

2007年 9 月 23 日実施

平成19年度（第37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

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 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。全員解答してください。
5. 【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。

[COBOL選択者のための問題]

2007.09 1-②

【4】 第1図のようなあるバス路線の1日分の乗降ファイルを読み、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。

入力形式

便番号 (BAN)	停留所コード (COD)	乗車数 (JYS)	降車数 (KOS)
××	××××	××	××

(第1図)

出力形式

(乗降数集計表)		
(停留所名)	(乗車数)	(降車数)
駅前	139	0
本町一丁目	56	5
文化会館前	105	87
南小学校前	66	51
}	}	}
(合計)	2,985	2,985

(第2図)

処理条件

- 第1図の乗降ファイルは、停留所コードの昇順に記録されている。また、乗車数と降車数がともにゼロのレコードは、ファイルに記録されていない。
- 乗降ファイルを読み、停留所ごとの乗車数と降車数を集計して印字する。ただし、乗降ファイルにレコードが1件もない停留所は印字しない。
- 停留所名は、停留所コードから第3図のテーブル TEIRYU-TBL を探索して求める。なお、テーブル TEIRYU-TBL には、停留所コードの昇順にデータが記憶されている。

テーブル TEIRYU-TBL

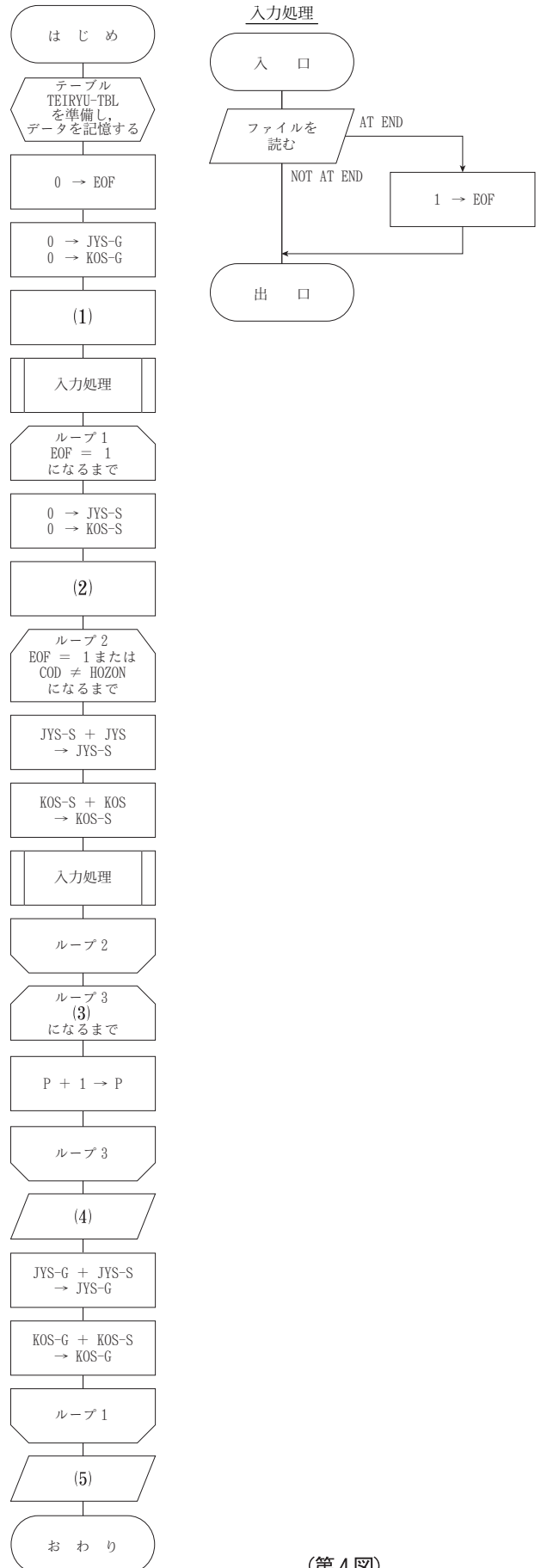
	COD-T	NM-T
(1)	××××	××~××
(2)	××××	××~××
(3)	××××	××~××
(4)	××××	××~××
(5)	××××	××~××
}	}	}
(50)	××××	××~××

(停留所コード) (停留所名) (第3図)

- データが終わりになったら、全体の合計を印字する。
- データにエラーはないものとする。

解答群

- ア. COD → HOZON
- イ. JYS-S, KOS-S を印字
- ウ. NM-T(HOZON), JYS-S, KOS-S を印字
- エ. COD-T(P) = HOZON
- オ. JYS-G, KOS-G を印字
- カ. BAN → HOZON
- キ. 1 → P
- ク. 0 → P
- ケ. NM-T(P), JYS-S, KOS-S を印字
- コ. COD-T(P) = COD



(第4図)

[COBOL選択者のための問題]

【5】 第1図のようなある知事選挙の地区別得票数を記録した開票ファイルを読み、処理条件にしたがって、第2図のような地区別得票一覧表を印字したい。第7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

地区番号 (TIBAN) ××	候補者番号 (KOBAN) ××	得票数 (SU) ××××××
-----------------------	------------------------	-----------------------

(第1図)

出力形式

(地区別得票一覧表)			
(候補者番号)	(候補者名)	(総得票数)	(総得票数順位)
8	佐藤△△	1,861,734	1
	(地区名)	(得票数)	(割合)
?	〇〇市	43,971	27.8
?	?	?	?

