

Programing

java実習課題



実習課題 1	点数の入力と表示（入力・演算・表示）	2
実習課題 2	合計と平均（繰り返し処理・ダウントータル・件数カウント）	3
実習課題 3	割引料金の計算（条件判定・ダウントータル）	4
実習課題 4	ガソリン価格の最高値と最安値（最大値と最小値）	5
実習課題 5	成績の入力と集計（最大値と最小値）	6
実習課題 6	売上一覧表（CSVファイルの読み込み）	7
実習課題 7	売上一覧表（条件分岐）	8
実習課題 8	売上一覧表（ダウントータルと平均）	9
実習課題 9	売上一覧表（最大値と最小値）	10
実習課題 10	成績一覧表（実習 6～9 の復習）	11
実習課題 11	アパートの入居状況一覧表の作成（判定、最小値）	13
実習課題 12	人口変動一覧表の作成（判定、最大値）	14
実習課題 13	旭川空港の就航率一覧表の作成（判定、最大値、最小値）	15
実習課題 14	ビジネス文書検定一覧表の作成（判定、集計）	16
実習課題 15	スイーツの商品別売上一覧表の作成（1次元配列）	17
実習課題 16	北海道のおみやげ通販売上表の作成（1次元配列と線形探索）	18
実習課題 17	観光ホテル予約状況（1次元配列と線形探索）	19

() 年 () 組 () 番 氏名 ()

プログラミング実習の取り組み方と評価の基準

プログラミングとは？ ～学習の目的～

「プログラミング」とは、プログラミング言語を使用して、コンピュータに順番に指示を出すことによって、意図した動作をさせることです。

コンピュータが動作をするための設計をし、その設計がうまくいくかを確認し、うまく動作しなかった場合は誤りを修正して、動作を成功させるまでの一連の作業のことです。

- 私たちの身近で「プログラミング」が活用されているものでは、
- ・ ウェブサイト（Yahoo!、Google など）
 - ・ スマートフォンアプリ（LINE、Google マップなど）
 - ・ 企業システム（ATM、在庫管理システム、営業支援システム、顧客情報管理システム、配送管理システムなど）
 - ・ ゲーム
 - ・ 人工知能、車の自動運転
- などがあがあります。

これらにより多くの社会の課題が解決されてきました。それと同時に新たな課題も生まれてきています。それらの課題を解決するための1つの手段がプログラミングです。

プログラミングを学ぶ目的は、「生活や社会における問題を解決し、望ましい情報社会の構築し、ビジネスと社会の発展に貢献すること」です。

java の特徴

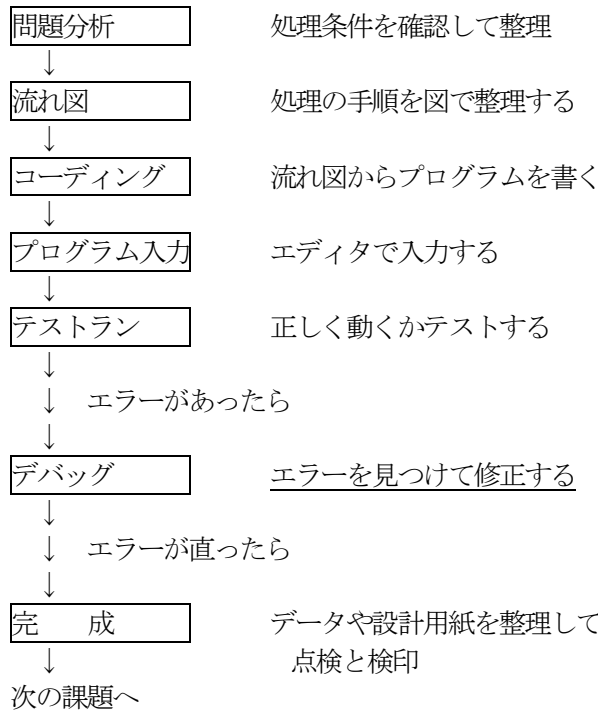
プログラミング言語には様々なものがあります。今回は、VM（バーチャルマシン）さえ稼働していれば、パソコンやスマートフォンやカーナビなどのさまざまな環境（プラットフォーム）で動かすことができる java 言語を使ってプログラミングを学んでいきます。

