

2006年 9 月24日実施

平成18年度（第35回）

情報処理検定試験  
〈プログラミング部門〉  
第2級 試験問題

注意事項

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 8 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】【4】は共通問題です。全員解答してください。
5. 【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50分です。

主催 財団法人 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. 2進数の1101と101の和を表す10進数。
2. アナログ音声データをデジタル化して記録するときのファイル形式。
3. 機械語とほぼ1対1に対応した命令を記述するプログラム言語。
4. 静止画を点の集まりとして、圧縮せずに記録するファイル形式。
5. ネットワーク上で他人をよそおうこと。

解答群

ア. BMP	イ. C言語	ウ. 18	エ. MIDI	オ. 暗号化
カ. 36	キ. WAVE	ク. GIF	ケ. なりすまし	コ. アセンブリ言語

【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. セキュリティホール
2. セクタ
3. 電子商取引
4. 産業財産権
5. サブディレクトリ

B群

- ア. ファイルを階層構造で管理するとき、最上位にあるディレクトリ。
- イ. フロッピーディスクやハードディスクにおいて、データの読み書きをする最小単位。
- ウ. 組織内のコンピュータネットワークに対する外部からの不正な侵入を防ぐシステム。
- エ. ソフトウェアの設計不良などによるシステムの安全機能上の欠陥。
- オ. 発明や商品名などの産業上の創作物について登録者を保護する権利。
- カ. インターネットを利用して、商品やサービスの売買などを行うこと。
- キ. フロッピーディスクやハードディスクの記録面にある同心円状の記憶場所。
- ク. ファイルを階層構造で管理するとき、あるディレクトリの下位に作成されるディレクトリ。
- ケ. 小説、音楽、美術、映画などの創作者を登録に関係なく保護する権利。
- コ. 学校や企業といった特定の場所において、1つのソフトウェアを複数のユーザが利用できる契約。

【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. ソフトウェアの不具合を、応急措置で修正するファイル。