(第2図)

処理条件

(1) 第3図のテーブル KU-TAB には、地区名が記憶されている。なお、地区番号は1~30までであり、テーブル KU-TAB と添字で対応している。

テーブル KU-TAB

KU	〇〇市	△△市	~	〇〇町	△△町
	(1)	(2)	~	(29)	(30)

(第3図)

(2) 第4図のテーブル BAN-TAB に候補者番号が、第5図のテーブル NA-TAB に候補者名が記憶されている。なお、候補者番号は1~20まで順に記憶されており、テーブル BAN-TAB の記憶データはテーブル NA-TAB の添字と対応している。

テーブル BAN-TAB

BAN	1	2	~	19	20
	(1)	(2)	~	(19)	(20)

(第4図)

テーブル NA-TAB

NA	山川〇〇	宇野△△	~	松田□□	伊藤◇◇
	(1)	(2)	~	(19)	(20)

(第5図)

(3) 第1図の開票ファイルを読み、第6図のテーブル SK-TAB を利用して、各候補者の得票数を地区ごとに入力する。なお、行方向の添字は候補者番号と対応しており、列方向の添字は地区番号と対応している。

さらに、21行目に地区の得票数計を集計し、31列目に各候補者の総得票数を集計する。

テーブル SK-TAB

SK	(1)	(2)	~	(30)	(31)
(1)					
(20)					
(21)					
(22)					
	(地区1)	(地区2)	~	(地区30)	(総得票数)

(第6図)

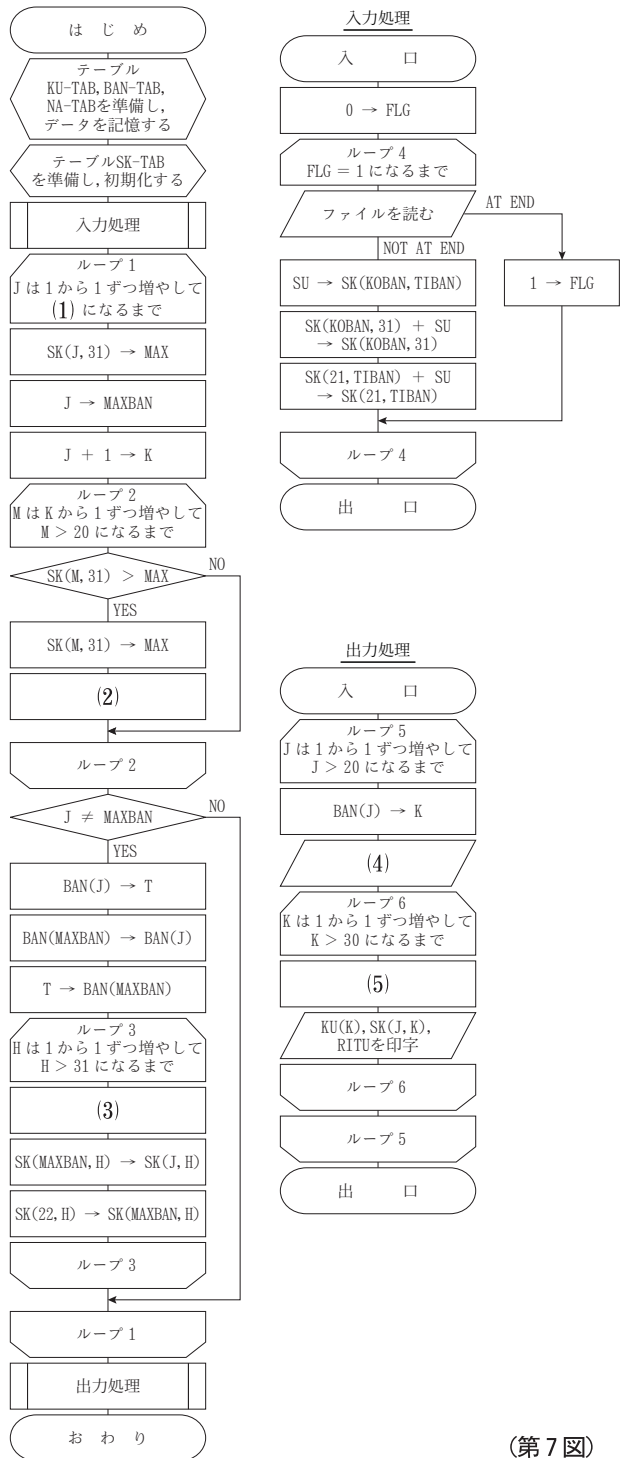
(4) 第4図のテーブル BAN-TAB と第6図のテーブル SK-TAB のデータを、総得票数の降順に並べ替える。ただし、同順位はないものとする。

(5) 第2図のように、次の印字を行う。

- ① 総得票数の降順に、候補者番号、候補者名、総得票数、総得票数順位を印字する。
- ② 地区番号順に、候補者の地区名、得票数、地区の得票数計に対する得票数の割合を印字する。なお、割合は次の例のように求める。

$$\text{例 割合} = \text{SK}(1, 1) \times 100 \div \text{SK}(21, 1)$$

(6) データにエラーはないものとする。



解答群

- ア. K, NA(K), SK(J, 31), J を印字
- イ. J = 19
- ウ. M -> MAXBAN
- エ. SK(K, J) × 100 ÷ SK(21, J) -> RITU
- オ. J = 20
- カ. K, NA(K), SK(K, 31), J を印字
- キ. SK(J, H) -> SK(22, H)
- ク. BAN(M) -> MAXBAN
- ケ. SK(22, H) -> SK(J, H)
- コ. SK(J, K) × 100 ÷ SK(21, K) -> RITU

[COBOL選択者のための問題]

