2010年 9 月26日 実施

平成22年度(第43回) 情報処理検定試験 〈プログラミング部門〉

第1級 試験問題

- 1. 監督者の指示があるまで、試験問題に手を触れないでください。
- 2. 試験問題は9ページあります。
- 3. 解答はすべて解答用紙に記入します。
- 4. 【1】【2】【3】は共通問題です。
- 【4】【5】【6】【7】の問題は、COBOL・イベント駆動型BASIC
 のどちらか1つを選択し、解答用紙の選択言語を()で囲んでください。
- 6. 電卓などの計算用具は使用できません。
- 7. 筆記用具などの物品の貸借はできません。
- 8. 問題用紙の回収については監督者の指示にしたがってください。
- 9. 制限時間は60分です。

主催 舞 全国商業高等学校協会

【1】 次の説明文に最も適した答えを解答群から選び、記号で答えなさい。

- 1. コンピュータに処理の指示を出してから、すべての実行結果が得られるまでの時間。
- 2. コンピュータに接続して使用するプリンタやイメージスキャナなどの周辺機器を動作させるためのソフトウェア。
- 3. 各国の言語や,画像,音声,動画など形式の異なる様々なデータを電子メールで扱うための規格。
- 4. コンピュータに接続された入出力装置を制御するプログラムの集まりのこと。電源を入れるとOSより先に起動し、入出力装置に対する認識や初期化を行う。
- 5. 開発したシステム全体が、設計したとおりの機能を備えているかを確認する開発者側の最終テスト。

- 解答群 -

ア. MIME

イ. レスポンスタイム

ウ. システムテスト

エ. 結合テスト

オ. VRAM

カ. デバイスドライバ

キ. スループット

Э. VоІР

ケ. デバイス

⊐. BIOS

サ. ターンアラウンドタイム

シ. Cookie

【2】 次のA群の語句に最も関係の深い説明文をB群から選び、記号で答えなさい。

〈A群〉 1. プライベートIPアドレス

2. 電子署名

3. DNS

4. ディスクキャッシュ

5. RASIS

〈B群〉

- ア、インターネットに接続された機器を識別するための番号。インターネット上で一意の番号となっている。
- **イ**. 電子商取引などで、送受信する電子文書の作成者が本人であることと、データが改ざんされていないことを確認できるしく み。
- ウ. 送信者と受信者で、暗号化と復号に同じ鍵を使用する暗号方式。
- エ. コンピュータシステムの評価に用いられるチェック項目で「信頼性」、「可用性」、「保守性」、「保全性」、「機密性」の5項目。
- **オ**. 会社や家庭のLAN内部でコンピュータ機器を識別するための番号。ネットワークアドレス部とホストアドレス部で構成されている。
- **カ**. コンピュータ機器をネットワークに接続したとき, 空いている I P アドレスを自動的に割り当てるしくみ。
- **キ.** CPUと主記憶装置の間に位置し、使用頻度が高いデータや命令を保存し処理を高速化するための記憶装置。
- **ク**. ハードディスクなど低速な記憶装置からデータを読み込むとき、データを一時的にメモリに格納しておくことにより処理を 高速化するしくみ。
- **ケ**. インターネットで使用するドメイン名と I P アドレスを相互に変換するしくみ。
- **コ.** 複数のハードディスクを1台のハードディスクのように使用し、信頼性や処理速度を高めるためのしくみ。

【3】 次の説明文に最も適した答えをア、イ、ウの中から選び、記号で答えなさい。

1. 2進数の00110111を2の補数で表したもの。

ア. 11000111

イ. 11001000

ウ. 11001001

- 2. レコードが入力順に連続して記録され、アクセスするときには、先頭レコードから順番にレコードを読み書きするファイル。
 - ア. 順次編成ファイル
- **イ**. 直接編成ファイル
- ウ. 索引順次編成ファイル
- 3. コンピュータ内部の数値を,上位ビットから符号部分,整数部分,小数部分の順に並べ,小数点の位置をあらかじめ決めて表現したもの。
 - **ア**. 浮動小数点形式
- **イ**. 固定小数点形式

- ウ. 2 進化10進数
- 4. ハードディスクなどの記憶装置で、磁気ヘッドがディスク上の目的のトラックに移動するまでの平均所要時間。
 - ア. データ転送時間
- イ. 平均回転待ち時間
- **ウ**. 平均シーク時間
- 5. 60,000バイト/トラック、30トラック/シリンダ、シリンダ数600の磁気ディスク装置の記憶容量は何Mバイトか。ただし、1 Kバイト=1,000バイト、1 Mバイト=1,000 Kバイトとする。
 - **ア.** 1.08Mバイト

イ. 1.080Mバイト

ウ. 1,080,000Mバイト

【4】 第1図のようなある野球チームの打席結果ファイルを読み、処理条件にしたがって第2図のような打席結果集計表を印字した い。第6図の流れ図(1)~(5)にあてはまる答えを解答群から選び、記号で答えなさい。

入力形式

				1
背番号	試合日	打席	結果コード	
(BAN)	(HI)	(DASEKI)	(KCD)	
$\times \times$	$\times \times \times \times$	×	×	(第1図)

出力形式

	(打席結果集計表)								
(背番号)	(選手名)	(打数)	(安打)	(犠打)	(四死球)	(打率)			
1	$\circ\sim\circ$	538	155	9	70	0.288			
6	$\triangle \sim \triangle$	519	135	8	93	0.260			
}	}	₹	}	₹	}	}			
67	$\square \sim \square$	75	24	0	8	0.320			
(チーム成績)		4, 480	1, 143			0.255			

(第2図)

