

2020年 9 月27日実施

令和 2 年度 (第 63 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】～【6】は共通問題です。
5. 【7】の問題はJava・マクロ言語のいずれか一つを選択し、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 公益財団法人 全国商業高等学校協会

受 験 番 号

【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。

- データ要素に、次のデータ要素の位置情報を持たせることで、複数のデータを連結するデータ構造。
- 周辺装置をコンピュータに接続した際、OSが自動的にその装置を利用するための必要な設定を行う機能。
- 故障率の低さやメンテナンスの容易さなど、五つの特性により判断するコンピュータシステムの総合的な評価指標。
- モジュール単位で開発されたプログラムをつなぎ合わせ、モジュール同士で正しくデータが受け渡され、機能するかを調べるテスト。
- 送信者が暗号化に用いる鍵と受信者が復号に用いる鍵が、それぞれ異なる暗号方式。

解答群

ア. ドライバ	イ. 結合テスト	ウ. NAS
エ. 共通鍵暗号方式	オ. ポインタ	カ. リスト
キ. 単体テスト	ク. 公開鍵暗号方式	ケ. シンクライアント
コ. プラグアンドプレイ	サ. RAID	シ. RASIS

【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. ハブ                                      2. 外部設計                                      3. クロック周波数  
4. システムログ                                      5. 情報落ち

<B群>

- ア. 極端に絶対値の差が大きい数値同士の加減算を行った結果、絶対値の小さい数値が無視されてしまう現象。
- イ. CPUやメモリなど、複数の回路間で同期をとるための電気信号が、1秒間に何回発生するかを表す値。
- ウ. システム開発において、システムの全体像を捉え、要求される項目を調査・分析・整理する工程。
- エ. 複数のネットワーク機器をLANケーブルにより集線し、送受信される電気信号を増幅して中継するLAN間接続装置。
- オ. サーバに接続された履歴の把握などに用いられ、コンピュータシステムへの接続時刻や接続機器のIPアドレスなどを記録したもの。
- カ. 異なるプロトコルのネットワーク同士を接続するため、プロトコルの変換を行うLAN間接続装置。
- キ. ハードウェアやアプリケーションソフトウェアの動作障害時の原因究明などに用いられ、コンピュータシステム内で起きた動作履歴をOSが記録したもの。
- ク. 浮動小数点演算において、絶対値の非常に近い数値同士の減算した結果、有効数字の桁数が極端に少くなる現象。
- ケ. システム開発において、要件定義に基づき、入出力画面や帳票などを設計する工程。
- コ. コンピュータ内部において、1秒間に処理できる命令数を100万単位で表す値。

【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. 8ビットの2進数の 00010010 と 00001001 がある。この2つの数値の積を16進数で表したものの。

ア. 09                                      イ. 1B                                      ウ. A2

2. コンピュータシステムの障害を修復してから、次の障害が発生するまでの平均稼働時間。

ア. HTTP                                      イ. MTBF                                      ウ. MTTR

3. 集合の関係を円で表現し、論理演算を視覚的にわかりやすくしたもの。

ア. ベン図                                      イ. 論理回路                                      ウ. OSI参照モデル

4. コンピュータシステムに大量のデータを処理させたり、アクセスを集中させたりすることで、耐久能力などを調べるテスト。

ア. 回帰テスト                                      イ. 機能テスト                                      ウ. 負荷テスト

5. 平均シーク時間が20ミリ秒で、5,000回転／分のハードディスク装置の平均待ち時間は何ミリ秒か。

ア. 22ミリ秒                                      イ. 26ミリ秒                                      ウ. 32ミリ秒

【 4 】 次の各問いに答えなさい。

問 1. 流れ図の説明を読んで、流れ図の(1)~(2)にあてはまる答えを解答群から選び、記号で答えなさい。

<流れ図の説明>

処理内容

配列に記憶した数値に順位をつけてディスプレイに表示する。

処理条件

1. 配列 Ten に数値を記憶する。なお、データ件数は n 件である。

配列

Ten	(0)	(1)	~	(n-1)	(n)
		270	~	832	465

2. 配列 Jun を利用し、配列 Ten の数値の降順に順位をつける。なお、数値が同じ場合は同順位とする。

配列

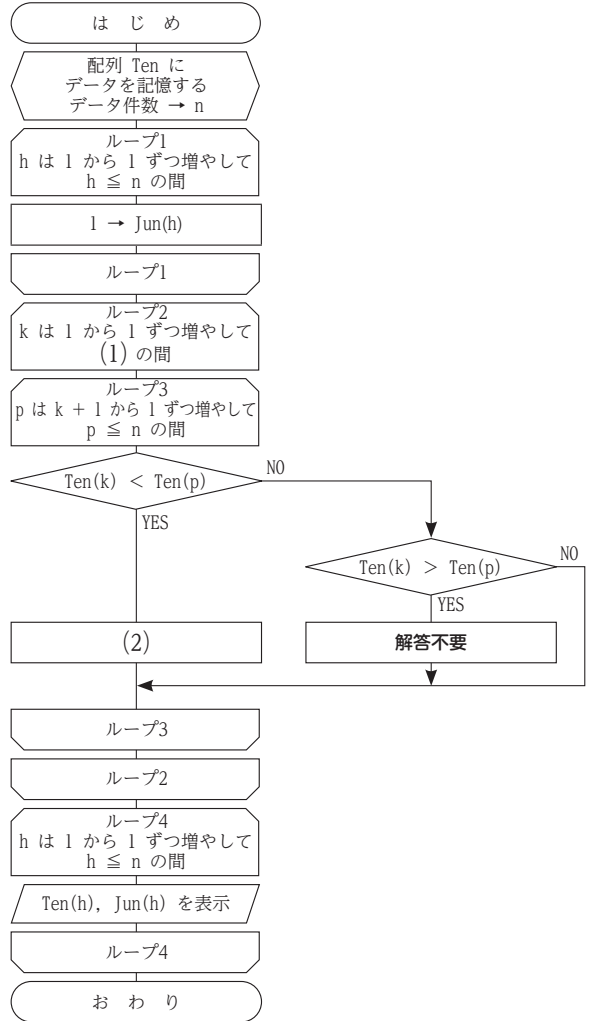
Jun	(0)	(1)	~	(n-1)	(n)
			~		

3. 順位をつけ終わったら、配列 Ten と配列 Jun の内容を表示する。

解答群

- ア.  $Jun(k) + 1 \rightarrow Jun(k)$
- イ.  $k < n - 1$
- ウ.  $k \leq n - 1$
- エ.  $Jun(p) + 1 \rightarrow Jun(p)$