ア. アーカイバ                      イ. パッチ                      ウ. アップデート

2. ピンの数が両面合わせて168本あり、64ビットのデータのやり取りが可能な増設用メモリ。

ア. DIMM                      イ. DNS                      ウ. 磁気ディスク装置

3. コンピュータシステムの動作を管理・コントロールするためのプログラム。

ア. 言語プロセッサ                      イ. サービスプログラム                      ウ. 制御プログラム

4. 最大127個の周辺機器をパソコンの電源を入れたまま接続可能で、直ちに使用できるインタフェース規格。

ア. IDE                      イ. USB                      ウ. FTTH

5. 専用の画面上を指やペンで触れることにより入力する装置。

ア. タッチパネル                      イ. スプリック                      ウ. イメージスキャナ

【4】 次の各問いに答えなさい。

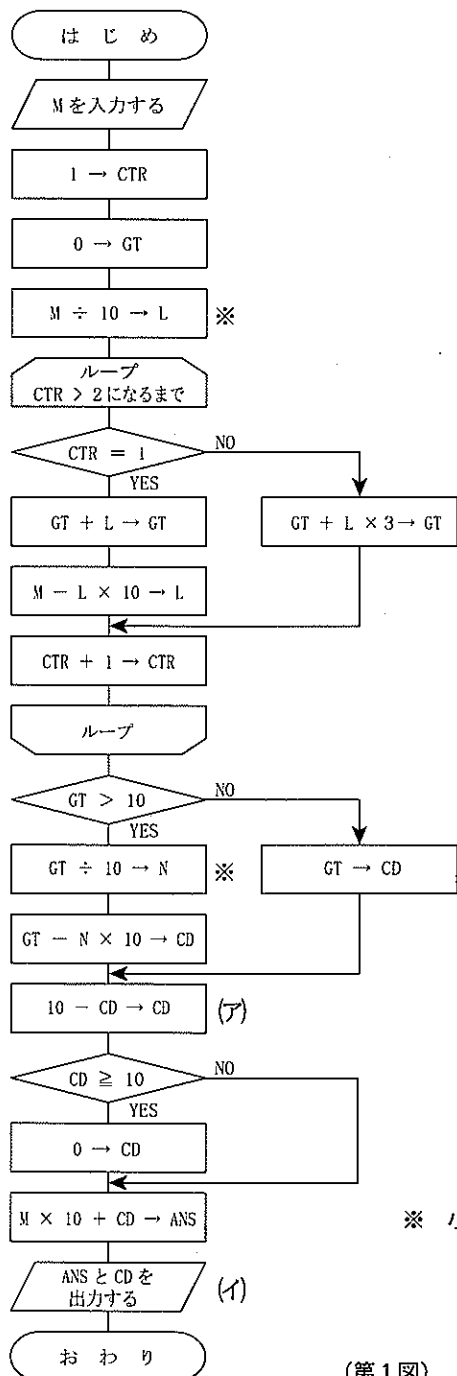
問1. 次の説明文に最も適した答えを解答群から選び、記号で答えなさい。

- (1) プログラムの文法的な誤りや論理的な誤りを探し、その誤りを訂正する作業。
- (2) プログラム言語で記述されたプログラムを、コンパイラなどのソフトウェアによって機械語に変換したプログラム。
- (3) Nの値が5のとき、次の命令を実行したあとのKの値。

＜COBOL選択者＞	＜イベント駆動型BASIC選択者＞
COMPUTE K = N ** 3	K = N ^ 3

解答群		
ア. デバッグ	イ. 15	ウ. 125
エ. 目的プログラム	オ. トレース	カ. 原始プログラム

問2. 第1図の流れ図にしたがって処理するとき、次の(1)~(3)に答えなさい。なお、入力されるMの値は、必ず2けたの整数とする。



- (1) Mの値が12のとき、(ア)を実行した直後のCDの値を答えなさい。
- (2) Mの値が55のとき、(イ)で出力されるANSの値を答えなさい。
- (3) (イ)で出力されるCDの値を説明した文で、正しいものはどれか、ア~ウから選び、記号で答えなさい。
  - ア. CDの値は必ず1けたの整数になる。
  - イ. CDの値は2けたの整数になることもある。
  - ウ. CDの値は必ず5以上の整数になる。

※ 小数点以下切り捨て

(第1図)

# [COBOL選択者のための問題]

2006.09 2-⑧

【5】 第1図のようなハンドボール投げの記録ファイルを読み、処理条件にしたがって第2図のように印字する。プログラムの空欄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

(ファイル名：SPORTS-F, レコード名：SPORTS-R)

生徒番号 (BAN)	1回目 (KAI-1)	2回目 (KAI-2)	3回目 (KAI-3)	(第1図)
××××	××	××	××	

処理条件

- (1) 第1図のファイルを読み、第2図のように生徒番号から記録までを印字する。なお、各生徒の1回目から3回目の中で最も良い結果を記録とする。
- (2) ファイルを読み終えたあと、最高記録の生徒番号とその最高記録を印字する。なお、最高記録とは、全生徒の記録の中で最も良いものである。
- (3) 最後に、平均記録を次の計算式で求めて印字する。  

$$\text{平均記録} = \frac{\text{記録の合計}}{\text{人数}}$$
 (小数点以下切り捨て)
- (4) 入力データにエラーはないものとする。

— 解答群 —

- ア. KAI-2 < KAI-3
- イ. KAI-2 > KAI-3
- ウ. KAI-1
- エ. KAI-2
- オ. MOVE HEIKIN TO S-KIROKU
- カ. MOVE HEIKIN TO M-KIROKU
- キ. COMPUTE NINZU = NINZU + 1
- ク. COMPUTE GOKEI = GOKEI + 1
- ケ. MOVE KIROKU TO MAX-KIROKU
- コ. MOVE GOKEI TO MAX-KIROKU

出力形式

(ファイル名：LIST-F, レコード名：LIST-R)

(ハンドボール投げ記録集計表)				
(生徒番号)	(1回目)	(2回目)	(3回目)	(記録)
1101	25	27	25	27
1102	36	30	32	36
}	}	}	}	}
1140	31	30	29	31
(最高記録の生徒番号)	1102	(最高記録)	36	(平均記録)
			26	

(第2図)

PROCEDURE DIVISION.

HAJIME.

```

OPEN INPUT SPORTS-F OUTPUT LIST-F
MOVE ZERO TO END-SW MAX-KIROKU GOKEI NINZU
PERFORM UNTIL END-SW = 1
  READ SPORTS-F
  AT END
    MOVE 1 TO END-SW
  NOT AT END
    IF KAI-1 > KAI-2
      THEN
        IF KAI-1 > KAI-3
          THEN MOVE (1) TO KIROKU
          ELSE MOVE KAI-3 TO KIROKU
        END-IF
      ELSE
        IF (2)
          THEN MOVE KAI-2 TO KIROKU
          ELSE MOVE KAI-3 TO KIROKU
        END-IF
      END-IF
    END-IF
    MOVE BAN TO M-BAN
    MOVE KAI-1 TO M-KAI-1
    MOVE KAI-2 TO M-KAI-2
    MOVE KAI-3 TO M-KAI-3
    MOVE KIROKU TO M-KIROKU
    WRITE LIST-R FROM MEISAI-G AFTER 1
    IF KIROKU > MAX-KIROKU
      THEN MOVE BAN TO MAX-BAN
      (3)
    END-IF
    COMPUTE GOKEI = GOKEI + KIROKU
    (4)
  END-READ
END-PERFORM
MOVE MAX-BAN TO S-BANGO
MOVE MAX-KIROKU TO S-KIROKU
WRITE LIST-R FROM SYUKEI-G AFTER 1
INITIALIZE SYUKEI-G
COMPUTE HEIKIN = GOKEI / NINZU
(5)
WRITE LIST-R FROM SYUKEI-G AFTER 1
CLOSE SPORTS-F LIST-F
STOP RUN.
    
```