処理条件

(1) 第1図の打席結果ファイルは、次の例のように記録され ており、背番号、試合日の昇順に整列されている。なお、 結果コードは次のとおりである。

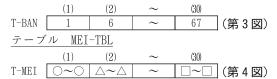
1:安打 2:犠打 3:四死球 4:凡打

例 06092631

背番号6の選手が、9月26日の試合で第3打席目に 安打を打った場合のレコード1件分の内容。

(2) 野球チームに所属している選手は30名であり、第3図の テーブル BAN-TBL に背番号が、第4図のテーブル MEI-TBL に選手名が、背番号の昇順にあらかじめ記憶されている。 なお, テーブル BAN-TBL とテーブル MEI-TBL は添字で対応 している。

テーブル BAN-TBL



(3) 第1図の打席結果ファイルを読み,第5図のテーブル SYU-TBL に打席結果を集計する。なお、結果コードはテーブル SYU-TBLの添字と対応している。

テーブル SYU-TBL



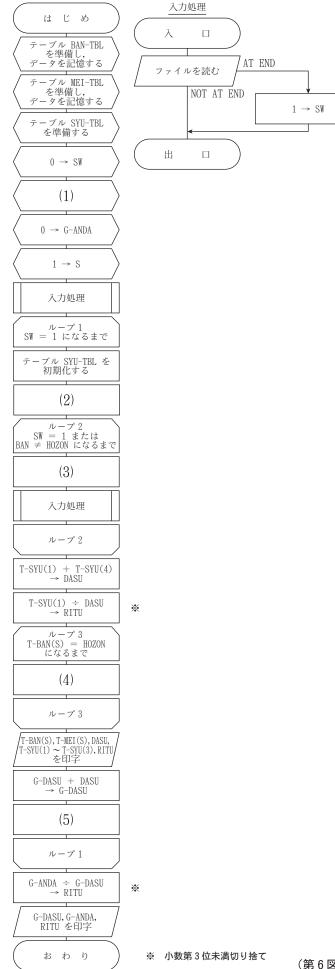
(4) 選手名ごとに打数と打率を次の式で計算し、第2図のよ うに背番号から打率を印字する。

打数 = 安打 + 凡打

打率 = 安打 ÷ 打数 (小数第3位未満切り捨て)

- (5) データが終わりになったら、チーム成績を印字する。
- (6) データにエラーはないものとする。

- $7. \text{ G-ANDA} + \text{T-SYU}(1) \rightarrow \text{G-ANDA}$
- **1.** T-SYU(HOZON) + 1 → T-SYU(HOZON)
- ウ. $S + 1 \rightarrow S$
- **I**. BAN → HOZON
- **オ.** 1 → G-DASU
- カ. T-SYU(KCD) + 1 \rightarrow T-SYU(KCD)
- **‡.** HI → HOZON
- \mathcal{D} . G-ANDA + T-SYU(S) \rightarrow G-ANDA
- σ . 0 \rightarrow G-DASU
- \exists . $S 1 \rightarrow S$



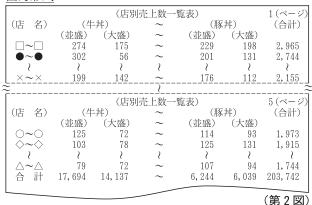
(第4図)

【5】 第1図のようなある丼ぶり屋チェーンの売上ファイルを読み、処理条件にしたがって第2図のような店別売上数一覧表を印字したい。第4図の流れ図(1)~(5)にあてはまる答えを解答群から選び、記号で答えなさい。

人刀形式				
月日	店コード	メニューコード	種類コード	売上数
(GAPPI)	(MISE)	(MENU)	(SHU)	(SU)
$\times \times \times \times$	$\times \times \times$	××	×	$\times \times \times$

(第1図)

出力形式



処理条件

(1) (2)

(100) (101)

(102)

(1) チェーン店は100店舗あり,第 3 図のテーブル MEI-T に店名が記憶されている。なお,店コードは $1\sim100$ まであり,テーブル MEI-T の添字とテーブル URI-T の行方向の添字は店コードに対応している。

 $\frac{\mathcal{F} - \mathcal{J} \mathcal{N} \quad \text{MEI-T}}{\mathcal{F}}$

 $\triangle \sim \triangle$

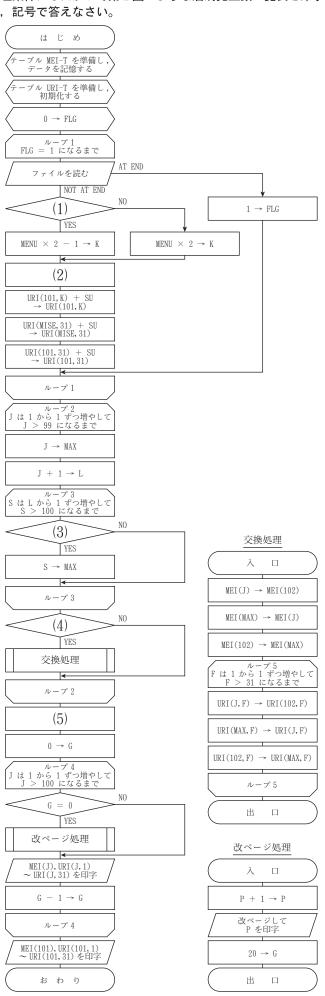
合 計

_			UICI I					
	URI	(1)	(2)	~	(29)	(30)	(31)	
	(1)			~				
	(2)			~				
	₹	₹	}	- }	₹	}	- }	
	(100)			~				
	(101)			~				(合計)
	(102)			~				
		(並盛)	(大盛)	~	(並盛)	(大盛)	(合計)	
		(牛	丼)	\sim	(豚	丼)		

(第3図)