1 実習予定

- 4 月
- ・教科書例題を使い、「入力」「コンパイル」「実行」の手順を学び、簡単なエラーを自分で見つけて「デバッグ」できるようにする
- 5 月～9 月
- ・基本的な java の文法を学ぶ
 - ・実習課題 1～9
 - ・前期期末考査
- 10 月～1 月
- ・実習課題 10～17
- 2 月～3 月
- ・後期期末考査
 - ・実習課題の残り
 - ・来年の検定試験に向けて

2 実習の手順

プログラミングの実習はすべて次の手順で行います。手順を飛ばして完成させても、検印は押しません。理由は、前の段階の設計ができていないのに、次の段階ができるわけがないからです。



3 実習の評価

（1）評価全体の30%を実習で評価します。

例：考査が 80 点、実習が 40 点の場合
考査（70%） 80 点→56 点
実習（30%） 40 点→12 点
計 68 点の評価

- （2）実習の評価は次の基準で行います
- ・授業でやった実習課題まで完成しているか
 - ・完成とは左の「手順」をすべて飛ばさずに行っており以下のものがすべてそろっていることが条件です。

設計用紙	①出力設計	実行結果の表示を設計する
	②変数の設計	処理に使う変数名とデータ型を設計する
	③流れ図	処理手順を図に整理する
	④コーディング	上の①～③をもとに、java 言語でプログラムを記述する
データ	完成したプログラム	設計したプログラムを実際に入力し、正しい実行結果が得られるようにする

4 実習上の注意

- （1）全員の完成を待たずに次の課題へ進みます。新しい課題を一斉に説明するときには、自分がそこまで進んでいなくても、必要なことを書き込み、しっかり説明を聞いてください。
- （2）紙ファイルは、課題ごとに綴じて整理してください。またサーバ上のデータも正しい名前を付けて整理してください。
- （3）サーバに保存されているデータは、評価のもとになる大切なものです。ほかの人のフォルダやデータに触れることは一切禁止します。
- （4）エラーを直したらプログラムのデータを直すだけでなく、流れ図やコーディングなども直しておいてください。（実際のデータと紙に書いたものが一致していること）

実習課題1 点数の入力と表示 (入力・演算・表示)

キーボードから名前、国語、数学の成績を入力し合計と平均を表示する。

1 入力データ

名前	簿記	情報処理
××	××	××

2 実行結果

```
名前を入力してください
yasuda
簿記の点数を入力してください
80
情報処理の点数を入力してください
90
yasuda さんの点数は170
yasuda さんの平均は 85
```

3 处理条件

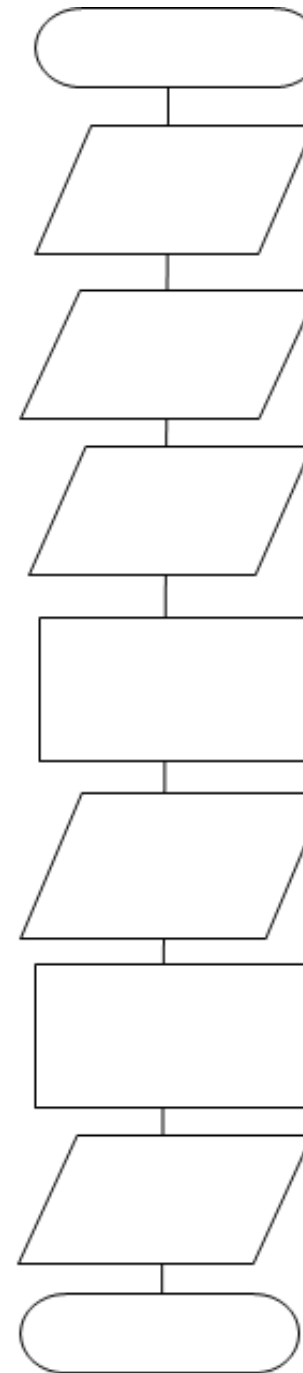
- (1) キーボードから名前を入力する。
- (2) 平均はつぎの式で計算する

$$\begin{aligned}\text{合計} &= \text{簿記の点数} + \text{情報処理の点数} \\ \text{平均} &= \text{合計} \div 2\end{aligned}$$

- (3) ただし、平均は小数点以下は切り捨てる
- (3) 名前、合計、平均を実行結果のようにディスプレイに表示する。

4 変数

5 流れ図とコーディング



```
import java.util.Scanner;
```

```
public class kadai1 {
```

```
public static void main(String[] args) {
```