WORKING-STORAGE SECTION.

```

01 END-SW          PIC 9(01).
01 MAX-BAN         PIC X(04).
01 MAX-KIROKU     PIC 9(02).
01 GOKEI          PIC 9(05).
01 NINZU          PIC 9(03).
01 KIROKU         PIC 9(02).
01 HEIKIN         PIC 9(02).
01 MEISAI-G.
02                PIC X(04) VALUE SPACE.
02 M-BAN         PIC X(04).
02                PIC X(07) VALUE SPACE.
02 M-KAI-1       PIC Z9.
02                PIC X(07) VALUE SPACE.
02 M-KAI-2       PIC Z9.
02                PIC X(07) VALUE SPACE.
02 M-KAI-3       PIC Z9.
02                PIC X(08) VALUE SPACE.
02 M-KIROKU      PIC Z9.
01 SYUKEI-G.
02                PIC X(22) VALUE SPACE.
02 S-BANGO       PIC X(04).
02                PIC X(17) VALUE SPACE.
02 S-KIROKU      PIC Z9.
    
```

# [COBOL選択者のための問題]

2006.09 2-④

【6】 第1図のようなある駐車場で1日の利用状況ファイルを読み、処理条件にしたがって第2図のように印字する。第4図の流れ図の(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

入庫時刻		出庫時刻	
時	分	時	分
(N-JI)	(N-FUN)	(S-JI)	(S-FUN)
××	××	××	××

(第1図)

### 処理条件

- 第1図の入庫時刻と出庫時刻には、0時0分から23時59分までの時刻が「時」と「分」に分かれて記録されている。なお、複数の日付にまたがるデータはない。
- 第1図のファイルを読み、次の手順で駐車時間を求め、第2図のように印字する。
  - 入庫時刻と出庫時刻をそれぞれ「分」に換算する。
  - 換算した出庫時刻から入庫時刻を差し引いて駐車時間(分)を求める。
  - 駐車時間(分)から、「時間」と「分」をそれぞれ求める。
- 駐車時間をもとに第3図のテーブルDAISU-Tを利用して、駐車時間区分ごとの台数を集計する。なお、駐車時間区分とは第2図のように駐車時間を1時間ごとに分けたものである。ただし、駐車時間が5時間以上の場合にはDAISU(6)にまとめて集計する。

### テーブル DAISU-T

(1)	~	(5)	(6)
DAISU	~		
(0時間0分~0時間59分の台数)	~	(4時間0分~4時間59分の台数)	(5時間以上の台数)

(第3図)

- ファイルを読み終えたあと、第2図のように駐車時間区分ごとの台数を順に印字する。
- 最後に、駐車場を利用した台数の合計を印字し、処理を終了する。
- 入力データにエラーはないものとする。