- (2) 第1図の売上ファイルを読み,第3図のテーブル URI-T に売上数を次のように集計する。
 - ① メニューコードは $1 \sim 15$ まであり、それぞれ種類コードにより 1 (並盛)、2 (大盛) に分かれて記録されている。なお、テーブル URI-T の構成は列方向に対して、メニューごとに並盛と大盛が連続している。
 - ② 101行目に売上数合計を,31列目に店別売上数合計 を,101行目の31列目にチェーン全体の売上数合計を 集計する。
- (3) テーブル URI-T に売上数を集計した後,第 3 図のテーブル MEI-T とテーブル URI-T を店別売上数合計の降順に並べ替える。
- (4) 第2図のように店別売上数一覧表を印字する。なお,明細行は20行印字するごとに,改ページする。また,最終ページには合計を印字する。
- (5) データにエラーはないものとする。

- 7. URI(S, 31) < URI(MAX, 31)
- 1. J ≠ MAX
- ウ. SHU = 1
- \mathbf{I} . 1 \rightarrow P
- オ. $URI(K, MISE) + SU \rightarrow URI(K, MISE)$
- カ. SHU = 2
- \pm . URI(MISE, K) + SU \rightarrow URI(MISE, K)
- $\boldsymbol{\mathcal{D}}$. J = MAX
- **ケ**. 0 → P
- \exists . URI(S, 31) > URI(MAX, 31)



【6】 ある企業では、2つの工場で共通して使用する部品を1か月ごとに一括して発注している。第1図のような部品入出庫ファイルと第2図のような部品在庫マスタファイルを読み、処理条件にしたがって、新部品在庫マスタファイルを作成するとともに、第3図のような発注一覧表を印字するプログラムの空欄をうめなさい。

入出力形式

上段:A工場部品入出庫ファイル(ファイル名:A-F, レコード名:A-R) 中段:B工場部品入出庫ファイル(ファイル名:B-F, レコード名:B-R)

下段:併合部品入出庫ファイル(ファイル名:II-F, レコード名:II-R)

日付	区分	部品コード	数量
(A-MD)	(A-KU)	(A-BCD)	(A-SURYO)
(B-MD)	(B-KU)	(B-BCD)	(B-SURYO)
(H-MD)	(H-KU)	(H-BCD)	(H-SURYO)
$\times \times \times \times$	×	$\times \times \times \times$	$\times \times \times \times$

|(第1図)

入出力形式

上段: 部品在庫マスタファイル(ファイル名: 0-F, レコード名: 0-R) 下段: 新部品在庫マスタファイル(ファイル名: N-F, レコード名: N-R)

部品コード	在庫数	基準在庫数
(O-BCD)	(O-ZAI)	(O-KZAI)
(N-BCD)	(N-ZAI)	(N-KZAI)
$\times \times \times \times$	$\times \times \times \times \times$	$\times \times \times \times \times$

(第2図)

出力形式

(ファイル名:LIST-F, レコード名:LIST-R)



(第3図)

処理条件

- (1) 第1図のA工場部品入出庫ファイルとB工場部品入出庫ファイルを読み、部品コードの昇順に併合し、併合部品入出庫ファイルを作成する。なお、2つの工場の部品入出庫ファイルは、部品コードの昇順に記録されている。また、同じ部品コードの場合は、A工場部品入出庫ファイルを先に、併合部品入出庫ファイルに書き出す。
- (2) 第2図の部品在庫マスタファイルと併合部品入出庫ファイルを読み、新部品在庫マスタファイルを作成する。なお、部品在庫マスタファイルは、部品コードの昇順に記録されており、部品在庫マスタファイルの部品コードに対して、併合部品入出庫ファイルの部品コードは複数件ある場合もあり、ない場合もある。
- (3) 第1図の部品入出庫ファイルの区分は次のとおりであり、 在庫数を更新する。
 - 1:入庫 (在庫数に加算する)
 - 2:出庫(在庫数から減算する)
- (4) 在庫数が基準在庫数より小さくなった場合は、第3図のように部品コードと発注数を印字する。なお、発注数は次の計算式で求める。

発注数 = 基準在庫数 - 在庫数

- (5) 併合部品入出庫ファイルの部品コードが部品在庫マスタファイルにない場合は、エラーとしてディスプレイに部品コードを表示する。
- (6) 部品コード以外のデータにエラーはないものとする。

```
PROCEDURE DIVISION
SYORL
    OPEN INPUT A-F B-F OUTPUT H-F
    PERFORM READ-A THRU READ-B
    PERFORM UNTIL A-BCD = 9999 AND B-BCD = 9999
                (ア)
        THEN
          WRITE H-R FROM B-R
         PERFORM READ-B
       FI SF
          WRITE H-R FROM A-R
          PERFORM READ-A
      END-IF
    END-PERFORM
    CLOSE A-F B-F H-F
    OPEN INPUT O-F H-F OUTPUT N-F LIST-F
    PERFORM READ-H THRU READ-O
    PERFORM UNTIL
      EVALUATE TRUE
        WHEN O-BCD = H-BCD
          IF H-KU = 1
            THEN
            ELSE
              COMPUTE O-ZAI = O-ZAI - H-SURYO
          FND-IF
          PERFORM READ-H
        WHEN O-BCD < H-BCD
          IF O-ZAI < O-KZAI
            THEN
              COMPUTE W-ORDER = O-KZAI - O-ZAI
              MOVE O-BCD TO P-BCD
              MOVE W-ORDER TO P-ORDER
              WRITE LIST-R FROM PRINT-R AFTER 1
          FND-IF
                     (I)
          PERFORM READ-0
        WHEN OTHER
          DISPLAY "部品コードエラー: "H-BCD
                (オ)
      END-EVALUATE
    END-PERFORM
    CLOSE O-F H-F N-F LIST-F
    STOP RUN.
READ-A.
    READ A-F AT END MOVE 9999 TO A-BCD.
    READ B-F AT END MOVE 9999 TO B-BCD.
READ-H.
    READ H-F AT END MOVE 9999 TO H-BCD.
READ-0.
```

READ O-F AT END MOVE 9999 TO O-BCD.