<流れ図>



問 2. 流れ図の説明を読んで、流れ図の(3)~(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。

<流れ図の説明>

処理内容

配列に記憶した文字列を並べ替えてディスプレイに表示する。

処理条件

1. 配列 Bird に文字列を記憶する。なお、データ件数は n 件である。

配列

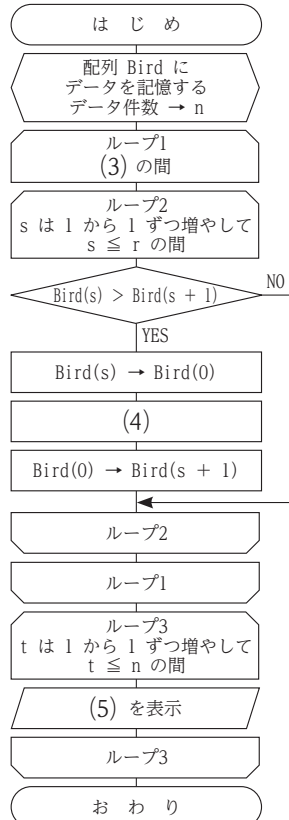
Bird	(0)	(1)	~	(n-1)	(n)
		duck	~	starling	eagle

2. 配列 Bird の文字列を昇順に並べ替える。  
3. 並べ替えが終わったら、配列 Bird の内容を表示する。

解答群

- ア. r は n から 1 ずつ減らして  $r \geq 1$
- イ.  $Bird(s + 1) \rightarrow Bird(s)$
- ウ.  $Bird(n)$
- エ.  $Bird(t)$
- オ. r は n - 1 から 1 ずつ減らして  $r \geq 1$
- カ.  $Bird(s) \rightarrow Bird(s + 1)$

<流れ図>



【5】 流れ図の説明を読んで、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。

<流れ図の説明>

処理内容

あるドラッグストアの1日の売上データを読み、集計結果をディスプレイに表示する。

入力データ

伝票番号 (Dban) ×~×	時刻 (Ji) ××××	商品コード (Sc) ××××	数量 (Su) ××
-----------------------	--------------------	-----------------------	------------------

(第1図)

実行結果

(ドラッグストアの売上集計結果)

(時間帯)	(医薬品)	(食品)	~	(日用品)	(その他)
9時台	20,000	16,400	~	50,800	30,700
}	}	}	}	}	}
20時台	58,800	39,000	~	68,600	60,600
(分類計)	802,200	480,600	~	727,800	534,900
(対医薬品比(%))	—	60	~	91	

(第2図)

処理条件

1. 第1図の時刻は次の例のように構成されている。なお、時間帯は9(9時台)~20(20時台)、分は00~59である。

例 1325 → 13 25  
          時間帯 分

2. 第1図の商品コードは次の例のように構成されている。なお、分類番号は1(医薬品)~9(その他)である。

例 1102 → 1 102  
          分類番号 連番

3. 配列 Sco に商品コードを、配列 Tanka に単価を、商品コードの昇順に記憶する。なお、商品は5,000種類であり、Sco と Tanka の添字は対応している。

配列

Sco	(0)	(1)	(2)	~	(499)	(500)
		1001	1002	~	9504	9505
Tanka	(0)	(1)	(2)	~	(499)	(500)
		1800	2100	~	1100	500

4. 第1図の入力データを読み、次の処理を行う。  
・ 商品コードをもとに配列 Sco を探索し、売上金額を次の計算式で求め、配列 Ukei に集計する。なお、0行目には分類計を求める。また、Ukei の行方向の添字は時間帯により求め、列方向の添字は分類番号と対応している。