}

実習課題2 合計と平均（繰り返し処理・ダウントータル・件数カウント）

キーボードから番号、国語、数学の成績を入力し合計と平均を表示する。

1 入力データ

番号	国語	数学
××	××	××

2 実行結果

```
番号を入力してください
1
国語を入力してください
80
数学を入力してください
90

----- (途中省略) -----

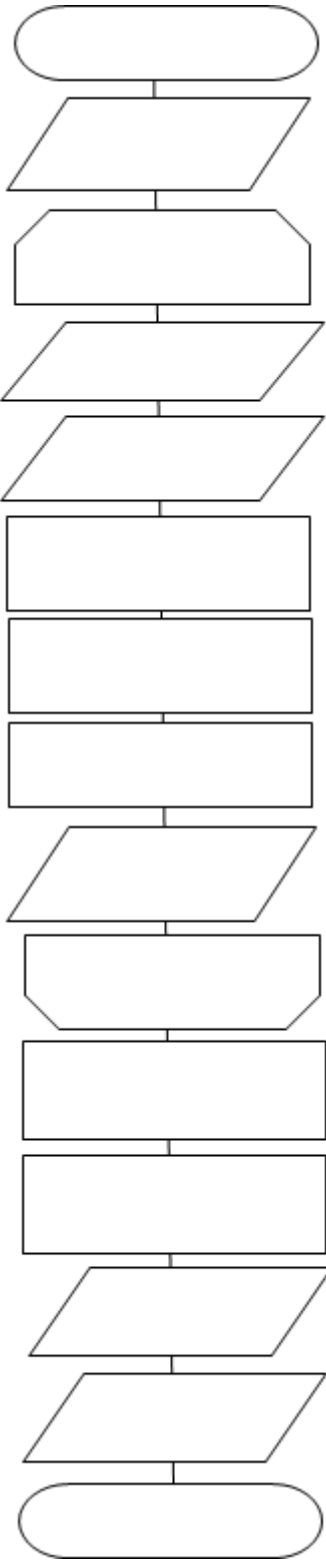
番号を入力してください
0

国語の合計    ××
数学の平均    ××
```

3 処理条件

1. キーボードから番号と国語・数学の点数を入力する。
2. 番号に0が入力されるまで処理を繰り返す
2. 処理を終了したら実行結果のように国語と数学の合計と平均をディスプレイに表示する。

4 変数



```
import java.util.Scanner;
public class kadai2{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int number = 0;
        int japanese = 0;
        int math = 0;
        int total_japanese = 0;
        int total_math = 0;
        int count = 0;

        while (number != 0) {
            System.out.println("番号を入力してください");
            number = scanner.nextInt();
            System.out.println("国語を入力してください");
            japanese = scanner.nextInt();
            System.out.println("数学を入力してください");
            math = scanner.nextInt();

            total_japanese += japanese;
            total_math += math;
            count++;
        }

        System.out.println("----- (途中省略) -----");

        System.out.println("番号を入力してください");
        number = scanner.nextInt();

        System.out.println("国語の合計    " + total_japanese);
        System.out.println("数学の平均    " + (total_math / count));
    }
}
```

実習課題3 割引料金の計算 (条件判定・ダウントータル)