【7】 第2図のようなあるミュージカル公演の予約希望ファイルを読み、処理条件にしたがって第3図のような予約希望明細と 残席状況一覧を印字するプログラムの空欄をうめなさい。

入力形式

公演ファイル(ファイル名: KOUEN-FL, レコード名: KOUEN-REC)

公演コード	公演日	公演時間
(K-KODO)	(K-TUKIHI)	(K-JI)
$\times \times \times$	$\times \times \times \times$	×~×

(第1図)

予約希望ファイル(ファイル名:YOYAKU-FL, レコード名:YOYAKU-REC)

申込番号	公演コード	席種
(Y-BAN)	(Y-KODO)	(Y-SYU)
$\times \times \times \times \times$	$\times \times \times$	×

(第2図)

出力形式(ファイル名: PRINT-FL, レコード名: PRINT-REC)

i							
	(予約希望明	[細]					
	(申込番号)	(公演コード)	(席種	(予	約状況)		
	10001	101	1	OK			
	₹	}	}	}			
	90321	505	3	S0	LD OUT		
			(S席) 38 ≀	(A席) 52 ≀	(B席) 10 ~	(空席率(%)) 33 {	(人気順位) 26 ~

(第3図)

処理条件

(1) 第1図の公演ファイルを読み,第4図のテーブル MAST-TBL に記憶する。この公演は全50回行われ,ペアチケット(2席分)のみが販売される。なお,データは公演コードの昇順に記録されている。

テーブル MAST-TBL

	M-KODO	M-TUKIHI	M-JI
MAST-T(1)			
₹		}	}
MAST-T (50)			

(公演コード) (公演日) (公演時間) (第4図)

(2) 第 5 図のテーブル ZAN-TBL の $1 \sim 3$ 列目に S 席,A 席,B 席 のそれぞれの座席数である 100 を記憶する。なお,テーブル ZAN-TBL の行方向の添字は,テーブル MAST-TBL の添字と対応している。

テーブ<u>ル ZAN-TBL</u>

ZAN		(1)	(2)	(3)	(4)	(5)	
	(1)						
	}	}		}	}	}	
	(50)						
		(S席)	(A席)	(B席)	(空席率)	(人気順位)	(第5図)

- (3) 第2図の予約希望ファイルを読み、次の処理を行う。
 - ① 第2図の席種は、1 (S席)、2 (A席)、3 (B席)のいずれかが記録されており、第5図のテーブル ZAN-TBL の列方向の添字と対応している。
 - ② 第2図の公演コードをもとに, 第4図のテーブル MAST-TBL を探索し, 行方向の添字を求め, 第5図のテーブル ZAN-TBL の席種ごとに残席数を計算する。
 - ③ 第3図のように、予約希望明細を印字する。なお、残席が2 席以上ある場合はOKを、残席が2席未満の場合はSOLDOUT を予約状況に印字する。
- (4) 予約希望ファイルを読み終えた後、次の計算式のように空席率を求め、第5図のテーブルZAN-TBLの4列目に記憶する。なお、計算式の残席計とは、第5図のテーブルZAN-TBLの席種ごとの残席数を集計したものである。

空席率 = 残席計 × 100 ÷ 300 (小数点以下切り捨て)

- (5) 空席率の昇順に順位をつけ、テーブル ZAN-TBL の 5 列目に記憶する。なお、空席率が同じ場合は、同順位とする。
- (6) 第3図のように、残席状況一覧を印字し、処理を終了する。
- (7) データにエラーはないものとする。

```
PROCEDURE DIVISION.
S1. OPEN INPUT KOUEN-FL YOYAKU-FL OUTPUT PRINT-FL
    INITIALIZE END-FLG
   PERFORM VARYING J FROM 1 BY 1 UNTIL J > 50
     READ KOUEN-FL INTO MAST-T(J)
     END-READ
   FND-PFRFORM
   PERFORM VARYING J FROM 1 BY 1 UNTIL J > 50
      PERFORM VARYING P FROM 1 BY 1 UNTIL P > 3
       MOVE 100 TO (ア)
     END-PERFORM
   END-PERFORM
    PERFORM UNTIL END-FLG = 1
      READ YOYAKU-FL
        AT FND
         MOVE 1 TO END-FLG
       NOT AT END
         MOVE 0 TO SW
         MOVE 51 TO UE
         MOVE 0 TO SITA
         PERFORM UNTIL SW = 1
           COMPUTE M = (UE + SITA) / 2
           EVALUATE TRUE
             WHEN [___
                           (イ)
               MOVE 1 TO SW
              WHEN M-KODO(M) > Y-KODO
               MOVE M TO UE
             WHEN OTHER
               MOVE M TO SITA
           END-EVALUATE
         END-PERFORM
           THEN COMPUTE ZAN(M Y-SYU) = ZAN(M Y-SYU) - 2
                MOVE "OK"
                              TO I -JYOU
           ELSE MOVE "SOLD OUT" TO L-JYOU
         END-IF
         MOVE Y-BAN TO L-BAN
         MOVE Y-KODO TO L-KODO
         MOVE Y-SYU TO L-SYU
         WRITE PRINT-REC FROM LIST-REC AFTER 1
     END-READ
    END-PERFORM
    PERFORM VARYING J FROM 1 BY 1 UNTIL J > 50
      COMPUTE G-ZAN = ZAN(J 1) + ZAN(J 2) + ZAN(J 3)
      COMPUTE ZAN(J 4) = G-ZAN * 100 / 300
    FND-PERFORM
    PERFORM VARYING J FROM 1 BY 1 UNTIL J = 50
      COMPUTE K = J + 1
      PERFORM VARYING P FROM
       EVALUATE TRUE
         WHEN ZAN(J 4) < ZAN(P 4)
           COMPUTE ZAN(P 5) = ZAN(P 5) + 1
         WHEN ZAN(J 4) > ZAN(P 4)
           COMPUTE ZAN(J 5) = ZAN(J 5) + 1
       END-EVALUATE
      END-PERFORM
   END-PERFORM
   PERFORM VARYING J FROM 1 BY 1 UNTIL J > 50
     MOVE M-TUKIHI(J) TO Z-TUKIHI
     MOVE M-JI(J)
                      TO Z-JI
      PERFORM VARYING P FROM 1 BY 1 UNTIL P > 5
       MOVE ZAN(J P)
                      TO Z-ZAN(P)
      END-PERFORM
      WRITE PRINT-REC FROM ZAN-REC AFTER 1
    FND-PFRFORM
    CLOSE KOUEN-FL YOYAKU-FL PRINT-FL
```

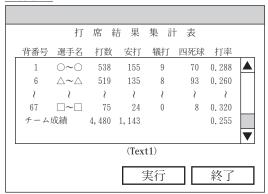
STOP RUN.