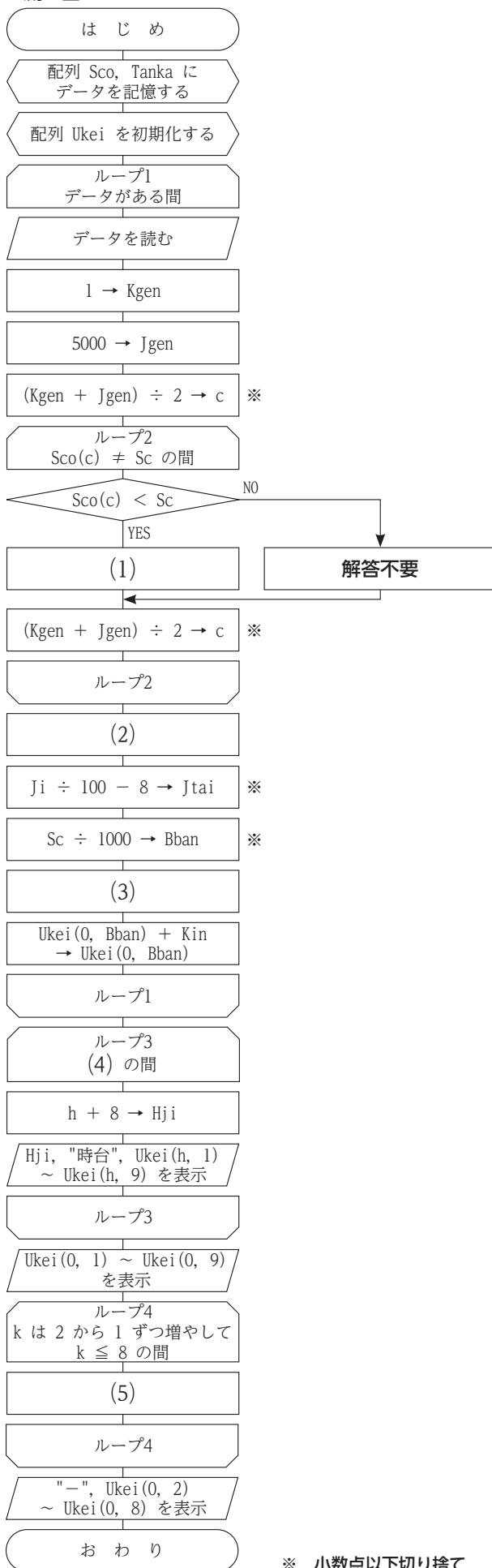
売上金額 = 数量 × 単価

配列

Ukei	(0)	(1)	(2)	~	(8)	(9)
(0)				~		
(1)				~		
}	}	}	}	}	}	}
(12)				~		
	(医薬品)	(食品)	~	(日用品)	(その他)	

5. 入力データが終了したら、次の処理を行う。  
・ 時間帯からその他までを第2図のように表示する。  
・ 分類計を第2図のように表示する。  
・ 配列 Ukei の0行目を利用して、食品から日用品までの対医薬品比(%)を次の計算式で求める。  
対医薬品比(%) = 分類計 × 100 ÷ 医薬品の分類計  
・ 対医薬品比(%)を第2図のように表示する。  
6. データにエラーはないものとする。

<流れ図>



※ 小数点以下切り捨て

解答群

- ア.  $c + 1 \rightarrow Kgen$
- イ.  $Ukei(Bban, Jtai) + Kin \rightarrow Ukei(Bban, Jtai)$
- ウ.  $h$  は 1 から 1 ずつ増やして  $h \leq 12$
- エ.  $Ukei(0, 1) \times 100 \div Ukei(0, k) \rightarrow Ukei(0, k)$
- オ.  $c - 1 \rightarrow Jgen$
- カ.  $Su \times Tanka(c) \rightarrow Kin$
- キ.  $Ukei(0, k) \times 100 \div Ukei(0, 1) \rightarrow Ukei(0, k)$
- ク.  $Su \times Tanka(Kgen) \rightarrow Kin$
- ケ.  $Ukei(Jtai, Bban) + Kin \rightarrow Ukei(Jtai, Bban)$
- コ.  $h$  は 0 から 1 ずつ増やして  $h \leq 12$

[6] 流れ図の説明を読んで、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。

<流れ図の説明>

処理内容

国内の1年分の宿泊者データを読み、集計結果をディスプレイに表示する。

入力データ

エリア番号 (Eban) ×	都道府県番号 (Tban) ××	年 (Nen) ××××	月 (Tuki) ××	種別番号 (Sban) ×	区分 (Kbun) ×	人数 (Nsu) ×~×
----------------------	------------------------	--------------------	-------------------	---------------------	-------------------	--------------------

(第1図)

実行結果

(国内宿泊人数集計表) (エリア名)	(都道府県名)	(高級旅館)	(リゾートホテル)	( ) 内は、うち外国人宿泊人数 (合計)				
北海道	北海道	5,366,670	【 1,036,620】	3,417,880	【 973,960】	~	26,603,300	【 5,568,400】
	(エリア計)	5,366,670		3,417,880		~	26,603,300	
九州・沖縄	福岡県	561,200	【 28,290】	684,570	【 166,420】	~	12,681,530	【 2,228,280】
	沖縄県	260,390	【 30,720】	7,038,700	【 1,317,510】	~	16,597,700	【 3,614,870】
	(エリア計)	7,886,580		13,217,570		~	59,194,530	
(種別名)	(上位5位の都道府県名：合計)							
高級旅館	北海道	5,366,670		静岡県	5,316,380	~		
ユースホステル	神奈川県	698,520		静岡県	557,760	~		

(第2図)

処理条件

- 第1図の宿泊者データは、エリア番号、都道府県番号の昇順に記録されている。なお、エリア番号は1~8、都道府県番号は1~47、種別番号は1~6、区分は1(日本人)と2(外国人)である。
- 配列 Emei にエリア名を、配列 Tmei に都道府県名を記憶する。なお、Emei の添字はエリア番号と対応し、Tmei の添字は都道府県番号と対応している。

配列

Emei	Tmei
(0) [ ]	(0) [ ]
(1) 北海道	(1) 北海道
}	}
(8) 九州・沖縄	(46) 鹿児島県
	(47) 沖縄県

- 配列 Smei に種別名を記憶する。なお、Smei の添字は種別番号と対応している。

配列

Smei	(0)	(1)	(2)	~	(6)
	[ ]	高級旅館	リゾートホテル	~	ユースホステル

- 第1図の入力データを読み、次の処理を行う。

- エリア番号が変わるごとにエリア名を第2図のように表示する。
- 配列 Gsyu に外国人の人数を集計する。なお、Gsyu(7) には合計を求める。また、Gsyu の添字は種別番号と対応している。

配列

Gsyu	(0)	(1)	(2)	~	(6)	(7)
	[ ]	[ ]	[ ]	~	[ ]	(合計)

- 配列 Tsyu に都道府県ごとに人数を集計する。なお、Tsyu の7列目には合計を求める。また、Tsyu の行方向の添字は都道府県番号と対応し、列方向の添字は種別番号と対応している。

