者情報! A1, D203, 4) \neq '*')

以上から、**空欄 e** は ア です。

設問3 マクロRecommendBooksの完成

空欄 f:空欄の前後は次のとおりです。

■J:1, J ≤ NumUser,1
・相対 (A1,201,J) ← 0

■B204 ≠ J
■K:1, K ≤ NumBook, 1
・相対 (A1,201,J) ← 相対 (A1,201,J)

+ f

・相対 (A1,201,J) ← 相対 (A1,201,J)

/ Numbook

この部分は、各利用者の利用者 i に対する類似度 s_{ij} を求めています。このうち、

■ K:1, K ≦ NumBook, 1
・相対(A1,201,J) ← 相対(A1,201,J)+ f

の部分は, $\sum_{k=1}^{200}(x_{ki}\times x_{kj})$ を求めます。ここで, x_{ki} 、 x_{kj} は,マクロの相対表現を使って,それぞれ次のように表せます。

xki:相対(A1, K, B204)

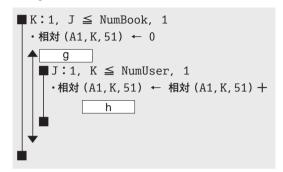
 x_{kj} :相対(A1, K, J)

したがって、**空欄f**には、次のような式が入ります。

相対 (A1, K, B204) * 相対 (A1, K, J)

以上から、**空欄f**は<u>カ</u>です。

空欄 q:空欄の前後は次のとおりです。



この部分は、利用者iの各図書IDkに対する推薦度 r_{ki} を求めています。このうち、

■J:1, K ≦ NumUser, 1
・相対(A1,K,51) ← 相対(A1,K,51) + h

の部分は, $\sum\limits_{j=1}^{50}\left(s_{ij}\times x_{kj}\right)$ の値を求めます。

問題文によれば、 r_{ki} は、 $x_{ki}=1$ または図書 $\mathrm{ID}k$ が貸出し中の場合 0 となります。 $r_{ki}=\sum_{j=1}^{50}\left(s_{ij}\times x_{kj}\right)$ となるのはそれ以外の場合、すなわち、

 $x_{ki} = 0$ かつ 図書 $\mathsf{ID} k$ が貸出し中でない

場合です。空欄gには、上の条件を表す条件式が入ります。

条件①: x_{ki} は、図 3 のセル B204 に入力した利用者が、図書 IDk を借りたかどうかを表す値で、マクロの相対表現で相対 (A1, K, B204) と表せます。したがって $x_{ki}=0$ に相当する条件式は次のようになります。

相対 (A1, K, B204) = 0

条件②:図書 IDk が現在貸出し中でないことは,ワークシート"図書情報"の該当する行の利用者 ID 欄が,空白であることから判断できます。したがって,条件式は次のようになります。

相対 (図書情報!A1, K, 2) = null

以上の条件を論理積関数の引数に指定すると、空欄 gに入る条件式になります。

論理積(相対(A1, K, B204)=0, 相対(図書情 報!A1, K, 2) = null)

以上から,**空欄 g** は <mark>ア</mark> です。

空欄 h:空欄は,推薦度 r_{ki} を求める式 $\sum_{i=1}^{50} (s_{ij} \times x_{kj})$ の, $s_{ij} \times x_{kj}$ の部分を求める計算式です。 s_{ij} 、 x_{kj} は,そ れぞれ次のように表せます。

 s_{ij} :相対(A1, 201, J) x_{kj} :相対(A1, K, J)

したがって. 空欄 h には. 次のような式が入ります。

相対(A1, 201, J)* 相対(A1, K, J)

解答欄には上記と同じ計算式はありませんが,

 #
 相対(A1, K, J)*
 相対(A1, 201, J)

でも同様なので、まが正解となります。

解答

設問 1 a - オ, b - ゥ, c - ゥ,

d — 🝍

設問 2 e - 7

設問3 f 一 力 , g 一 ア , h 一 丰

かんたん合格 基本情報技術者過去問題集 平成 26 年度秋期 購入者限定特典

平成 24 年度春期試験(ダウンロード版)

2014年6月1日 初版発行

著 者 株式会社ノマド・ワークス

発行人 土田米一

発 行 株式会社インプレスジャパン An Impress Group Company

〒 102-0075 東京都千代田区三番町 20 番地

発 売 株式会社インプレスコミュニケーションズ An Impress Group Company

〒 102-0075 東京都千代田区三番町 20 番地

出版営業 TEL 03-5275-2442 http://www.ips.co.jp

本書は著作権法上の保護を受けています。本書の一部あるいは全部について、株式会社インプレスジャパンから文書による許諾を得ずに、いかなる方法においても無断で複写、複製することは禁じられています。

Copyright © 2014 NOMAD WORKS. All rights reserved.