【4】 第1図のようなある野球チームの打席結果データを読み、処理条件にしたがって第2図のような打席結果集計表を表示したい。 第6図の流れ図(1)~(5)にあてはまる答えを解答群から選び、記号で答えなさい。

入力形式

背番号	試合日	打席	結果コード	
(Bn)	(Hi)	(Daseki)	(Kcd)	
$\times \times$	\times	×	×	(第1図)

実行形式



(第2図)

処理条件

(1) 第1図の打席結果データは、次の例のように記録されて おり、背番号、試合日の昇順に整列されている。なお、結 果コードは次のとおりである。

1 : 安打 2 : 犠打 3 : 四死球 4 : 凡打

例 06,0926,3,1

背番号 6 の選手が、9月26日の試合で第3打席目に 安打を打った場合のデータ1件分の内容。

(2) 野球チームに所属している選手は30名であり、 ロード時に第3図の配列Banに背番号を,第4図の配列 Mei に選手名を、背番号の昇順に記憶する。なお、配列 Ban と配列 Mei は添字で対応している。 配列

	(1)	(2)	\sim	(30)	
Ban	1	6	~	67	(第3図)
西己安日					

	(1)	(2)	\sim	(30)	
Mei	0~0	$\triangle \sim \triangle$	~	_~_	(第4図)

(3) 「実行」ボタンをクリックすると、次の処理を行う。 ① 第1図の打席結果データを読み,第5図の配列 Syu に 打席結果を集計する。なお、結果コードは配列 Syu の 添字と対応している。

配列



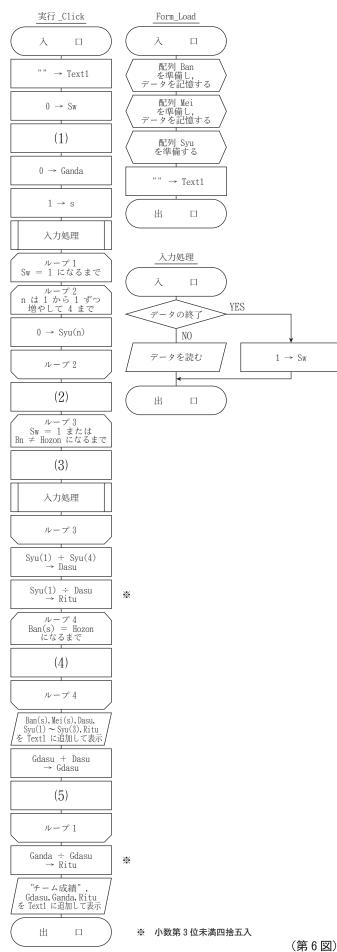
② 選手名ごとに打数と打率を次の式で計算し、第2図の ように背番号から打率を Text1 に追加して表示する。

打数 = 安打 + 凡打

打率 = 安打 ÷ 打数 (小数第3位未満四捨五入)

- データが終わりになったら、チーム成績を Text1 に 追加して表示する。
- (4) データにエラーはなく、操作は正常に行われるものとする。

- **7.** Ganda + Syu(1) \rightarrow Ganda
- **1.** Syu(Hozon) + 1 → Syu(Hozon)
- ウ. $s + 1 \rightarrow s$
- \mathbf{I} . Bn \rightarrow Hozon
- **オ.** 1 → Gdasu
- カ. $Syu(Kcd) + 1 \rightarrow Syu(Kcd)$
- **‡.** Hi → Hozon
- \mathcal{D} . Ganda + Syu(s) \rightarrow Ganda
- σ . 0 \rightarrow Gdasu
- \exists . s $-1 \rightarrow s$

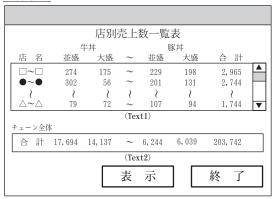


【5】 第1図のようなある丼ぶり屋チェーンの売上データを読み、処理条件にしたがって第2図のような店別売上数一覧表を表示し たい。第4図の流れ図(1)~(5)にあてはまる答えを解答群から選び、記号で答えなさい。

人刀形式				
月日	店コード	メニューコード	種類コード	売上数
(Gappi)	(Mise)	(Menu)	(Shu)	(Su)
$\times \times \times \times$	$\times \times \times$	$\times \times$	×	$\times \times \times$

(第1図)

実行形式



(第2図)

(1) チェーン店は100店舗あり、フォームロード時に第3図の 配列 Mei に店名を記憶する。なお、店コードは1~100 まであり、配列 Mei の添字と配列 Uri の行方向の添字 は店コードに対応している。