2007.09 1-④

- 【6】 あるスイミングスクールの運営会社では、2校の会員利用時間を1か月ごとに一括して集計している。第1図の各校の会員ファイルを読み、処理条件にしたがって第2図の会員マスタファイルを更新するプログラムの空欄をうめなさい。

入出力形式

上段：A校ファイル（ファイル名：A-KAIIN-F、レコード名：A-KAIIN-R）
 中段：B校ファイル（ファイル名：B-KAIIN-F、レコード名：B-KAIIN-R）
 下段：併合ファイル（ファイル名：H-KAIIN-F、レコード名：H-KAIIN-R）

会員コード	会員名	利用時間
(A-CO)	(A-NA)	(A-JIKAN)
(B-CO)	(B-NA)	(B-JIKAN)
(H-CO)	(H-NA)	(H-JIKAN)
×××	×～×	×～×

(第1図)

入出力形式

上段：会員マスタファイル
 （ファイル名：M-KAIIN-F、レコード名：M-KAIIN-R）

下段：新会員マスタファイル
 （ファイル名：N-KAIIN-F、レコード名：N-KAIIN-R）

会員コード	会員名	時間累計
(M-CO)	(M-NA)	(M-JIKAN)
(N-CO)	(N-NA)	(N-JIKAN)
×××	×～×	×～×

(第2図)

処理条件

- (1) A校ファイル、B校ファイル、会員マスタファイルは、会員コードの昇順に記録されている。なお、A校ファイルとB校ファイルに同じ会員コードはなく、会員マスタファイルの会員コードに対して1件あるかない場合もある。
- (2) A校ファイルとB校ファイルを会員コードの昇順に併合し、併合ファイルを作成する。
- (3) 作成された併合ファイルで会員マスタファイルを更新し、新会員マスタファイルを作成する。なお、更新内容は次のとおりである。
 - ・ 併合ファイルの会員コードが、会員マスタファイルにない場合は、入会を示しており、新会員マスタファイルに追加する。
 - ・ 併合ファイルの会員コードが、会員マスタファイルにある場合は、併合ファイルの利用時間を、会員マスタファイルの時間累計に加算する。なお、併合ファイルの利用時間が999の場合は退会を示しており、新会員マスタファイルに出力しない。
- (4) 入力データにエラーはないものとする。

PROCEDURE DIVISION.

```

ST. OPEN INPUT A-KAIIN-F B-KAIIN-F OUTPUT H-KAIIN-F
    PERFORM R1 THRU R2
    PERFORM UNTIL A-CO = 999 AND B-CO = 999
        EVALUATE TRUE
            WHEN  (ア)
                WRITE H-KAIIN-R FROM A-KAIIN-R
                PERFORM R1
            WHEN OTHER
                WRITE H-KAIIN-R FROM B-KAIIN-R
                PERFORM R2
        END-EVALUATE
    END-PERFORM
    CLOSE A-KAIIN-F B-KAIIN-F H-KAIIN-F
    OPEN INPUT M-KAIIN-F H-KAIIN-F OUTPUT N-KAIIN-F
    PERFORM R3 THRU R4
    PERFORM UNTIL  (イ)
        EVALUATE TRUE
            WHEN M-CO < H-CO
                 (ウ)
                PERFORM R3
            WHEN M-CO > H-CO
                WRITE N-KAIIN-R FROM H-KAIIN-R
                PERFORM R4
            WHEN OTHER
                IF  (エ)
                    THEN COMPUTE M-JIKAN = M-JIKAN + H-JIKAN
                    WRITE N-KAIIN-R FROM M-KAIIN-R
                END-IF
                 (オ)
        END-EVALUATE
    END-PERFORM
    CLOSE M-KAIIN-F H-KAIIN-F N-KAIIN-F
    STOP RUN.

R1. READ A-KAIIN-F AT END MOVE 999 TO A-CO END-READ.
R2. READ B-KAIIN-F AT END MOVE 999 TO B-CO END-READ.
R3. READ M-KAIIN-F AT END MOVE 999 TO M-CO END-READ.
R4. READ H-KAIIN-F AT END MOVE 999 TO H-CO END-READ.
    
```

[COBOL選択者のための問題]

2007.09 1-⑤

【7】 第2図のようなある販売店の売上ファイルを読み、処理条件にしたがって、第3図のような商品別売上一覧表を印字するプログラムの空欄をうめなさい。

入力形式

商品ファイル (ファイル名: HIN-F, レコード名: HIN-R)

商品コード (H-CO)	商品名 (H-MEI)	単価 (H-TAN)
×××××	×～×	×××

(第1図)

売上ファイル (ファイル名: URI-F, レコード名: URI-R)

伝票番号 (DENBAN)	支店コード (U-SCO)	商品コード (U-HCO)	数量 (U-SUU)
××××	××	×××××	×××

(第2図)

出力形式 (ファイル名: LIST-F, レコード名: LIST-R)

(商品別売上一覧表)					
(商品名)	(支店1)	～	(支店10)	(売上合計)	(順位)
×××～×××	×××, ×××	～	×××, ×××	×××, ×××	1
×××～×××	×××, ×××	～	×××, ×××	×××, ×××	1
×××～×××	×××, ×××	～	×××, ×××	×××, ×××	3
}	}	}	}	}	}

(第3図)

処理条件

(1) 第1図の商品ファイルを読み、各データを第4図のテーブル HIN-TBL に記憶する。なお、商品は100種類あり、商品コードの昇順に記録されている。

テーブル HIN-TBL

HIN-T	T-CO	T-MEI	T-TAN
(1)			
(2)			
}	}	}	}
(100)			

(商品コード) (商品名) (単価) (第4図)