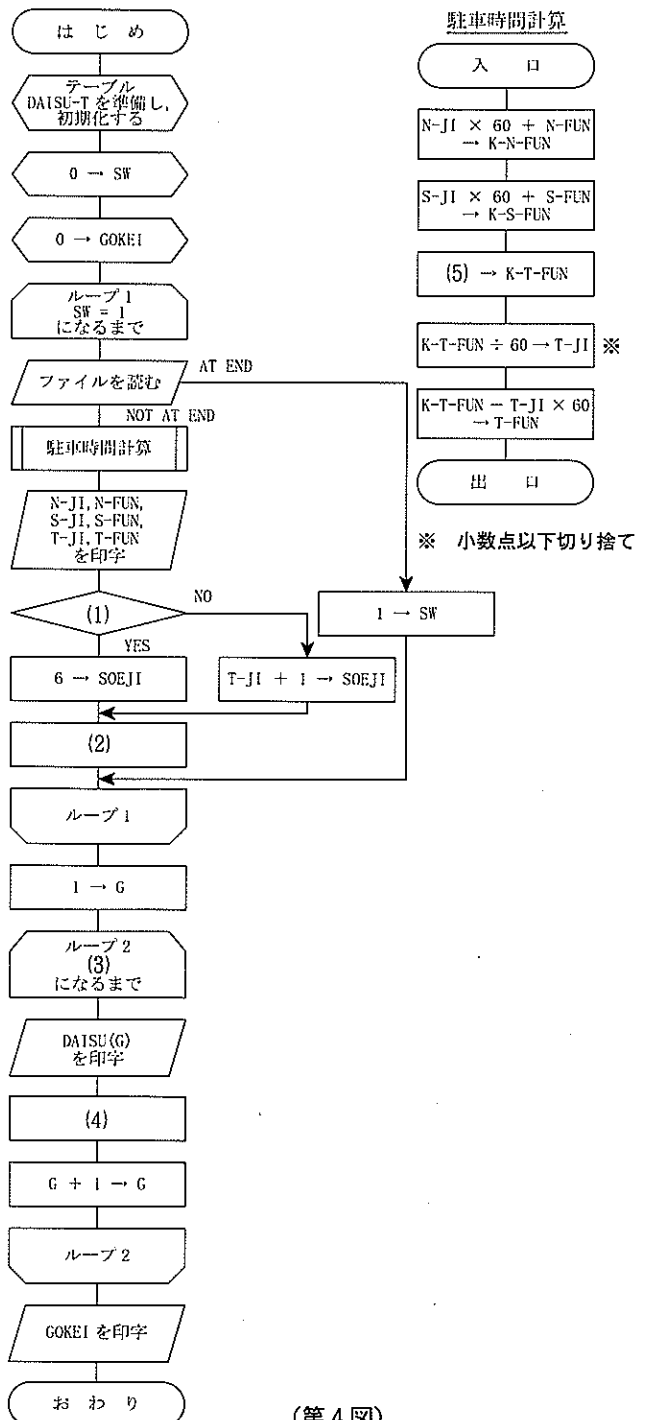
### 解答群

- ア.  $T-JI \geq 5$
- イ.  $GOKEI + DAISU(G) \rightarrow GOKEI$
- ウ.  $DAISU(G) + 1 \rightarrow DAISU(G)$
- エ.  $G > 6$
- オ.  $S-FUN - N-FUN$
- カ.  $T-FUN \geq 5$
- キ.  $DAISU(SOEJI) + 1 \rightarrow DAISU(SOEJI)$
- ク.  $GOKEI + 1 \rightarrow GOKEI$
- ケ.  $K-S-FUN - K-N-FUN$
- コ.  $G \geq 6$

### 出力形式

(駐車時間集計表)					
(入庫時刻)		(出庫時刻)		(駐車時間)	
(時)	(分)	(時)	(分)	(時間)	(分)
0	15	1	20	1	5
1	20	4	10	2	50
?	?	?	?	?	?
22	30	22	50	0	20
(駐車時間区分)				(台数)	
(0時間0分 ~ 0時間59分)				4	
(1時間0分 ~ 1時間59分)				9	
(2時間0分 ~ 2時間59分)				14	
(3時間0分 ~ 3時間59分)				8	
(4時間0分 ~ 4時間59分)				6	
(5時間以上)				4	
(合計)				45	

(第2図)



(第4図)

# [COBOL選択者のための問題]

2006.09 2-⑤

【7】 第1図のようなある製パン工場の注文ファイルを読み、処理条件にしたがって第2図のような1週間分の小麦粉必要量計算表を印字するプログラムの空欄をうめなさい。

### 入力形式

(ファイル名: CHUMON-F, レコード名: CHUMON-R)

伝票番号 (DENCD)	商品コード (HINCD)	注文数 (SURYO)
××××	××××	×××

(第1図)

### 処理条件

(1) 商品は全部で10種類あり、商品コードと商品1個あたりの必要量(単位:g)は、それぞれ第3図のテーブルHIN-TBL, KOMUGI-TBLに記憶されている。なお、HIN(11)は印字のみに利用する。

#### テーブル HIN-TBL

HIN	SP1K	FPHS	RP08	~	MP2S	合計
(1)	(2)	(3)	~	(10)	(11)	

#### テーブル KOMUGI-TBL

KOM	100	150	20	~	40
(1)	(2)	(3)	~	(10)	

(第3図)