年齢、料金をキーボードから入力し、正規料金合計、割引料金合計、総料金合計を表示する。

1 入力データ

年齢	料金
××	××

2 処理条件

- (1) キーボードからデータを入力し、999 で終了する。
- (2) 結果を実行結果のようにディスプレイに表示する。
12 歳以下は正規料金の半額とする。

3 実行結果

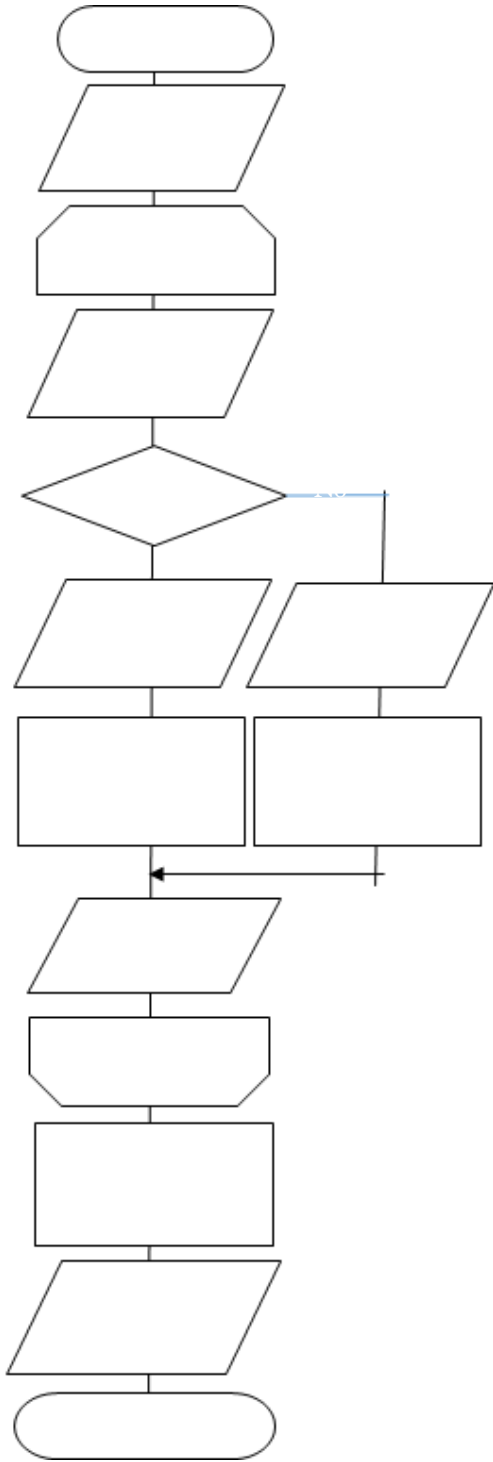
年齢を入力してください
16
料金を入力してください
500
正規料金です

年齢を入力してください
10
料金を入力してください
500
割引料金です

年齢を入力してください
999

正規料金合計 500
割引料金合計 250
総合計 750

4 変数



```
import java.util.Scanner;

public class kadai3 {

    public static void main(String[] args) {

        }

    }

}
```

実習課題4 ガソリン価格の最高値と最安値（最大値と最小値）

日付ごとのガソリン価格のデータをキーボードから入力し、最高値と最安値およびその日付を表示する。

1 入力データ

日	ガソリン価格
hi	kakaku

2 実行結果：例

日付を入力：1
価格を入力：160
日付を入力：2
価格を入力：161

Σ (途中省略)

日付を入力：30
価格を入力：159
日付を入力：99

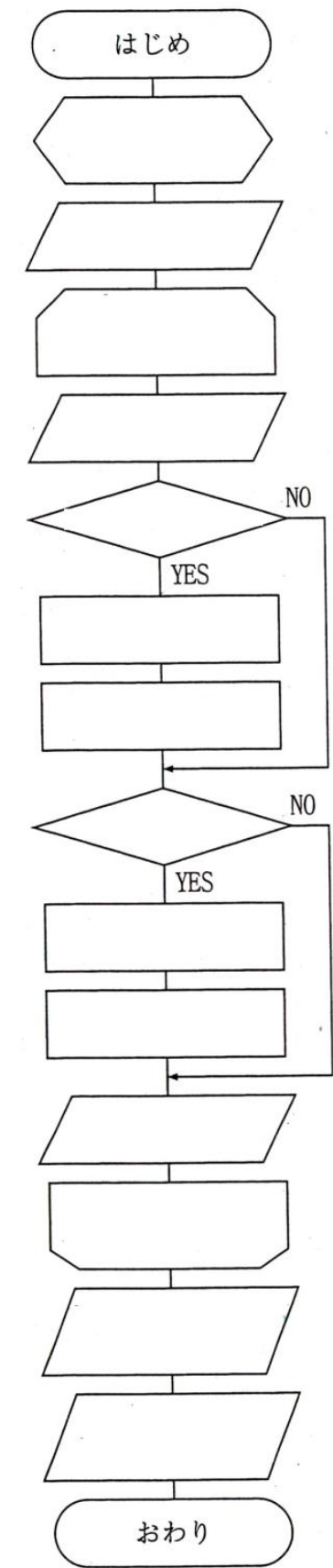
(最高値) 2日 161円
(最安値) 28日 158円

3 処理条件

- (1) 入力データは1リットルあたりの価格である。
(2) 日付に99が入力されたら、最高値と最安値を記録した日付、最高値と最安値を表示する。
最高値または最安値を記録した日が複数ある場合には、後から入力した日付を表示する。