(2) 第2図の売上ファイルを読み、商品コードからテーブル HIN-TBL を探索し、単価を求める。なお、商品コードがテーブル HIN-TBL にはない場合はエラーメッセージを表示する。

(3) 第5図のテーブル KEI-TBL には以下の集計を行う。

- 売上金額を次の計算式で求め集計する。

$$\text{売上金額} = \text{単価} \times \text{数量}$$

ただし、行方向の添字はテーブル HIN-TBL の添字に対応している。また、列方向の添字は支店コード 1～10に対応している。

- 11列目には、各商品の売上合計を集計する。

テーブル KEI-TBL

KEI	(1)	(2)	～	(10)	(11)	(12)
(1)			～			
(2)			～			
}	}	}	}	}	}	}
(100)			～			

(支店1) (支店2) ～ (支店10) (売上合計) (順位)

(第5図)

(4) 売上合計の降順に、テーブル KEI-TBL の12列目に順位をつけながら、第3図のように商品別売上一覧表を印字する。

(5) 商品コード以外のデータにエラーはないものとする。

WORKING-STORAGE SECTION.

```

}
01 HIN-TBL.
02 HIN-T OCCURS 100 TIMES
   (ア)
   INDEXED BY M.
03 T-CO PIC X(05).
03 T-MEI PIC X(15).
03 T-TAN PIC 9(03).
01 KEI-TBL.
02 KEI-L OCCURS 100 TIMES.
03 KEI PIC 9(06) OCCURS 12 TIMES.
01 MEISAI-G.
02 M-MEI PIC X(15).
02 M-KIN PIC ZZZZ,ZZ9 OCCURS 12 TIMES.
PROCEDURE DIVISION.
SYORI.
OPEN INPUT HIN-F URI-F OUTPUT LIST-F
INITIALIZE KEI-TBL
PERFORM (イ)
READ HIN-F INTO HIN-T(L)
END-READ
END-PERFORM
MOVE SPACE TO E-FLG
PERFORM UNTIL E-FLG = "ON"
READ URI-F
AT END
MOVE "ON" TO E-FLG
NOT AT END
SEARCH ALL HIN-T
AT END
DISPLAY "ERR U-HCO=" U-HCO
WHEN T-CO(M) = U-HCO
COMPUTE W-KIN = T-TAN(M) * U-SUU
SET P TO M
COMPUTE KEI(P U-SCO) = KEI(P U-SCO) + W-KIN
(ウ)
END-SEARCH
END-READ
END-PERFORM
MOVE 999999 TO HOZON
PERFORM VARYING L FROM 1 BY 1 UNTIL L > 100
PERFORM VARYING K FROM 1 BY 1 UNTIL KEI(K 12) = 0
CONTINUE
END-PERFORM
MOVE K TO MAX
PERFORM UNTIL K = 100
COMPUTE K = K + 1
IF KEI(K 12) = 0 AND KEI(K 11) > KEI(MAX 11)
THEN (エ)
END-IF
END-PERFORM
IF KEI(MAX 11) < HOZON
THEN MOVE L TO JUN
MOVE KEI(MAX 11) TO HOZON
END-IF
MOVE JUN TO KEI(MAX 12)
MOVE T-MEI(MAX) TO M-MEI
PERFORM VARYING K FROM 1 BY 1 UNTIL K > 12
(オ)
END-PERFORM
WRITE LIST-R FROM MEISAI-G AFTER 1
END-PERFORM
CLOSE HIN-F URI-F LIST-F
STOP RUN.

```

[イベント駆動型BASIC選択者のための問題]

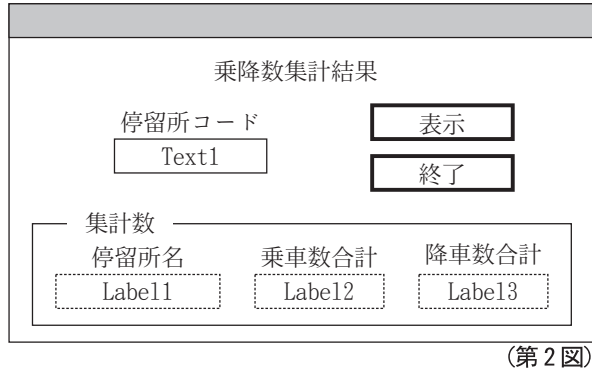
【4】 第1図のようなあるバス路線の1日分の乗降データを読み、処理条件にしたがって第3図のような乗降数集計結果を表示したい。第5図の流れ図(1)~(5)にあてはまる答えを解答群から選び、記号で答えなさい。

入力形式

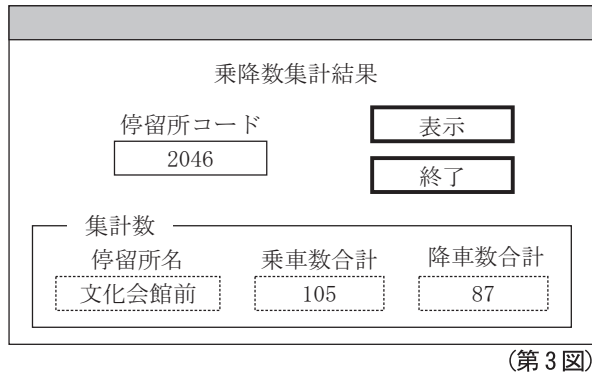
便番号 (Ban)	停留所コード (Tcd)	乗車数 (Jys)	降車数 (Kos)
××	××××	××	××

(第1図)