(2) 第1図のファイルを読み、商品コードをもとに第3図のテーブルHIN-TBLを探索し、第4図のテーブルSHUKEI-TBLに、商品ごとの注文数を集計する。また、全体の合計はSHUKEI(11)に集計する。なお、第3図と第4図のすべてのテーブルは添字で対応している。

#### テーブル SHUKEI-TBL

SHUKEI	(1)	(2)	(3)	~	(10)	(11)

(第4図)

(3) ファイルを読み終えたあと、第3図のテーブルKOMUGI-TBLに記憶されている商品1個あたりの必要量をもとに、商品ごとの必要量を次の計算式で求め、第2図のように印字する。ただし、印字は注文のあった商品のみとする。

商品ごとの必要量 = 商品1個あたりの必要量 × 注文数計

(4) 最後に、注文数計の合計と必要量の合計を印字し、処理を終了する。

(5) 入力データにエラーはないものとする。

### 出力形式

(ファイル名: PRINT-F, レコード名: PRINT-R)

(商品コード)	(小麦粉必要量計算表) (注文数計)	(必要量)
SP1K	300	30,000
RP08	450	9,000
}	}	}
MP2S	900	36,000
合計	5,100	292,500

(第2図)

### WORKING-STORAGE SECTION.

```

WORKING-STORAGE SECTION.
    >
01 HITSU          PIC 9(08).
01 HITSU-KEI     PIC 9(09).
01 MEISAI-GYO.
    02             PIC X(05) VALUE SPACE.
    02 M-HIN      PIC X(04).
    02             PIC X(10) VALUE SPACE.
    02 M-CHUKEI  PIC ZZZ,ZZ9.
    02             PIC X(10) VALUE SPACE.
    02 M-HITSU   PIC ZZZ,ZZZ,ZZ9.

PROCEDURE DIVISION.
ST. OPEN INPUT CHUMON-F OUTPUT PRINT-F
    INITIALIZE E-FLG (ア) HITSU-KEI
    PERFORM UNTIL E-FLG = 1
        READ CHUMON-F
        AT END
            MOVE 1 TO E-FLG
        NOT AT END
            MOVE 1 TO N
            PERFORM UNTIL (イ)
                COMPUTE N = N + 1
            END-PERFORM
            COMPUTE SHUKEI(N) = SHUKEI(N) + SURYO
            COMPUTE (ウ)
        END-READ
    END-PERFORM
    MOVE 1 TO K
    PERFORM UNTIL K > 11
        IF SHUKEI(K) NOT = 0
            THEN
                MOVE HIN(K)      TO M-HIN
                MOVE SHUKEI(K)   TO M-CHUKEI
                IF K <= 10
                    THEN
                        COMPUTE HITSU = KOM(K) * SHUKEI(K)
                        MOVE (エ) TO M-HITSU
                        COMPUTE HITSU-KEI = HITSU-KEI + HITSU
                    ELSE
                        MOVE HITSU-KEI TO M-HITSU
                END-IF
            WRITE PRINT-R FROM MEISAI-GYO AFTER 1
        END-IF
        (オ)
    END-PERFORM
CLOSE CHUMON-F PRINT-F
STOP RUN.
    
```

# [イベント駆動型BASIC選択者のための問題]

2006.09 2-⑥

【5】 ハンドボール投げの記録データを入力し、処理条件にしたがって第2図の実行形式のように表示する。プログラムの空欄にあてはまる答えを解答群から選び、記号で答えなさい。

フォーム

(第1図)

実行形式

(第2図)

### 処理条件

(1) 第2図のように生徒番号と1回目から3回目までのデータを入力し、「実行」ボタンをクリックすると次の処理を行う。

- ① 1回目から3回目の中で、最も良い結果を記録に表示する。
- ② 記録が最高記録より良い場合は、最高記録の生徒番号とその最高記録を表示し直す。なお、最高記録とは、全生徒の記録の中で最も良いものである。
- ③ 平均記録は、次の計算式で求めて表示する。

$$\text{平均記録} = \frac{\text{記録の合計}}{\text{人数}}$$

(小数点以下切り捨て)

(2) 表示に関する設定は、第2図の実行形式のとおりとする。また、入力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。

### 解答群

- ア. Kai2 < Kai3
- イ. Kai2 > Kai3
- ウ. Kai1
- エ. Kai2
- オ. Label4.Caption = Str(Heikin)
- カ. Label4.Caption = Str(Gokei)
- キ. Ninzu = Ninzu + 1
- ク. Gokei = Gokei + 1
- ケ. Saikou = Kiroku
- コ. Saikou = Gokei

```
Option Explicit
Dim Gokei, Ninzu, Saikou As Integer
```

```
Private Sub Form_Load()
    Call Syokyo
    Label2.Caption = ""
    Label3.Caption = ""
    Label4.Caption = ""
    Gokei = 0
    Ninzu = 0
    Saikou = 0
End Sub
```