4 変数

5 流れ図とコーディング



```
import java.util.Scanner;

public class kadai4{

    public static void main( String[] args ){

        Scanner scanner = new Scanner(System.in);

        String date;
        int price;

        int maxPrice = 0;
        int minPrice = 1000;
        String maxDate = "";
        String minDate = "";

        while (true) {
            System.out.print("日付を入力してください(99で終了)：");
            date = scanner.next();

            if (date.equals("99")) {
                break;
            }

            System.out.print("価格を入力してください：");
            price = scanner.nextInt();

            if (price > maxPrice) {
                maxPrice = price;
                maxDate = date;
            }

            if (price < minPrice) {
                minPrice = price;
                minDate = date;
            }
        }

        System.out.println("最高値：" + maxPrice + "円 (" + maxDate + "日)");
        System.out.println("最安値：" + minPrice + "円 (" + minDate + "日)");
    }
}
```

実習課題5 成績の入力と集計（最大値と最小値）

番号」「点数」をキーボードから入力し、合計点・平均点・最高点・最低点・成績優秀者の人数を求めて実行結果のように表示する。

1 入力データ

番号	点数
ban	ten

2 実行結果：例

番号を入力：1
点数を入力：100
番号を入力：2
点数を入力：70
番号を入力：3
点数を入力：40
番号を入力：4
点数を入力：80
番号を入力：5
点数を入力：79
番号を入力：99

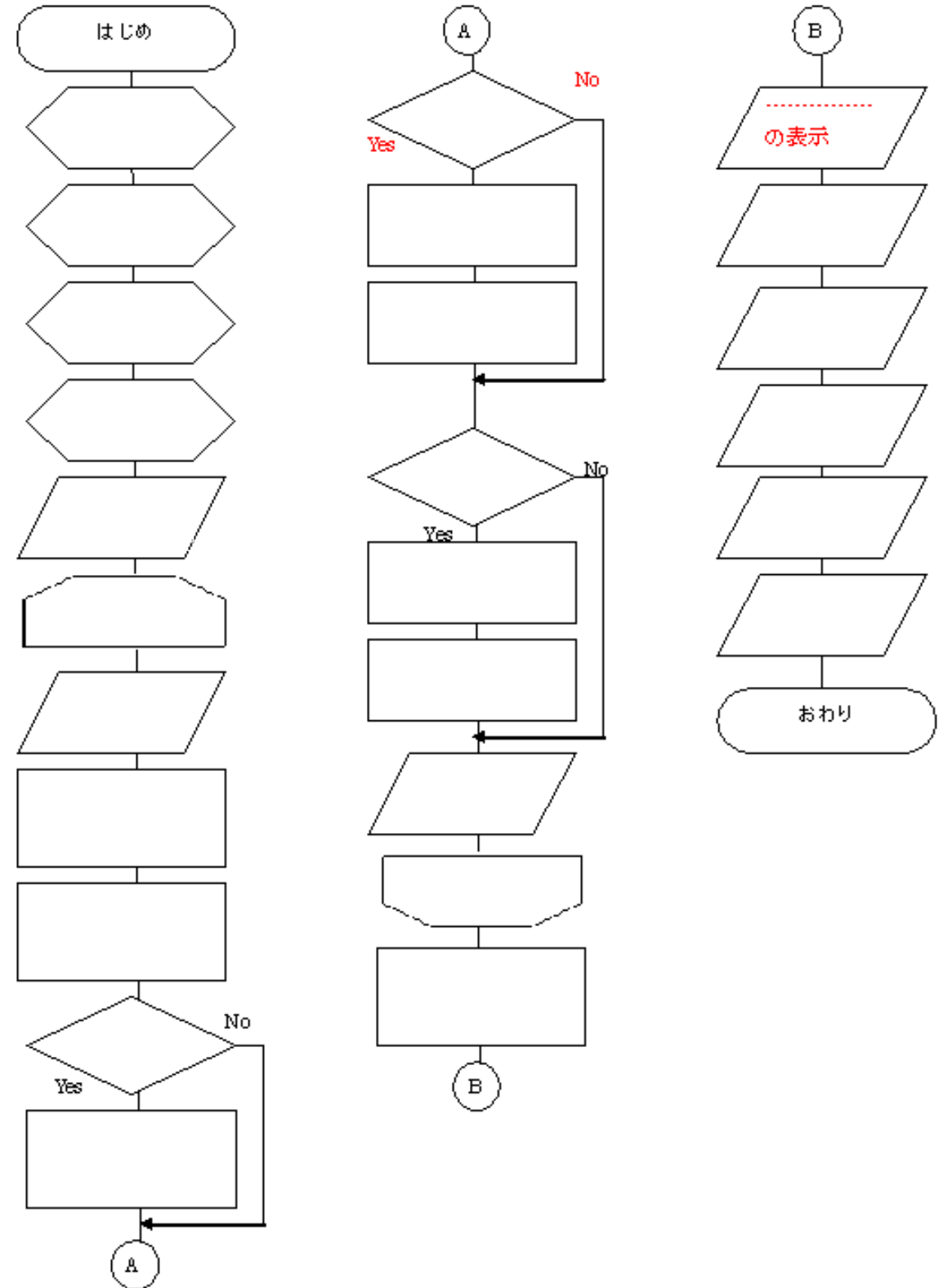
(合計) 369点
(平均) 73.0点
(最高) 1番100点
(最低) 3番40点
(成績優秀者) 2人