フォーム



実行形式



処理条件

- (1) 第1図の乗降データは、停留所コードの昇順に記録されている。
- (2) フォームロード時に、乗降データを読み、停留所ごとの乗車数と降車数を第4図の配列 JysH と配列 KosH に集計する。なお、停留所コードと停留所名は、配列 TcdH と配列 MeiH にあらかじめ記憶されており、それぞれの配列は添字で対応している。

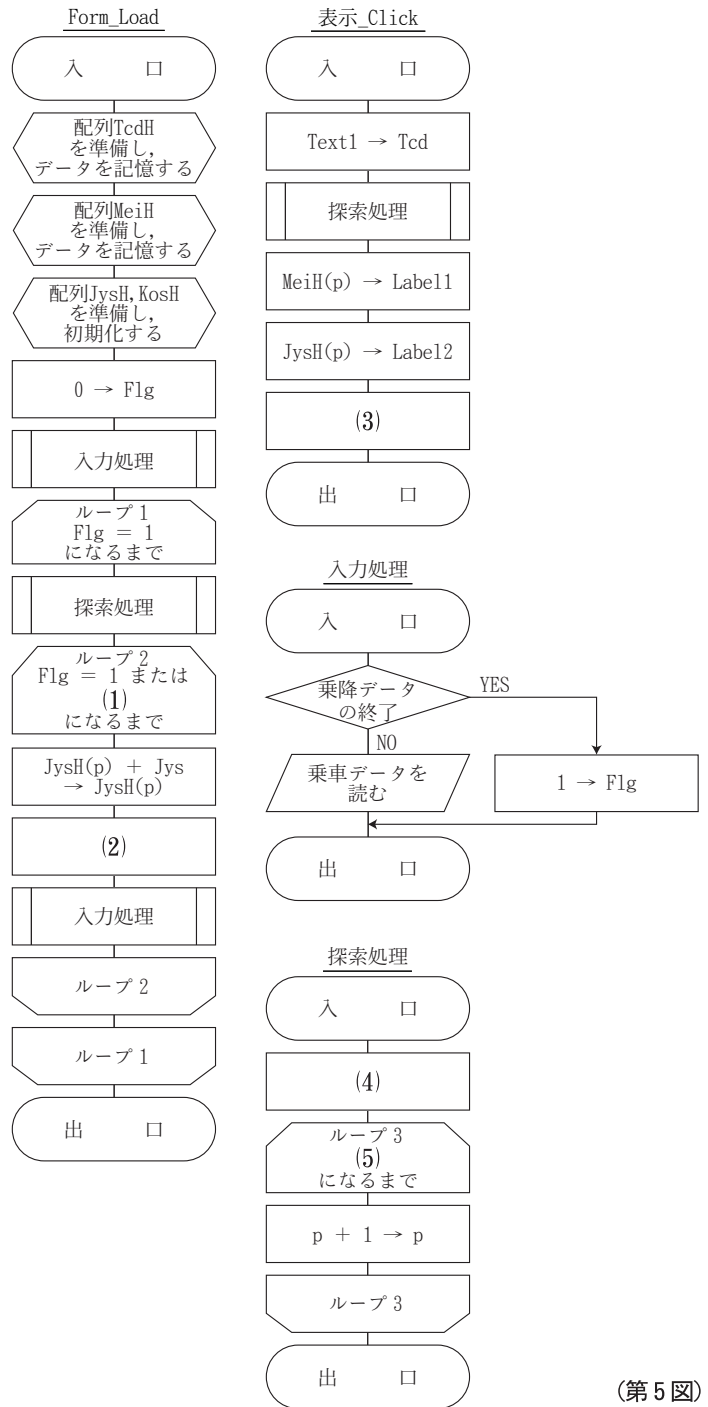
配列

	TcdH	MeiH	JysH	KosH
(1)	××××	××~××		
(2)	××××	××~××		
(3)	××××	××~××		
(4)	××××	××~××		
(5)	××××	××~××		
?	?	?	?	?
(50)	××××	××~××		

(停留所コード) (停留所名) (乗車数合計) (降車数合計)

(第4図)

- (3) Text1 に停留所コードを入力し、「表示」ボタンをクリックすると、第4図の配列から停留所コードを探索し、該当の停留所名、乗車数合計、降車数合計をLabel1 ~ Label3 に表示する。
- (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。



解答群

- ア. $KosH(p) + Kos \rightarrow KosH(p)$
- イ. $TcdH(p) = Text1$
- ウ. $0 \rightarrow p$
- エ. $KosH(p) \rightarrow Label3$
- オ. $TcdH(p) = Tcd$
- カ. $KosH(p) - Kos \rightarrow KosH(p)$
- キ. $TcdH(p) \neq Tcd$
- ク. $Kos \rightarrow Label3$
- ケ. $1 \rightarrow p$
- コ. $TcdH(Ban) \neq Tcd$

[イベント駆動型BASIC選択者のための問題]

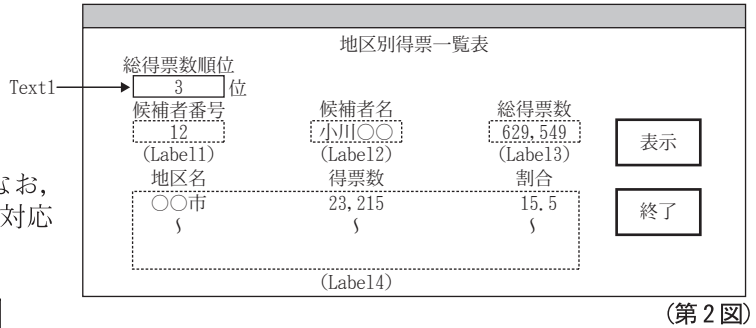
【5】 第1図のようなある知事選挙の地区別得票数を記録した開票データを読み、処理条件にしたがって、第2図のような地区別得票一覧表を表示したい。第7図の流れ図(1)~(5)にあてはまる答えを解答群から選び、記号で答えなさい。