配列

Tsyu	(0)	(1)	(2)	~	(6)	(7)
(0)	[ ]	[ ]	[ ]	~	[ ]	[ ]
(1)	[ ]	[ ]	[ ]	~	[ ]	[ ]
}	}	}	}	}	}	}
(46)	[ ]	[ ]	[ ]	~	[ ]	[ ]
(47)	[ ]	[ ]	[ ]	~	[ ]	[ ]

(エリア計)

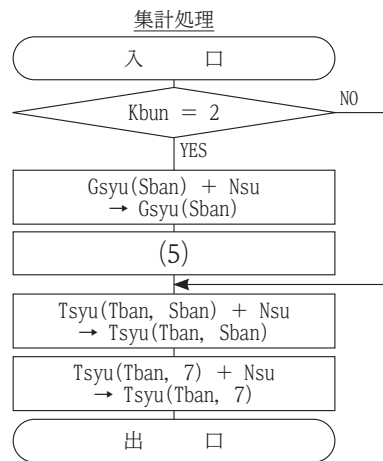
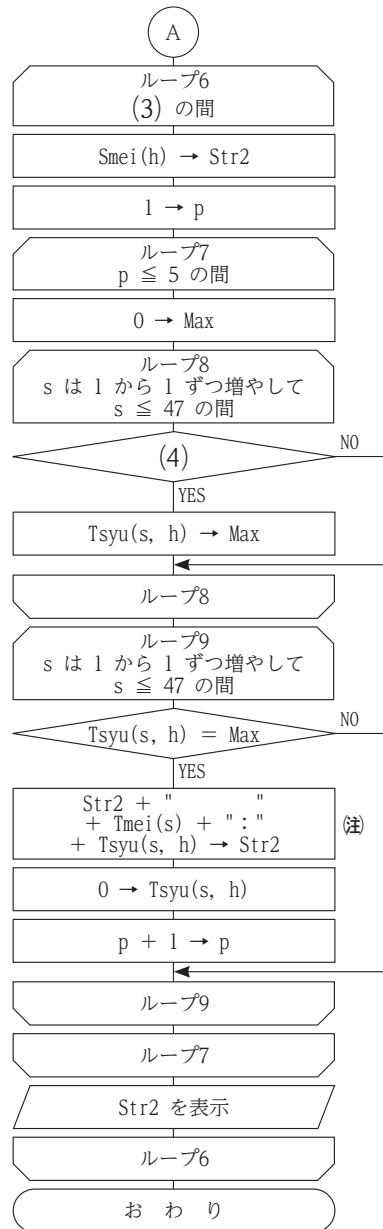
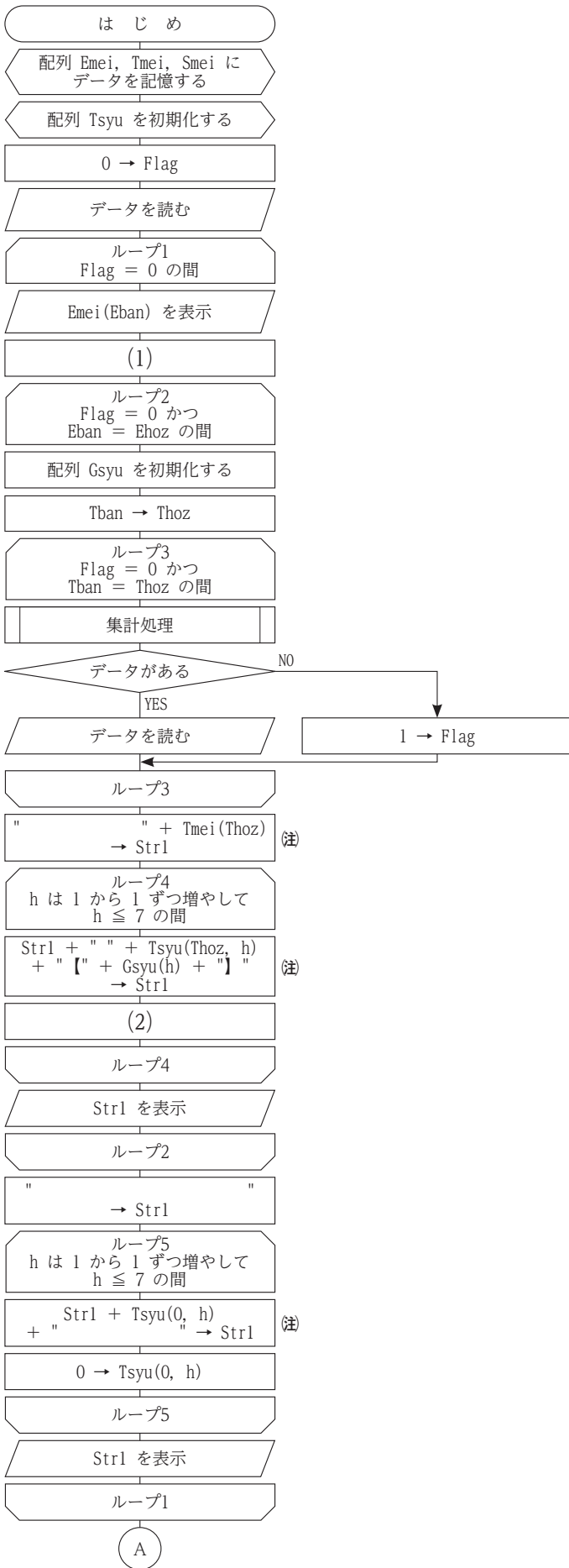
(合計)

- 都道府県番号が変わるごとに都道府県名から合計までを第2図のように表示する。なお、配列 Tsyu の0行目を利用してエリア計を求める。
  - エリア番号が変わるごとにエリア計を第2図のように表示する。
- 入力データが終了したら、種別ごとに合計の上位5位を求め、種別名と上位5位の都道府県名：合計を第2図のように表示する。
  - データにエラーはないものとする。