Private Sub Command1\_Click() (実行)

```
Dim Kai1, Kai2, Kai3 As Integer
Dim Kiroku, Heikin As Integer
Dim Bango As String
Kai1 = Val(Text1.Text)
Kai2 = Val(Text2.Text)
Kai3 = Val(Text3.Text)
Bango = Text4.Text
If Kai1 > Kai2 Then
    If Kai1 > Kai3 Then
        Kiroku = (1)
    Else
        Kiroku = Kai3
    End If
Else
    If (2) Then
        Kiroku = Kai2
    Else
        Kiroku = Kai3
    End If
End If
Label1.Caption = Str(Kiroku)
If Kiroku > Saikou Then
    (3)
    Label2.Caption = Bango
    Label3.Caption = Str(Saikou)
End If
Gokei = Gokei + Kiroku
(4)
Heikin = Int(Gokei / Ninzu)
(5)
End Sub
```

Private Sub Command2\_Click() (クリア)

```
Call Syokyo
End Sub
```

Private Sub Command3\_Click() (終了)

```
End
End Sub
```

Private Sub Syokyo() (消去)

```
Text1.Text = ""
}
Text4.Text = ""
Label1.Caption = ""
End Sub
```

# [イベント駆動型BASIC選択者のための問題]

2006.09 2-⑦

【6】 第1図のようなある駐車場で1日の利用状況データを読み、処理条件にしたがって第3図のように表示する。

第5図の流れ図の(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

入庫時刻		出庫時刻	
時 (NyuJi)	分 (NyuFun)	時 (SyuJi)	分 (SyuFun)
××	××	××	××

(第1図)

処理条件

(1) 第1図の入庫時刻と出庫時刻には、0時0分から23時59分までの時刻が「時」と「分」に分かれて記録されている。なお、複数の日付にまたがるデータはない。

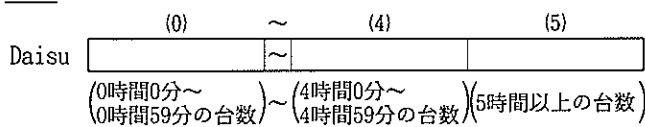
(2) 「表示」ボタンをクリックすると次の処理を行い、第3図のように表示する。

① 第1図のデータを読み、次の手順で駐車時間を求め、Label1に入庫時刻から駐車時間までを表示する。

- ・ 入庫時刻と出庫時刻を「分」に換算する。
- ・ 換算した出庫時刻から入庫時刻を差し引いて駐車時間(分)を求める。
- ・ 駐車時間(分)から、「時間」と「分」をそれぞれ求める。

② 駐車時間をもとに第4図の配列Daisuを利用して、駐車時間区分ごとの台数を集計する。なお、駐車時間区分とは第3図のように駐車時間を1時間ごとに分けたものである。ただし、駐車時間が5時間以上の場合はDaisu(5)にまとめて集計する。

配列



(第4図)

③ データを読み終えたあと、Label2に駐車時間区分ごとの台数を、順に表示する。

④ 最後に、Label3に駐車場を利用した台数の合計を表示する。

(3) 表示に関する設定は、第3図の実行形式のとおりとする。また、入力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。

解答群

- ア.  $TyuJi \geq 5$
- イ.  $Gokei + Daisu(g) \rightarrow Gokei$
- ウ.  $Daisu(g) + 1 \rightarrow Daisu(g)$
- エ.  $g > 5$
- オ.  $SyuFun - NyuFun$
- カ.  $TyuFun \geq 5$
- キ.  $Daisu(TyuJi) + 1 \rightarrow Daisu(TyuJi)$
- ク.  $Gokei + 1 \rightarrow Gokei$
- ケ.  $KanSyuFun - KanNyuFun$
- コ.  $g > 0$

フォーム

駐車時間集計表		
入庫時刻 時 分	出庫時刻 時 分	駐車時間 時間 分
Label1		
駐車時間区分		台数
0時間0分 ~ 0時間59分		Label2
1時間0分 ~ 1時間59分		
2時間0分 ~ 2時間59分		
3時間0分 ~ 3時間59分		
4時間0分 ~ 4時間59分		
5時間以上		4
合計		Label3