入力形式

地区番号 (Tiban)	候補者番号 (Koban)	得票数 (Su)
××	××	××××××

(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。
- 第3図の配列 Ku には、地区名を記憶する。なお、地区番号は1~30まであり、配列 Ku と添字で対応している。

配列

Ku	〇〇市	△△市	~	〇〇町	△△町
	(1)	(2)	~	(29)	(30)

(第3図)

- 第4図の配列 Ban に候補者番号を、第5図の配列 Na に候補者名を記憶する。なお、候補者番号は1~20まで順に記憶されており、配列 Ban の記憶データは配列 Na の添字と対応している。

配列

Ban	1	2	~	19	20
	(1)	(2)	~	(19)	(20)

(第4図)

配列

Na	山川〇〇	宇野△△	~	松田□□	伊藤◇◇
	(1)	(2)	~	(19)	(20)

(第5図)

- 第1図の開票データを読み、第6図の配列 Sk を利用して、各候補者の得票数を地区ごとに入力する。なお、行方向の添字は候補者番号と対応しており、列方向の添字は地区番号と対応している。
さらに、21行目に地区の得票数計を集計し、31列目に各候補者の総得票数を集計する。

配列

Sk	(1)	(2)	~	(30)	(31)
(1)			~		
(2)			~		
(21)			~		
(22)			~		

(地区の得票数計)

(地区1) (地区2) ~ (地区30) (総得票数) (第6図)

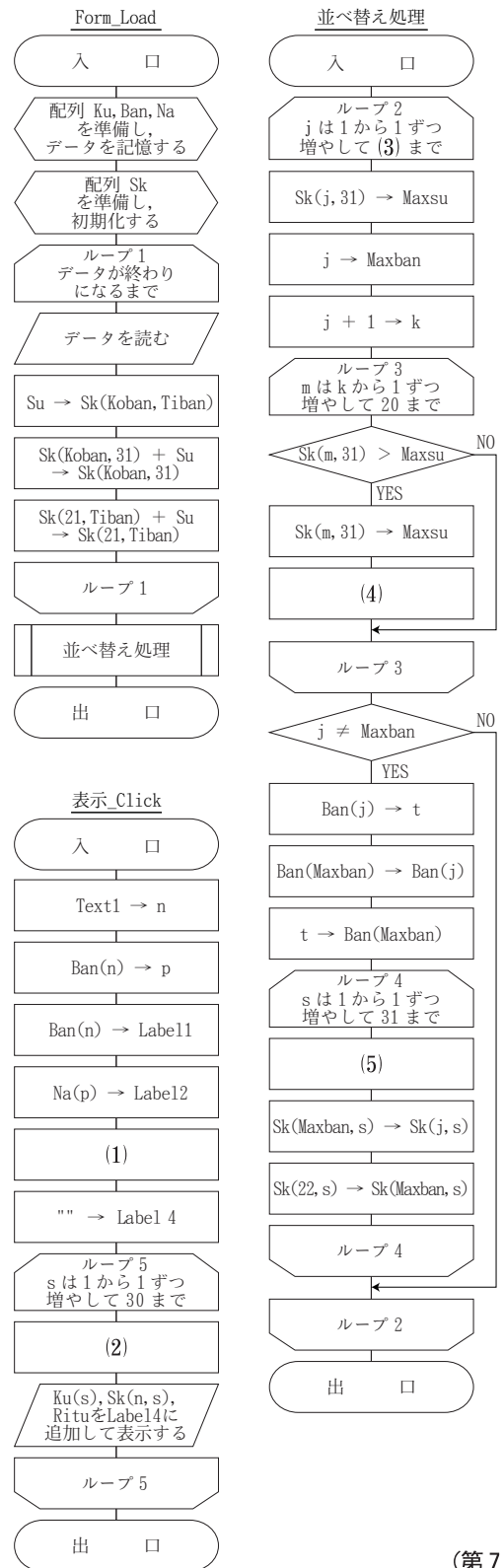
- 第4図の配列 Ban と第6図の配列 Sk のデータを総得票数の降順に並べ替える。ただし、同順位はないものとする。
- (2) Text1 に表示させたい総得票数順位を入力し、「表示」ボタンをクリックすると、その順位の候補者番号を Label1 に、候補者名を Label2 に、総得票数を Label3 に表示する。
さらに、地区番号順に、地区名、得票数、地区の得票数計に対する得票数の割合を Label4 に表示する。なお、割合は次の例のように求める。

例 割合 = $Sk(1, 1) \times 100 \div Sk(21, 1)$

- (3) データにエラーはないものとする。

解答群

- ア. $m \rightarrow Maxban$
- イ. 18
- ウ. $Sk(n, s) \times 100 \div Sk(21, s) \rightarrow Ritu$
- エ. $Sk(22, s) \rightarrow Sk(j, s)$
- オ. $Sk(n, 31) \rightarrow Label3$
- カ. $Ban(m) \rightarrow Maxban$
- キ. 19
- ク. $Sk(p, 31) \rightarrow Label3$
- ケ. $Sk(s, n) \times 100 \div Sk(21, n) \rightarrow Ritu$
- コ. $Sk(j, s) \rightarrow Sk(22, s)$



(第7図)

【6】 あるスイミングスクールの運営会社では、会員利用時間を1か月ごとに一括して集計している。第1図の利用時間ファイルを読み、処理条件にしたがって第2図の会員マスタファイルを更新し、第3図の会員入退会確認表を表示するプログラムの空欄をうめなさい。