解答群

- |  |  |
|--|--|
| ア. Tsyu(0, h) + Tsyu(Thoz, h) → Tsyu(0, h) | イ. Tuki → Ehoz                             |
| ウ. h は 6 から 1 ずつ減らして h ≥ 1                 | エ. Gsyu(7) + Nsu → Gsyu(7)                 |
| オ. Tsyu(s, h) > Max                        | カ. Tsyu(0, h) + Tsyu(Tban, h) → Tsyu(0, h) |
| キ. Gsyu(Tban) + Nsu → Gsyu(Tban)           | ク. Eban → Ehoz                             |
| ケ. Tsyu(s, h) < Max                        | コ. h は 1 から 1 ずつ増やして h ≤ 6                 |

<流れ図>



(注) ここでの「+」は、文字列結合を意味する。

[7] プログラムの説明を読んで、プログラムの(1)~(5)を答えなさい。

<プログラムの説明>

処理内容

旅行かばん店の1週間分の販売データを読み、集計結果を表示する。

入力データ

販売データ (ファイル名: sales.csv)

日付	レシート番号	店舗番号	商品番号	販売数量
××××	×~×	×	×××	×

(第1図)

実行結果

店舗番号(1~7)を入力してください→7			
羽田店 【空港内店舗】			
商品番号	商品名	数量計	金額計
101	ビジネス黒	7	148,800
137	北欧バッグ	1	18,000
	合計		3,039,100
商品番号	サービス商品名		金額計
801	鍵交換修理		24,500
902	油差し		4,500
	合計		213,620
店舗番号(1~7)を入力してください→0			

(第2図)

処理条件

- 第1図の店舗番号は 1 (新千歳店) ~7 (羽田店) である。
- 配列 sName に店舗名を、配列 sKind に店舗種別を記憶する。なお、sName, sKind の添字は店舗番号と対応している。また、店舗種別は 1 (通常店舗), 2 (空港内店舗) である。

配列

sName	(0)	(1)	(2)	~	(6)	(7)
		新千歳店	仙台店	~	渋谷駅店	羽田店
sKind	(0)	(1)	(2)	~	(6)	(7)
		2	1	~	1	2

- 通常店舗の場合は通常店舗を管理する Shop クラスを、空港内店舗の場合は空港内店舗を管理する AirportShop クラスをインスタンス化し、配列 sp に記憶する。なお、Shop クラスは AirportShop クラスのスーパークラスである。また、sp の添字は店舗番号と対応している。

配列

sp	(0)	(1)	(2)	~	(6)	(7)
				~		

- 配列 gCode に商品番号を、配列 gName に商品名を、配列 gPrice に単価を記憶する。なお、空港内店舗のみで提供される空港内サービス商品については、同様に配列 asCode, 配列 asName, 配列 asPrice に記憶する。また、gCode, gName, gPrice の添字, asCode, asName, asPrice の添字はそれぞれ対応している。

配列 (通常店舗)

gCode	(0)	(1)	~	(48)	(49)
	101	102	~	236	238
gName	(0)	(1)	~	(48)	(49)
	ビジネス黒	ビジネス紺	~	キャリー橙	キャリー黒
gPrice	(0)	(1)	~	(48)	(49)
	24000	24000	~	28000	28000

配列 (空港内店舗)

asCode	(0)	~	(19)
	801	~	902
asName	(0)	~	(19)
	鍵交換修理	~	油差し
asPrice	(0)	~	(19)
	3500	~	1500

- 第1図の販売データを読み、次の①または②の処理を行う。

- 通常商品の場合、商品番号をもとに配列 gCode を探索し、配列 gSales の0行目に数量を集計する。なお、金額を次の計算式で求め gSales の1行目に集計する。ただし、数量が2個以上の場合、20%の割引きを行う。また、gSales の列方向の添字は gCode の添字と対応している。

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

配列 (通常店舗)

gSales	(0)	(1)	~	(48)	(49)
(0)			~		
(1)			~		

(数量計)  
(金額計)

- 空港内サービス商品の場合、商品番号をもとに配列 asCode を探索し、金額を次の計算式で求め配列 asSales に集計する。なお、商品番号が見つからなかった場合、①の処理を行う。また、asSales の添字は asCode の添字と対応している。

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

配列 (空港内店舗)

asSales	(0)	~	(19)
		~	

(金額計)

- データを読み終えたあと、キーボードから店舗番号が入力されたら、次の③または④の処理を行う。

- 通常店舗の場合、配列 wk を利用して金額計の降順に並べ替え、販売がある商品について第2図のようにディスプレイに表示する。
- 空港内店舗の場合、③の処理を行ったあと、販売がある空港内サービス商品についてを商品番号順に表示する。