表示 終了

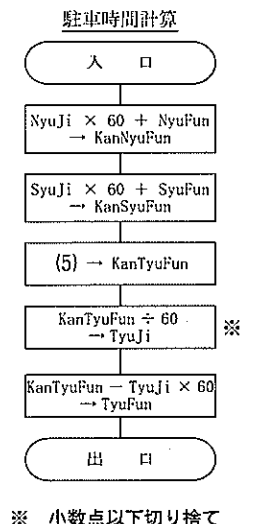
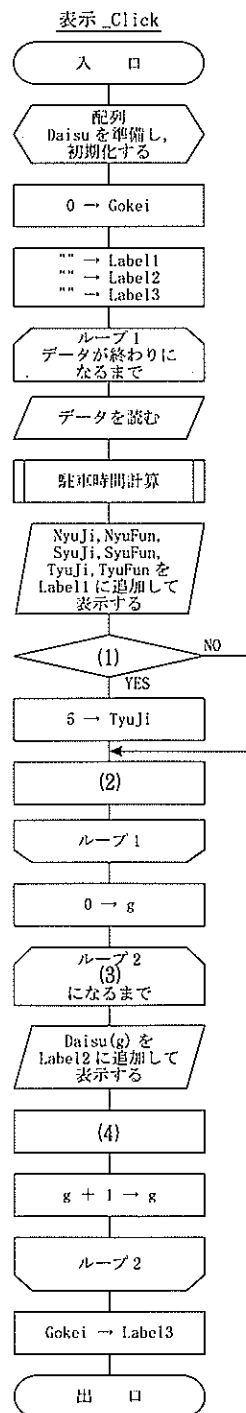
(第2図)

実行形式

駐車時間集計表		
入庫時刻 時 分	出庫時刻 時 分	駐車時間 時間 分
00 15	01 20	01 05
01 20	04 10	02 50
?	?	?
22 30	22 50	00 20
駐車時間区分		台数
0時間0分 ~ 0時間59分		4
1時間0分 ~ 1時間59分		9
2時間0分 ~ 2時間59分		14
3時間0分 ~ 3時間59分		8
4時間0分 ~ 4時間59分		6
5時間以上		4
合計		49

表示 終了

(第3図)



(第5図)

【7】 ある製パン工場での注文データを入力し、処理条件にしたがって第2図のような小麦粉必要量計算表を表示するプログラムの空欄をうめなさい。

フォーム

(第1図)

実行形式

(第2図)

処理条件

- (1) 商品は全部で10種類あり、商品コードと商品1個あたりの必要量(単位:g)は、それぞれ第3図の配列Hin, Komugiへ、フォームロード時に記憶される。なお、Hin(11)は表示のみに利用する。

配列

Hin	SP1K	FPHS	RP08	~	MP2S	合計
	(1)	(2)	(3)	~	(10)	(11)

Komugi	100	150	20	~	40
	(1)	(2)	(3)	~	(10)

(第3図)

- (2) 第2図の商品コードと注文数を入力し、「計算と集計」ボタンをクリックすると、次の手順で処理を行う。

- ① 商品コードをもとに第3図の配列Hinを探し、第4図の配列Shukeiに、商品ごとの注文数を集計する。また、全体の合計はShukei(11)に集計する。なお、第3図と第4図のすべての配列は添字で対応している。

配列

Shukei				~		
	(1)	(2)	(3)	~	(10)	(11)

(第4図)

- ② 第3図の配列Komugiに記憶されている商品1個あたりの必要量をもとに、商品ごとの必要量を次の計算式で求め、商品コード、注文数計、必要量を表示する。ただし、表示は注文のあった商品のみとする。

$$\text{商品ごとの必要量} = \text{商品1個あたりの必要量} \times \text{注文数計}$$

- ③ 最後に、注文数計の合計と必要量の合計を表示する。

- (3) 表示に関する設定は、第2図の実行形式のとおりとする。また、入力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 n As Integer
Dim Hin(1 To 11) As String
Dim Komugi(1 To 10) As Integer
Dim Shukei(1 To 11) As Long
```

```
Private Sub Form_Load()
    Call Syokyo
    Hin(1) = "SP1K"
}
Hin(11) = "合計"
Komugi(1) = 100
}
Komugi(10) = 40
For n = 1 To 11
    (ア)
Next n
End Sub
```