配列

(1

(1

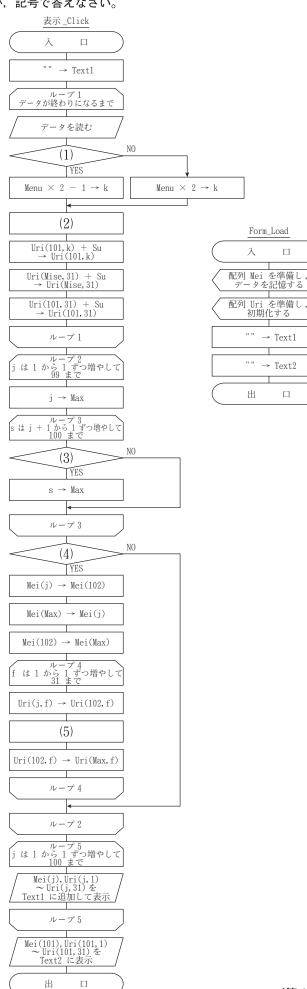
(1

	Mei	Uri	(1)	(2)	~	(29)	(30)	(31)	
(1)	0~0	(1)			~				
(2)	$\triangle \sim \triangle$	(2)			~				
₹	~	7	₹	₹	- }	- }	₹	- }	
(00	\times \sim \times	(100)			~				
01)	合 計	(101)			~				(合計)
02)		(102)			~				
	(店名)		(並盛)	(大盛)	~	(並盛)	(大盛)	(合計)	•
			(牛	丼)	~	(豚	丼)		

(第3図)

- (2) 「表示」ボタンをクリックすると、第1図の売上デー タを読み、第3図の配列 Uri に売上数を次のように集計 する。
 - ① メニューコードは1~15まであり、それぞれ種類コー ドにより1(並盛),2(大盛)に分かれて記録されて いる。なお、配列 Uri の構成は列方向に対して、メニュー ごとに並盛と大盛が連続している。
 - ② 101行目に売上数合計を、31列目に店別売上数合計 を、101行目の31列目にチェーン全体の売上数合計を 集計する。
- (3) 配列 Uri に売上数を集計した後,第3図の配列 Mei と 配列 Uri を店別売上数合計の降順に並べ替える。
- (4) 第2図のText1に配列Meiと配列Uriの内容を追加し て表示する。
- (5) チェーン全体の売上数合計を, 第2図のText2に表示
- (6) データにエラーはなく、操作は正常に行われるものと する。

- 7. Uri(s, 31) < Uri(Max, 31)
- **1.** $j \neq Max$
- ウ. Shu = 1
- \perp . Uri(j, f) \rightarrow Uri(Max, f)
- オ. $Uri(k, Mise) + Su \rightarrow Uri(k, Mise)$
- カ. Shu = 2
- **‡**. Uri(Mise, k) + Su → Uri(Mise, k)
- $\mathbf{\mathcal{D}}$. j = Max
- σ . Uri(Max, f) \rightarrow Uri(j, f)
- \exists . Uri(s, 31) > Uri(Max, 31)



【6】 ある企業では、2つの工場で共通して使用する部品を1か月ごとに一括して発注している。第1図のような部品入出庫ファイルと第2図のような部品在庫マスタファイルを読み、処理条件にしたがって、新部品在庫マスタファイルを作成するとともに、第3図のような発注一覧表を表示するプログラムの空欄をうめなさい。

入出力形式

上段:A工場部品入出庫ファイル(ファイル名:afile.csv) 中段:B工場部品入出庫ファイル(ファイル名:bfile.csv)

下段:併合部品入出庫ファイル(ファイル名:hfile.csv)

日付	区分	部品コード	数量
(MdA)	(KuA)	(BcdA)	(SuryoA)
(MdB)	(KuB)	(BcdB)	(SuryoB)
(MdH)	(KuH)	(BcdH)	(SuryoH)
$\times \times \times \times$	×	$\times \times \times \times$	$\times \times \times \times$

」(第1図)

入出力形式

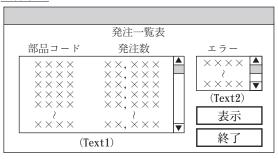
部品在庫マスタファイル(ファイル名:ozai.csv)

新部品在庫マスタファイル(ファイル名:nzai.csv)

部品コード	在庫数	基準在庫数
(BcdM)	(ZaiM)	(KzaiM)
$\times \times \times \times$	$\times \times \times \times \times$	$\times \times \times \times \times$

(第2図)

実行形式



(第3図)

処理条件

- (1) フォームロード時に,第1図のA工場部品入出庫ファイルとB工場部品入出庫ファイルを読み,部品コードの昇順に併合し,併合部品入出庫ファイルを作成する。なお,2つの工場の部品入出庫ファイルは,部品コードの昇順に記録されている。また,同じ部品コードの場合は,A工場部品入出庫ファイルを先に,併合部品入出庫ファイルに書き出す。
- (2) 「表示」ボタンをクリックすると、次の処理を行う。
 - ① 第2図の部品在庫マスタファイルと併合部品入出庫ファイルを読み、新部品在庫マスタファイルを作成する。なお、部品在庫マスタファイルは、部品コードの昇順に記録されており、部品在庫マスタファイルの部品コードに対して、併合部品入出庫ファイルの部品コードは複数件ある場合もあり、ない場合もある。
 - ② 第1図の部品入出庫ファイルの区分は次のとおりであり、 在庫数を更新する。
 - 1:入庫(在庫数に加算する)
 - 2:出庫(在庫数から減算する)
 - ③ 在庫数が基準在庫数より小さくなった場合は、第3図のように部品コードと発注数をText1に追加して表示する。なお、発注数は次の計算式で求める。