配列

wk	(0)	(1)	~	(48)	(49)
	0	1	~	48	49

- キーボードから 0 が入力されたら処理を終了する。

## &lt;Javaプログラム&gt;

```

//クラスShop
public class Shop {
    private static int[] gCode = { 101, 102, ~ 236, 238 };
    private static String[] gName = { "ビジネス黒", "ビジネス紺", ~ "キャリア橙", "キャリア黒" };
    private static int[] gPrice = { 24000, 24000, ~ 28000, 28000 };
    private int[][] gSales = new int[2][50]; private String sName; private int gTotal; private int[] wk = new int[50];
    public Shop(String sName) {
        this.sName = sName;
    }
    public void calc(int gCd, int gQn) {
        int idx = 0;
        while(gCode[idx] != gCd) {
            idx += 1;
        }
        gSales[0][idx] += gQn;
        int sFee = gQn * gPrice[idx];
        if(gQn >= 2) {
            sFee = (int) (sFee * (1 - 0.2));
        }
        (1);
        gTotal += sFee;
    }
    public void sort() {
        for(int j = 0; j < gCode.length; j++) {
            wk[j] = j;
        }
        for(int j = 1; j < gCode.length; j++) {
            int k, work = wk[j];
            for(k = j; k > 0 && gSales[1][wk[k - 1]] < gSales[1][work]; k--) {
                (2);
            }
            wk[k] = work;
        }
    }
    public void output() {
        System.out.printf("%-11s¥n", sName);
        System.out.println(" 商品番号 商品名          数量計      金額計");
        for(int j = 0; j < gCode.length; j++) {
            if(gSales[0][wk[j]] > 0) {
                System.out.printf("          %3d %-5s          %2d      %,7d¥n", gCode[wk[j]], (3), gSales[0][wk[j]], gSales[1][wk[j]]);
            } else {
                break;
            }
        }
        System.out.printf("          合計      %,9d¥n", (4));
    }
}

//クラスAirportShop
public class AirportShop extends Shop {
    private int[] asCode = { 801, ~ 902 }; private String[] asName = { "鍵交換修理", ~ "油差し" };
    private int[] asPrice = { 3500, ~ 1500 };
    private int[] asSales = new int[20]; private int asTotal;
    public AirportShop(String sName) {
        super(sName + " 【空港内店舗】");
    }
    public void calc(int gCd, int gQn) {
        int idx = 0;
        while(idx < asCode.length && asCode[idx] != gCd) {
            idx += 1;
        }
        if(idx < asCode.length) {
            int sFee = gQn * asPrice[idx];
            asSales[idx] += sFee;
            asTotal += sFee;
        } else {
            super.calc(gCd, gQn);
        }
    }
    public void output() {
        super.output();
        (注) 省略されている処理の詳細は処理条件6④
        System.out.printf("          合計      %,7d¥n", asTotal);
    }
}

//クラスAnalysis
import java.io.BufferedReader;

public class Analysis {
    private static String[] sName = { "", "新千歳店", "仙台店", ~ "渋谷駅店", "羽田店" }; private static int[] sKind = { 0, 2, 1, ~ 1, 2 };
    private static Shop[] sp = new Shop[8];
    public static void main(String[] args) {
        for(int i = 1; i < sName.length; i++) {
            if((5)) {
                sp[i] = new Shop(sName[i]);
            } else {
                sp[i] = new AirportShop(sName[i]);
            }
        }
    }
    try {
        BufferedReader fileIn = new BufferedReader(new FileReader("sales.csv"));
        String line;
        while((line = fileIn.readLine()) != null) {
            String[] str = line.split(",");
            int sCd = Integer.parseInt(str[2]), gCd = Integer.parseInt(str[3]), gQn = Integer.parseInt(str[4]);
            sp[sCd].calc(gCd, gQn);
        }
        fileIn.close();
    } catch(Exception e) { System.out.println("エラーが発生しました" + e); }
    Scanner codeIn = new Scanner(System.in);
    System.out.print("店舗番号(1~7)を入力してください→"); int sCdIn = codeIn.nextInt();
    while(sCdIn != 0) {
        sp[sCdIn].sort(); sp[sCdIn].output();
        System.out.print("店舗番号(1~7)を入力してください→"); sCdIn = codeIn.nextInt();
    }
    codeIn.close();
}
}

```



[7] プログラムの説明を読んで、プログラムの(1)~(5)を答えなさい。

<プログラムの説明>

処理内容

あるノベルティグッズショップの顧客情報データと販売データを読み、顧客種別販売金額集計と顧客別販売分析を表示する。

入力データ

顧客情報データ (ファイル名: kokyaku.csv)

顧客番号	顧客種別	顧客名
×××	×	×～×

(第1図)

販売データ (ファイル名: hanbai.csv)

販売年月日	顧客番号	商品コード	販売数量
××××××××	×××	×××	××××

(第2図)

<顧客種別販売金額集計>

高等学校	専門学校	大学等	その他	合計
9,625,665	15,794,735	13,197,210	3,273,270	41,890,880

(Label1)

顧客種別

高等学校 (OptionButton1)

専門学校 (OptionButton2)

大学等 (OptionButton3)

その他 (OptionButton4)

合計 (OptionButton5)

分析項目

商品コード	商品名
101	カレンダー

(TextBox1) (Label2)

表示

クリア

終了

<顧客別販売分析>

顧客名	販売数量	販売金額	備考
△△専門学校	570	794,250	★★★★★★★★★★★★★★
▲▲専門学校	430	593,400	★★★★★★★★★★★★★★
}	}	}	}
●●専門学校	105	149,625	★★
合計	2,245	3,130,725	

(TextBox2)

(第3図)

処理条件

- 第1図の顧客情報データは顧客番号の昇順に記録されている。なお、顧客番号は1からの連番であり、顧客は500件以下である。また、顧客種別は1(高等学校)~4(その他)である。
- ユーザーフォーム初期化時に、次の処理を行う。
  - 配列 Scod に商品コードを、配列 Smei に商品名を、配列 Tanka に単価を記憶する。なお、Scod, Smei, Tanka の添字はそれぞれ対応している。
  - 配列 Wkijun に割引基準数量の上限を、配列 Writu に割引率を記憶する。なお、Wkijun と Writu の添字は対応している。