入力形式

利用時間ファイル(ファイル名: tkaiin.csv)

会員コード (Tco)	会員名 (Tna)	利用時間 (Tjikan)
×××	×～×	×～×

(第1図)

入出力形式

上段: 会員マスタファイル(ファイル名: mkaiin.csv)

下段: 新会員マスタファイル(ファイル名: nkaiin.csv)

会員コード (Mco)	会員名 (Mna)	時間累計 (Mjikan)
(Nco)	(Nna)	(Njikan)
×××	×～×	×～×

(第2図)

実行形式

(第3図)

処理条件

- (1) 会員マスタファイルおよび利用時間ファイルは、会員コードの昇順に記録されており、会員マスタファイルの会員コードに対して、利用時間ファイルの会員コードは1件あるか、ない場合もある。
- (2) フォームロード時に、次の手順で処理を行う。
 - ① 会員マスタファイルを利用時間ファイルで更新し、新会員マスタファイルに出力する。更新内容は次のとおりである。
 - ・ 利用時間ファイルの会員コードが、会員マスタファイルにない場合は、入会を示しており、新会員マスタファイルに追加する。
 - ・ 利用時間ファイルの会員コードが、会員マスタファイルにある場合は、利用時間ファイルの利用時間を、会員マスタファイルの時間累計に加算する。なお、利用時間ファイルの利用時間が999の場合は退会を示しており、新会員マスタファイルに出力しない。
 - ② 入会者および退会者の会員コードを、第4図の配列Nyukaiおよび配列Taikaiに記憶する。

なお、1か月間の入会・退会者は100人を超えることはない。

配列

Nyukai			～		
	(1)	(2)	～	(99)	(100)
Taikai			～		
	(1)	(2)	～	(99)	(100)

(第4図)

- (3) 「入会」、「退会」チェックボックスを選択し、「表示」ボタンをクリックすると、第3図の実行形式のように、それぞれの会員コードを表示する。
- (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。

```

Option Explicit
Dim Nyukai(1 To 100), Taikai(1 To 100) As Integer
Dim Mco, Tco, Mjikan, Tjikan As Integer
Dim Mna, Tna As String
Dim s, t As Integer

Private Sub Form_Load()
    s = 0: t = 0
    Check1.Value = 0
    Check2.Value = 0
    Text1.Text = ""
    Open "mkaiin.csv" For Input As #1
    Open "tkaiin.csv" For Input As #2
    Open "nkaiin.csv" For Output As #3
    Call Mkaiin
    Call Tkaiin
    Do Until (ア)
        Select Case Mco
            Case Is < Tco
                Write #3, (イ)
                Call Mkaiin
            Case Is > Tco
                Write #3, Tco, Tna, Tjikan
                s = s + 1
                Nyukai(s) = Tco
                Call Tkaiin
            Case Else
                If (ウ) Then
                    t = t + 1
                    Taikai(t) = Mco
                Else
                    Mjikan = Mjikan + Tjikan
                    Write #3, Mco, Mna, Mjikan
                End If
            Call Mkaiin
        End Select
    Loop
    Close #1, #2, #3
End Sub

Private Sub 表示_Click()
    Dim i As Integer
    Text1.Text = ""
    If Check1.Value = 1 Then
        For i = 1 To s
            Text1.Text = Text1.Text & "入会コード" & _
                (オ) & Chr(13) & Chr(10)
        Next i
    End If
    If Check2.Value = 1 Then
        For i = 1 To t
            Text1.Text = Text1.Text & "退会コード" & _
                Taikai(i) & Chr(13) & Chr(10)
        Next i
    End If
End Sub

Private Sub 終了_Click()
    End
End Sub

Private Sub Mkaiin()
    If EOF(1) Then
        Mco = 999
    Else
        Input #1, Mco, Mna, Mjikan
    End If
End Sub

Private Sub Tkaiin()
    If EOF(2) Then
        Tco = 999
    Else
        Input #2, Tco, Tna, Tjikan
    End If
End Sub
    
```

(注) 第3図のText1の表示は見やすく示してある。

[イベント駆動型BASIC選択者のための問題]

【7】 ある販売店の売上データを入力し、処理条件にしたがって第2図のように表示するプログラムの空欄をうめなさい。

入力形式

商品データ (ファイル名: hin.csv)

商品コード ×××	商品名 ×～×	単価 ×～×
--------------	------------	-----------

(第1図)

処理条件

- (1) フォームロード時に、第1図の商品データを読み、各データを第3図の配列に記憶する。なお、商品は100種類あり、商品コードの昇順に記録されている。また、各配列は添字で対応している。

配列

	Tco	Tmei	Tnk
(1)			
(2)			
}	}	}	}
(00)			

(商品コード) (商品名) (単価) (第3図)

- (2) 第2図のText1～Text3に売上データを入力し、「処理」ボタンをクリックすると、次の処理を行う。
- Text1に入力された商品コードから第3図の配列Tcoを探索し、配列Tmeiより商品名をLabel1に表示する。ただし、商品コードが配列Tcoにない場合はエラーメッセージを表示する。
 - 売上金額を次の式より求め、Label2に表示する。
売上金額 = 単価 × 数量
 - 第4図の配列Ktに、商品別、支店別の売上金額を集計する。ただし、行方向の添字は配列Tcoの添字と対応しており、列方向の添字は支店コード1～5と対応している。また、6列目には、売上合計を集計する。
 - 配列Ktの7列目に、6列目で集計された売上合計の降順に順位をつけ、売上一覧表としてText4に表示する。