発注数 = 基準在庫数 - 在庫数

- ④ 併合部品入出庫ファイルの部品コードが部品在庫マスタファイルにない場合は、エラーとして部品コードを Text2に追加して表示する。
- (3) 部品コード以外のデータにエラーはなく、操作は正常に行われるものとする。

```
Option Explicit
Dim MdA, KuA, BcdA, SuryoA As Long
Dim MdB, KuB, BcdB, SuryoB As Long
Dim MdH, KuH, BcdH, SuryoH As Long
Dim BcdM, ZaiM, KzaiM As Long
Private Sub 終了_Click()
End
End Sub
```

```
Private Sub Form_Load()
  Text1.Text = "": Text2.Text = ""
Open "afile.csv" For Input As #1
Open "bfile.csv" For Input As #2
Open "hfile.csv" For Output As #3
   Text1. Text =
                        Text2. Text = ""
   Call ReadA
   Call ReadB
   Do Until BcdA = 9999 And BcdB = 9999
       f <u>(ア)</u> Then
Write #3, MdB, KuB, BcdB, SuryoB
     Ιf
        Call ReadB
     Flse
        Write #3, MdA, KuA, BcdA, SuryoA
        Call ReadA
     End If
   Loop
   Close #1, #2, #3
End Sub
Private Sub 表示_Click()
Dim OrderM As Long
Text1.Text = "": Text2.Text = ""
Open "hfile.csv" For Input As #3
Open "ozai.csv" For Input As #4
Open "nzai.csv" For Output As #5
   Call ReadH
   Call ReadM
   Do Until
                                (イ)
     Select Case BcdM
        Case Is = BcdH
          ISE 13 _ .
| If <u>KuH = 1 Then</u>
(ウ
          Else
             ZaiM = ZaiM - SuryoH
          End If
          Call ReadH
        Case Is < BcdH
           If ZaiM < KzaiM Then
             OrderM = KzaiM - ZaiM
             End If
          Write #5,
        Call ReadM
Case Else
          (オ)
     End Select
   Loop
   Close #3, #4, #5
End Sub
Private Sub ReadA()
If EOF(1) Then
     BcdA = 9999
   Else
     Input #1, MdA, KuA, BcdA, SuryoA
   End If
End Sub
Private Sub ReadB()
If EOF(2) Then
     BcdB = 9999
   Else
     Input #2, MdB, KuB, BcdB, SuryoB
  End If
End Sub
Private Sub ReadH()
If EOF(3) Then
     BcdH = 9999
     Input #3, MdH, KuH, BcdH, SuryoH
  End If
End Sub
Private Sub ReadM()
If EOF(4) Then
     BcdM = 9999
   Else
     Input #4, BcdM, ZaiM, KzaiM
   End If
```

第3図の表示は見やすく示している。

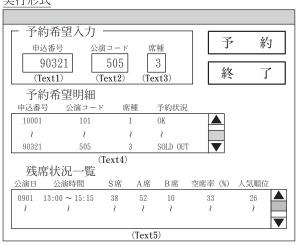
【7】 あるミュージカル公演の予約希望データを入力し、処理条件にしたがって第2図のような予約希望明細と残席状況一覧を表示するプログラムの空欄をうめなさい。

入力形式

公演データ(ファイル名:kouen.csv)

		· modeli, c	
公演コード	公演日	公演時間	
$\times \times \times$	$\times \times \times \times$	\times ~ \times	(第1図)

実行形式



(第2図)

処理条件

- (1) フォームロード時に次の処理を行う。
 - ① 第1図の公演データを読み,第3図の配列にそれぞれ記憶する。この公演は全50回行われ,ペアチケット(2席分)のみが販売される。なお,データは公演コードの昇順に記録されている。

配列



② 第 4 図の配列 Zan の $1 \sim 4$ 列目に、S 席、A 席、B 席のそれぞれの座席数と空席率の初期値である100 を記憶する。なお、配列 MKodo、配列 MTukihi、配列 MJi の添字と配列 Zan の行方向の添字は対応している。

配列

Zan	(1)	(2)	(3)	(4)	(5)	
(1)						
}	}	}	₹	}	}	
(50)						
	(S席)	(A席)	(B席)	(空席率)	(人気順位)	(第4図)

- (2) 第 2 図のように、予約希望データを Text1 \sim Text3 に入力し、「予約」ボタンをクリックすると、次の処理を行う。
 - ① 予約希望入力の席種には、1(S席)、2(A席)、3(B席) のいずれかを入力する。なお、席種は第4図の列方向の添字 と対応している。
 - ② 予約希望入力の公演コードをもとに,第3図の配列 MKodo を探索し,行方向の添字を求め,第4図の配列 Zan の席種ごとに残席数を計算する。
 - ③ 第2図のように、予約希望明細をText4に追加して表示する。 なお、残席が2席以上ある場合はOKを、残席が2席未満の場合はSOLDOUTを予約状況に表示する。
 - ④ 次の計算式により、入力時点までの空席率を求め、第4図の配列 Zanの4列目に記憶する。なお、計算式の残席計とは、第4図の配列 Zanの席種ごとの残席数を集計したものである。

空席率 = 残席計 × 100 ÷ 300 (小数点以下切り捨て)