配列

<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Scod</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">~</td> <td style="width: 25%;">(9)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">101</td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;">701</td> </tr> </table>	Scod	(0)	(1)	~	(9)			101	~	701	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Wkijun</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">(2)</td> <td style="width: 25%;">(3)</td> <td style="width: 25%;">(4)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">49</td> <td style="border: 1px solid gray;">199</td> <td style="border: 1px solid gray;">499</td> <td style="border: 1px solid gray;">999</td> <td style="border: 1px solid gray;">9999</td> </tr> </table>	Wkijun	(0)	(1)	(2)	(3)	(4)		49	199	499	999	9999
Scod	(0)	(1)	~	(9)																			
		101	~	701																			
Wkijun	(0)	(1)	(2)	(3)	(4)																		
	49	199	499	999	9999																		
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Smei</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">~</td> <td style="width: 25%;">(9)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">カレンダー</td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;">マグカップ</td> </tr> </table>	Smei	(0)	(1)	~	(9)			カレンダー	~	マグカップ	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Writu</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">(2)</td> <td style="width: 25%;">(3)</td> <td style="width: 25%;">(4)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">0</td> <td style="border: 1px solid gray;">0.05</td> <td style="border: 1px solid gray;">0.08</td> <td style="border: 1px solid gray;">0.1</td> <td style="border: 1px solid gray;">0.12</td> </tr> </table> <p style="font-size: small;">(~49個) (~199個) (~499個) (~999個) (1,000個~)</p>	Writu	(0)	(1)	(2)	(3)	(4)		0	0.05	0.08	0.1	0.12
Smei	(0)	(1)	~	(9)																			
		カレンダー	~	マグカップ																			
Writu	(0)	(1)	(2)	(3)	(4)																		
	0	0.05	0.08	0.1	0.12																		
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Tanka</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">~</td> <td style="width: 25%;">(9)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">1500</td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;">1000</td> </tr> </table>	Tanka	(0)	(1)	~	(9)			1500	~	1000													
Tanka	(0)	(1)	~	(9)																			
		1500	~	1000																			

- 第1図の顧客情報データを読み、配列 Ksyu に顧客種別を、配列 Kmei に顧客名を記憶する。なお、Ksyu, Kmei の添字は顧客番号と対応している。

配列

Ksyu	Kmei
(0)	(0)
(1)	(1)
}	}
(500)	(500)

- 第2図の販売データを読み、次の処理を行う。
  - 商品コードをもとに配列 Scod を探索し、販売金額を次の計算式で求める。なお、販売数量は 9999 以下とする。  
**販売金額 = 販売数量 × 単価**
  - 販売数量をもとに配列 Wkijun を探索し、割引率に応じて販売金額から割引きを行う。
  - 配列 Susyu に販売数量を、配列 Kinsyu に販売金額を集計する。なお、Susyu と Kinsyu の行方向の添字は顧客番号と対応し、列方向の添字は配列 Scod の添字と対応している。

配列

<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Susyu</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">~</td> <td style="width: 25%;">(9)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">(0)</td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;"></td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">(1)</td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;"></td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">}</td> <td style="border: 1px solid gray;">}</td> <td style="border: 1px solid gray;">}</td> <td style="border: 1px solid gray;">}</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">(500)</td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;"></td> </tr> </table>	Susyu	(0)	(1)	~	(9)		(0)		~			(1)		~			}	}	}	}		(500)		~		<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Kinsyu</td> <td style="width: 25%;">(0)</td> <td style="width: 25%;">(1)</td> <td style="width: 25%;">~</td> <td style="width: 25%;">(9)</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">(0)</td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;"></td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">(1)</td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;"></td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">}</td> <td style="border: 1px solid gray;">}</td> <td style="border: 1px solid gray;">}</td> <td style="border: 1px solid gray;">}</td> </tr> <tr> <td></td> <td style="border: 1px solid gray;">(500)</td> <td style="border: 1px solid gray;"></td> <td style="border: 1px solid gray;">~</td> <td style="border: 1px solid gray;"></td> </tr> </table>	Kinsyu	(0)	(1)	~	(9)		(0)		~			(1)		~			}	}	}	}		(500)		~	
Susyu	(0)	(1)	~	(9)																																															
	(0)		~																																																
	(1)		~																																																
	}	}	}	}																																															
	(500)		~																																																
Kinsyu	(0)	(1)	~	(9)																																															
	(0)		~																																																
	(1)		~																																																
	}	}	}	}																																															
	(500)		~																																																

- 顧客種別ごとに配列 Ksyukei に販売金額を集計する。なお、Ksyukei(5) には合計を求める。また、Ksyukei の添字は顧客種別と対応している。

配列

Ksyukei	(0)	(1)	(2)	(3)	(4)	(5)
	(高等学校) (専門学校) (大学等) (その他) (合計)					

- データを読み終えたあと、第3図のように顧客種別販売金額集計を Label1 に表示する。
- 第3図のように顧客種別を指定し、TextBox1 に商品コードを入力したあと、「表示」ボタンをクリックすると、次の処理を行う。
    - Label2 に商品名を表示する。
    - 配列 Wk を利用して、指定した顧客種別、商品コードに該当する顧客を販売金額の降順に並べ替える。

配列

Wk	
(0)	
(1)	1
(2)	2
}	}
(500)	

- 販売がある顧客名から備考までを TextBox2 に表示する。なお、備考は、販売金額の 50000 ごとに ★ を表示する。
- 販売数量と販売金額の合計を TextBox2 に表示する。