```
Private Sub Command1_Click() (計算と集計)
    Dim Hincd As String
    Dim k, Suroyo As Integer
    Dim Hitsu, Hitsuken As Long
    Hitsuken = 0
    Hincd = Text1.Text
    Suroyo = Val(Text2.Text)
    n = 1
    Do Until (イ)
        n = n + 1
    Loop
    Shukei(n) = Shukei(n) + Suroyo
    Shukei(11) = (ウ)
    Call Syokyo
    k = 1
    Do Until k > 11
        If Shukei(k) <> 0 Then
            Label1.Caption = Label1.Caption & _
                Hin(k) & _
                Chr(13) & Chr(10)
            Label2.Caption = Label2.Caption & _
                Format(Shukei(k), "#,###") & _
                Chr(13) & Chr(10)
            If k <= 10 Then
                Hitsu = Shukei(k) * Komugi(k)
                Label3.Caption = Label3.Caption & _
                    Format((エ), "###,###") & _
                    Chr(13) & Chr(10)
                Hitsuken = Hitsuken + Hitsu
            Else
                Label3.Caption = Label3.Caption & _
                    Format(Hitsuken, "###,###") & _
                    Chr(13) & Chr(10)
            End If
        End If
        k = k + 1
    Loop
End Sub (オ)
```

```
Private Sub Command2_Click() (終了)
    End
End Sub
```

```
Private Sub Syokyo() (消去)
    Text1.Text = ""
    Text2.Text = ""
    Label1.Caption = ""
    Label2.Caption = ""
    Label3.Caption = ""
End Sub
```

(注) Chr(13) & Chr(10)は改行のための制御文字を表す。

主催 財団法人 全国商業高等学校協会

平成18年度 (第35回) 情報処理検定試験プログラミング部門 第2級  
解 答 用 紙

【1】

1	2	3	4	5

【2】

1	2	3	4	5

【3】

1	2	3	4	5

【4】

問 1			問 2		
(1)	(2)	(3)	(1)	(2)	(3)

小 計

..... 【COBOL】 ・ 【イベント駆動型BASIC】 .....

【5】

(1)	(2)	(3)	(4)	(5)

【6】

(1)	(2)	(3)	(4)	(5)

【7】

(ア)	
(イ)	
(ウ)	
(エ)	
(オ)	

試験場校名	受験番号	選 択 言 語	
		COBOL	イベント駆動型 BASIC

小 計

合 計

選択言語を  で囲むこと

主催 財団法人 全国商業高等学校協会

平成18年度 (第35回) 情報処理検定試験プログラミング部門 第2級

審査基準

【1】

1	2	3	4	5
ウ	キ	コ	ア	ケ

各2点  
計10点

【2】

1	2	3	4	5
エ	イ	カ	オ	ク

各2点  
計10点

【3】

1	2	3	4	5
イ	ア	ウ	イ	ア

各2点  
計10点

【4】

問1			問2		
(1)	(2)	(3)	(1)	(2)	(3)
ア	エ	ウ	3	550	ア

問1各2点  
問2各3点  
計15点

小計
45

[COBOL]

【5】

(1)	(2)	(3)	(4)	(5)
ウ	イ	ケ	キ	オ

各3点  
計15点

【6】

(1)	(2)	(3)	(4)	(5)
ア	キ	エ	イ	ケ

各4点  
計20点

【7】

(ア)	SHUKEI-TBL
(イ)	HIN(N) = HINC D
(ウ)	SHUKEI(11) = SHUKEI(11) + SURYO
(エ)	HITSU
(オ)	COMPUTE K = K + 1

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

選択言語を  で囲むこと

小計
55

合計
100

平成18年度 (第35回) 情報処理検定試験プログラミング部門 第2級  
審査基準

【1】

1	2	3	4	5
ウ	キ	コ	ア	ケ

各2点  
計10点

【2】

1	2	3	4	5
エ	イ	カ	オ	ク

各2点  
計10点

【3】

1	2	3	4	5
イ	ア	ウ	イ	ア

各2点  
計10点

【4】

問1			問2		
(1)	(2)	(3)	(1)	(2)	(3)
ア	エ	ウ	3	550	ア

問1各2点  
問2各3点  
計15点

小計
45

..... 【イベント駆動型BASIC】 .....

【5】

(1)	(2)	(3)	(4)	(5)
ウ	イ	ケ	キ	オ

各3点  
計15点

【6】

(1)	(2)	(3)	(4)	(5)
ア	キ	エ	イ	ケ

各4点  
計20点

【7】

(ア)	Shukei (n) = 0
(イ)	Hin (n) = Hincd
(ウ)	Shukei (11) + Suryo
(エ)	Hitsu
(オ)	k = k + 1

(注) 大文字, 小文字および空白は問わない。

各4点 計20点

試験場校名	受験番号	選択言語
		COBOL <input checked="" type="radio"/> イベント駆動型 BASIC

選択言語を  で開むこと

小計
55

合計
100