配列

Kt	(1)	(2)	～	(5)	(6)	(7)
(1)			～			
(2)			～			
}	}	}	}	}	}	}
(00)			～			

(支店1) (支店2) ～ (支店5) (売上合計) (順位) (第4図)

- (3) 商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 Tco(1 To 100) As Integer
Dim Tmei(1 To 100) As String
Dim Tnk(1 To 100) As Long
Dim Kt(1 To 100, 1 To 7) As Long
Dim Row, Col As Integer

Private Sub Form_Load()
    Open "hin.csv" For Input As #1
    For Row = 1 To 100
        Input #1, Tco(Row), Tmei(Row), Tnk(Row)
        For Col = 1 To 6
            (ア)
        Next Col
    Next Row
    Close #1
    Text1.Text = "": Text2.Text = ""
    Text3.Text = "": Text4.Text = ""
    Label1.Caption = "": Label2.Caption = ""
End Sub
```

実行形式

(第2図)

```
Private Sub 処理_Click()
    Dim Flg, k As Integer
    Dim Sita, Ue, m As Integer
    Dim Hco, Sco As Integer
    Dim Su, Urikin As Long
    Hco = Val(Text1.Text)
    Flg = 0: Sita = 0: Ue = 101
    Do Until Flg = 1 Or Ue = Sita + 1
        m = Int((イ))
        If Tco(m) = Hco Then
            Flg = 1
        ElseIf Tco(m) < Hco Then
            Sita = m
        Else
            Ue = m
        End If
    Loop
    Sco = Val(Text2.Text)
    Su = Val(Text3.Text)
    If Flg = 1 Then
        Urikin = Tnk(m) * Su
        (ウ)
        Kt(m, 6) = Kt(m, 6) + Urikin
        Label1.Caption = Tmei(m)
        Label2.Caption = Format(Urikin, "#,###,##0")
        Text4.Text = ""
        For Row = 1 To 100
            Kt(Row, 7) = 1
            For k = 1 To 100
                If (エ) Then
                    Kt(Row, 7) = Kt(Row, 7) + 1
                End If
            Next k
            Text4.Text = Text4.Text & _
                Format((オ), "&&&&&&&&") & " "
        For Col = 1 To 7
            Text4.Text = Text4.Text & _
                Format(Kt(Row, Col), "#,###,##0") & " "
        Next Col
        Text4.Text = Text4.Text & Chr(13) & Chr(10)
    End If
    MsgBox ("商品コードエラー" & Hco & "番は登録なし")
    Text1.Text = "": Text2.Text = ""
    Text3.Text = "": Text4.Text = ""
    Label1.Caption = "": Label2.Caption = ""
End If
End Sub

Private Sub 終了_Click()
    End
End Sub
```

(注) 第2図のText4の表示は見やすく示してある。

主催 財団法人 全国商業高等学校協会

平成19年度 (第37回) 情報処理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)

【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】

(ア)	
(イ)	
(ウ)	
(エ)	
(オ)	

【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19年度 (第37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
ウ	ク	シ	コ	エ

各2点
計10点

【2】

(1)	(2)	(3)	(4)	(5)
サ	エ	ケ	イ	カ

各2点
計10点

【3】

1	2	3	4	5
イ	ア	ア	ウ	イ

各2点
計10点

小 計
30

..... [COBOL]

【4】

(1)	(2)	(3)	(4)	(5)
キ	ア	エ	ケ	オ

各3点
計15点

【5】

(1)	(2)	(3)	(4)	(5)
オ	ウ	キ	ア	コ

各3点
計15点

【6】

(ア)	A-CO < B-CO
(イ)	M-CO = 999 AND H-CO = 999
(ウ)	WRITE N-KAIIN-R FROM M-KAIIN-R
(エ)	H-JIKAN NOT = 999
(オ)	PERFORM R3 THRU R4

各4点 計20点

【7】

(ア)	ASCENDING KEY T-CO
(イ)	VARYING L FROM 1 BY 1 UNTIL L > 100
(ウ)	COMPUTE KEI (P 11) = KEI (P 11) + W-KIN
(エ)	MOVE K TO MAX
(オ)	MOVE KEI (MAX K) TO M-KIN (K)

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 計
70

合 計
100

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19年度 (第37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
ウ	ク	シ	コ	エ

各2点
計10点

【2】

(1)	(2)	(3)	(4)	(5)
サ	エ	ケ	イ	カ

各2点
計10点

【3】

1	2	3	4	5
イ	ア	ア	ウ	イ

各2点
計10点

小 計
30

..... [イベント駆動型BASIC]

【4】

(1)	(2)	(3)	(4)	(5)
キ	ア	エ	ケ	オ

各3点
計15点

【5】

(1)	(2)	(3)	(4)	(5)
オ	ウ	キ	ア	コ

各3点
計15点

【6】

(ア)	Mco = 999 And Tco = 999
(イ)	Mco, Mna, Mjikan
(ウ)	Tjikan = 999
(エ)	Call Tkaiin
(オ)	Nyukai(i)

各4点 計20点

【7】

(ア)	Kt (Row, Col) = 0
(イ)	(Sita + Ue) / 2
(ウ)	Kt (m, Sco) = Kt (m, Sco) + Urikin
(エ)	Kt (Row, 6) < Kt (k, 6)
(オ)	Tmei (Row)

(注) 【6】，【7】については、大文字、小文字および=、演算子の前後の空白は問わない。 各4点 計20点

試験場校名	受験番号	選択言語	小 計	合 計
		COBOL	70	100

イベント駆動型 BASIC

選択言語を で囲むこと