## &lt;マクロ言語プログラム&gt;

```

Option Explicit
Dim Scod(9) As Long, Smei(9) As String, Tanka(9) As Long, Ksb As Long, Ksu As Long, Ksyu(500) As Long, Kmei(500) As String, h As Long
Dim Susyu(500, 9) As Long, Kinsyu(500, 9) As Long

Private Sub UserForm_Initialize()
Dim Wkijun(4) As Long, Writu(4) As Double, Kban As Long, Hnen As Long, Sco As Long, Hsu As Long, j As Long, Hkin As Long, Ksyukei(5) As Long, k As Long
Scod(1) = 101: ~ Scod(9) = 701: Smei(1) = "カレンダー": ~ Smei(9) = "マグカップ": Tanka(1) = 1500: ~ Tanka(9) = 1000
Wkijun(0) = 49: Wkijun(1) = 199: Wkijun(2) = 499: Wkijun(3) = 999: Wkijun(4) = 9999
Writu(0) = 0: Writu(1) = 0.05: Writu(2) = 0.08: Writu(3) = 0.1: Writu(4) = 0.12
Label1.Caption = ""
Call Syokika
Open ThisWorkbook.Path & "%kokyaku.csv" For Input As #1
Do While Not EOF(1)
Ksu = Ksu + 1
Input #1, Kban, Ksyu(Ksu), Kmei(Ksu)
Loop
Close #1
Open ThisWorkbook.Path & "%hanbai.csv" For Input As #2
Do While Not EOF(2)
Input #2, Hnen, Kban, Sco, Hsu
h = 1
Do While Scod(h) <> Sco
h = h + 1
Loop
j = 0
Do While Wkijun(j) < Hsu
j = j + 1
Loop
Hkin = Hsu * Tanka(h) * (1 - Writu(j))
Susyu(Kban, h) = Susyu(Kban, h) + Hsu
Ksyukei(Ksyu(Kban)) = Ksyukei(Ksyu(Kban)) + Hkin
Ksyukei(5) = Ksyukei(5) + Hkin
Loop
Close #2
For k = 1 To 5
Label1.Caption = Label1.Caption & Format(Format( (1) ), "##,###,##0"), "#####") & " "
Next k
End Sub

Private Sub 表示_Click()
Dim Sc As Long, s As Long, Wk(500) As Long, Work As Long, t As Long, Hosi As String, u As Long, Sukei As Long, Kinkei As Long
Label2.Caption = "": TextBox2.Text = ""
Sc = Val(TextBox1.Text)
h = 1
Do While Scod(h) <> Sc
h = h + 1
Loop
Label2.Caption = Smei(h)
For s = 1 To Ksu
Wk(s) = s
Next s
For s = 2 To Ksu
Work = Wk(s)
t = s
Do While t > 1 And Kinsyu(Wk(t - 1), h) < Kinsyu(Work, h)
(3)
t = t - 1
Loop
Wk(t) = Work
Next s
For s = 1 To Ksu
If Ksyu(Wk(s)) = Ksb Or (4) Then
If Kinsyu(Wk(s), h) > 0 Then
t = Int(Kinsyu(Wk(s), h) / 50000)
Hosi = ""
For u = 1 To t
Hosi = Hosi & "★"
Next u
TextBox2.Text = TextBox2.Text & Kmei(Wk(s)) & " " & _
Format(Format( (5) ), "##,##0"), "#####") & " " & _
Format(Format(Kinsyu(Wk(s), h), "#,###,##0"), "#####") & "-" & Hosi & Chr(13) & Chr(10)
Sukei = Sukei + Susyu(Wk(s), h)
Kinkei = Kinkei + Kinsyu(Wk(s), h)
Else
Exit For
End If
End If
Next s
TextBox2.Text = TextBox2.Text & " 合計 " & Format(Format(Sukei, "##,##0"), "#####") & " " & _
Format(Format(Kinkei, "#,###,##0"), "#####") & Chr(13) & Chr(10)
End Sub

Private Sub クリア_Click()
Call Syokika
End Sub

Private Sub 終了_Click()
End
End Sub

Private Sub OptionButton1_Click()
Ksb = 1
End Sub

Private Sub OptionButton5_Click()
Ksb = 5
End Sub

Private Sub Syokika()
OptionButton1.Value = True: Ksb = 1: TextBox1.Text = "": Label2.Caption = "": TextBox2.Text = ""
End Sub

```

(令和2年9月27日実施)

主催 公益財団法人 全国商業高等学校協会

## 令和2年度（第63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

【4】	(1)	(2)	(3)	(4)	(5)

【5】	(1)	(2)	(3)	(4)	(5)

【6】	(1)	(2)	(3)	(4)	(5)

小 計

..... 【Java】・【マクロ言語】 .....

【7】	(1)	
	(2)	
	(3)	
	(4)	
	(5)	

試験場校名	受験番号	選択言語		小 計	合 計
		Java	マクロ言語		

選択言語を  で囲むこと

(令和2年9月27日実施)

主催 公益財団法人 全国商業高等学校協会

## 令和2年度(第63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	各2点 計10点
	カ	コ	シ	イ	ク	

【2】	1	2	3	4	5	各2点 計10点
	エ	ケ	イ	キ	ア	

【3】	1	2	3	4	5	各2点 計10点
	ウ	イ	ア	ウ	イ	

小 計

30

【4】	(1)	(2)	(3)	(4)	(5)	各3点 計15点
	ウ	ア	オ	イ	エ	

【5】	(1)	(2)	(3)	(4)	(5)	各3点 計15点
	ア	カ	ケ	ウ	キ	

【6】	(1)	(2)	(3)	(4)	(5)	各3点 計15点
	ク	ア	コ	オ	エ	

小 計

45

..... 【Java】・【マクロ言語】 .....

【Java】 (注) =, 演算子の前後の空白は問わない。

【7】	(1)	<code>gSales[l][idx] += sFee</code>
	(2)	<code>wk[k] = wk[k - 1]</code>
	(3)	<code>gName[wk[j]]</code>
	(4)	<code>gTotal</code>
	(5)	<code>sKind[i] == 1</code>

【マクロ言語】 (注) 大文字, 小文字および=, 演算子の前後の空白は問わない。

【7】	(1)	<code>Kinsyu(Kban, h) = Kinsyu(Kban, h) + Hkin</code>
	(2)	<code>Ksyukei(k)</code>
	(3)	<code>Wk(t) = Wk(t - 1)</code>
	(4)	<code>Ksb = 5</code>
	(5)	<code>Susyuu(Wk(s), h)</code>

各5点 計25点

小 計

25

合 計

100