- ⑤ 空席率の昇順に順位をつけ、配列 Zan の 5 列目に記憶する。 なお、空席率が同じ場合は、同順位とする。
- ⑥ 第2図のように、残席状況一覧を表示する。
- (3) データにエラーはなく、操作は正常に行われるものとする。

```
Option Explicit
Dim MKodo (1 To 50), MTukihi (1 To 50) As Long
Dim MJi(1 To 50) As String
Dim Zan(1 To 50, 1 To 5), j, p As Long
Private Sub Form_Load()
  Text1. Text = "": Text2. Text = "": Text3. Text = ""
Text4. Text = "": Text5. Text = ""
Open "kouen.csv" For Input As #1
  For j = 1 To 50
    Input #1, MKodo(j), MTukihi(j), MJi(j)
For p = 1 To 4
             (ア)
                        = 100
    Next p
  Next i
  Close #1
End Sub
Private Sub 予約_Click()
  Dim YBan, YKodo, YSyu As Long
  Dim Sw, Ue, Sita, m, ZanG, k As Long
  Dim Jyou As String
  Text5. Text =
  YBan = Val(Text1. Text): YKodo = Val(Text2. Text)
  YSyu = Val (Text3. Text)
  Sw = 0: Ue = 51: Sita = 0
  Do Until Sw = 1
    m = Int((Ue + Sita) / 2)
    Select Case YKodo
       Case Is [
         Sw = 1
      Case Is < MKodo(m)
         Ue = m
       Case Else
         Sita = m
    End Select
  Loop
                                Then
    Zan(m, YSyu) = Zan(m, YSyu) - 2
    Jyou = "OK"
  Else
    Jyou = "SOLD OUT"
  End If
  ZanG = Zan(m, 1) + Zan(m, 2) + Zan(m, 3)
  Zan(m, 4) = Int(ZanG * 100 / 300)
  Text4. Text = Text4. Text &
                                            " & _
      Format(YBan, "00000") & "
      Format(YBan, 00000) & & _ _
Format(YKodo, "000") & " " & _
Format(YSyu, "0") & " " & _
Format(Jyou, "&&&&&&&") & Chr(13) & Chr(10)
                                          " & _
  For j = 1 To 50
            (I)
  Next j
  For j = 1 To 49
    k = j + 1
    For p = \boxed{(\pi)}
Select Case Zan(j, 4)
                 <u>(才)</u>
         Case Is < Zan(p, 4)
            Zan(p, 5) = Zan(p, 5) + 1
         Case Is > Zan(p, 4)
            Zan(j, 5) = Zan(j, 5) + 1
       End Select
    Next p
  Next j
  For j = 1 To 50
    Text5.Text = Text5.Text & _
Format(MTukihi(j), "0000") & "
Format(MJi(j), "&&&&&&&&\") & "
         Format(MJi(j),
    For p = 1 To 5
       Text5. Text = Text5. Text &
           Format(Zan(j, p), "##0\overline{}") & "
    Next p
    Text5. Text = Text5. Text & Chr(13) & Chr(10)
  Next j
End Sub
Private Sub 終了_Click()
  End
```

注 第2図の表示は見やすく示してある。

End Sub

主催 點 全国商業高等学校協会

亚成99年度 (第49回) 情報加冊檢定試験プログラミング部門 **第1 類**

平成	22年度	(第43回)	情報処理	里検定試	験プログ	ラミング部門	第1級
			解	答 用	紙		
[1]	1	2	3	4	5		
[2]	1	2	3	4	5		
						l	
[3]	1	2	3	4	5		小計
	•						
•••••	• • • • • • • • • • • • • • • • • • • •	• • • • • • • • • • • • • • • • • • • •	·· (COBOL)	・〔イベント	駆動型BASIC)]	•••••
[4]	(1)	(2)	(3)	(4)	(5)		
【5】	(1)	(2)	(3)	(4)	(5)		
[6]	(7)						
	(イ) (ウ)						
	(<u>x</u>)						
	(3)						
【 7 】	(7)						
	(1)						
	(ウ)						
	(1)						
	(4)						

試験場校名	受験番号	選択	言語
		COBOL	イベント駆動型 BASIC

合	計

(オ) K

主催 點 全国商業高等学校協会

平成22年度(第43回)情報処理検定試験プログラミング部門 第1級

			審	査 基	準	
[1]	1	2	3	4	5	
	サ	カ	ア	コ	ウ	各 2 点 計10点
F 0 Y					_	1
[2]	1	2	3	4	5	
	才	1	ケ	ク	エ	各 2 点 計10点
[3]	1	2	3	4	5	小青十
	ウ	ア	イ	ウ	イ	各 2 点 30 計10点
• • • • • • • • • • • • • • • • • • • •	• • • • • • • • • • • • • • • • • • • •	• • • • • • • • • • • • • • • • • • • •	• • • • • • • • • • • • • • • • • • • •	··· (COBOL)	••••	
				(00202)		
[4]	(1)	(2)	(3)	(4)	(5)	
	ケ	エ	カ	ウ	ア	各 3 点 計15点
[5]	(1)	(2)	(3)	(4)	(5)]
	ウ	丰	コ	1	ケ	各 3 点 計15点
[6]) > B – B (. D			
[0]				- B C D =	9999	
	(イ) O — B C D (ウ) C O M P U	TF O = 7	$\Delta I = \Omega -$	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	SURVO	
		N-R FF				
		RM REAI				
						各4点 計20点
[7]	(7) Z A N (J	P)				
	(1) M – K O D	O (M) =	Y - K O D O			
	(ウ) ZAN (M	Y - SYU	>= 2			
	(I) MOVE	1 TO Z A	AN (J 5)			

各 4 点 計20点

試験場校名	受験番号	選択言語		
		COBOL	イベント駆動型 BASIC	

小 **7**0



主催 點 全国商業高等学校協会

平成22年度(第43回)情報処理検定試験プログラミング部門 第1級

審査基準

			審	査 基	準	
[1]	1	2	3	4	5	
	サ	カ	ア	コ	ウ	各 2 点 計10点
[2]	1	2	3	4	5	1
	オ	1	ケ	ク	エ	各 2 点 計10点
[3]	1	2	3	4	5	小計
101	ウ	ア	1	ウ	1	82点 30
				 ント駆動型BA	CIC)	計10点
				ノ 下極勤空ログ		
[4]	(1)	(2)	(3)	(4)	(5)	
	ケ	エ	カ	ウ	ア	各 3 点 計15点
[5]	(1)	(2)	(3)	(4)	(5)]
	ウ	丰	コ	イ	ケ	各 3 点 計15点
[6]	(7) B c d A	> B c d B				
	(1) B c d M	= 99999	And Bc	dH = 99	9 9	
				о Н		
	(<u>T</u>) B c d M,		KzaiM			
	(オ) C a 1 1	ReadH				各4点 計20点
【 7】	(7) Zan (j	, p)				
		d o (m)			1 1 1 1 1 1	
	(ウ) Z a n (m					
	(土) Zan (j (オ) k To		1			
			,大文字,小文字	 字および=, 演算	 [子の前後の空白	」

試験場校名	受験番号	選択言語	
		COBOL イベント駆動型 BASIC	

小 計