3 処理条件

- (1) 80点以上は成績優秀者としてカウントする
- (2) 最高点・最低点で同点だった場合は、最初に入力されたデータを最高・最低とする
- (3) 番号に99が入力されたら処理を終わる

4 変数

5 流れ図



実習課題6 売上一覧表 (CSVファイルの読み込み)

支店ごとの2カ月分の売上データを読み、1月と2月の合計を求め、実行結果のように表示する。

1 入力データ

ファイル名 : uriage.csv

番号	支店名	1 月売上	2 月売上
×	×××××	×××	×××

2 実行結果

売上一覧表		作成者×××	
No	店舗名	1 月	2 月 合計
1	曙支店	100	90 190
2	神楽支店	80	100 180
3	神楽岡支店	100	140 240
4	春光支店	60	60 120
5	永山支店	75	70 145
6	末広支店	80	90 170
7	東光支店	90	90 180
8	忠和支店	100	100 200

3 处理条件

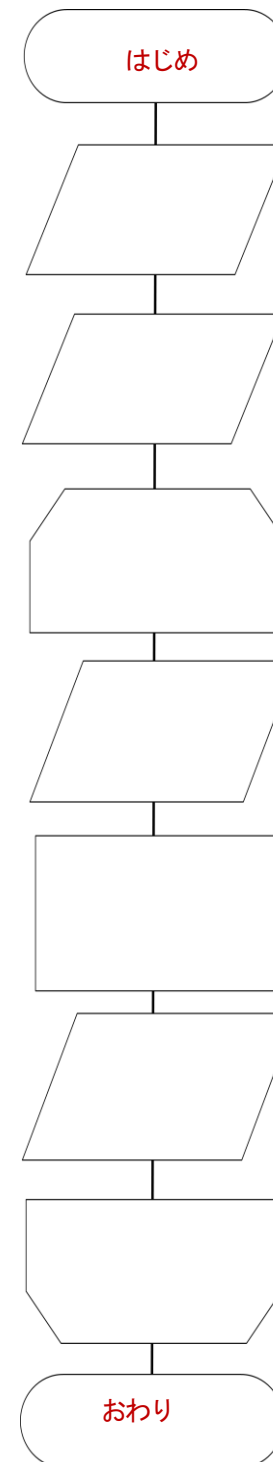
- (1) 売上データ (uriage.csv) を読み込んで変数に記憶する
- (2) 1月と2月の売上から合計を計算し、売上一覧表を表示する
- (3) データにエラーはないものとする

4 出力設計

[illegible]

5 変数

6 流れ図とコーディング



```
import java.io.*;
import java.util.Scanner;

public class ka6{

    public static void main(String[] args) throws
```


}

実習課題7 売上一覧表 (条件分岐)

支店ごとの2カ月分の売上データを読み、1月と2月の合計を求め、実行結果のように表示する。

1 入力データ

ファイル名 : uriagr.csv

番号	支店名	1 月売上	2 月売上
×	×××××	×××	×××

2 実行結果

売上一覧表			作成者×××		
No	店舗名	1 月	2 月	合計	判定
1	曙支店	100	90	190	減少
2	神楽支店	80	100	180	増加
3	神楽岡支店	100	140	240	増加
4	春光支店	60	60	120	増加
5	永山支店	75	70	145	減少
6	末広支店	80	90	170	増加
7	東光支店	90	90	180	同じ
8	忠和支店	100	100	200	同じ

3 处理条件

- (1) 売上データ (uriage.csv) を読み込んで変数に記憶する
- (2) 1月と2月の売上から合計を計算し、売上一覧表を表示する
- (3) データにエラーはないものとする
- (4) 「判定」は1月より2月が多ければ「増加」、1月より2月が少なければ「減少」
1月と2月が同額の場合は「同じ」と表示する

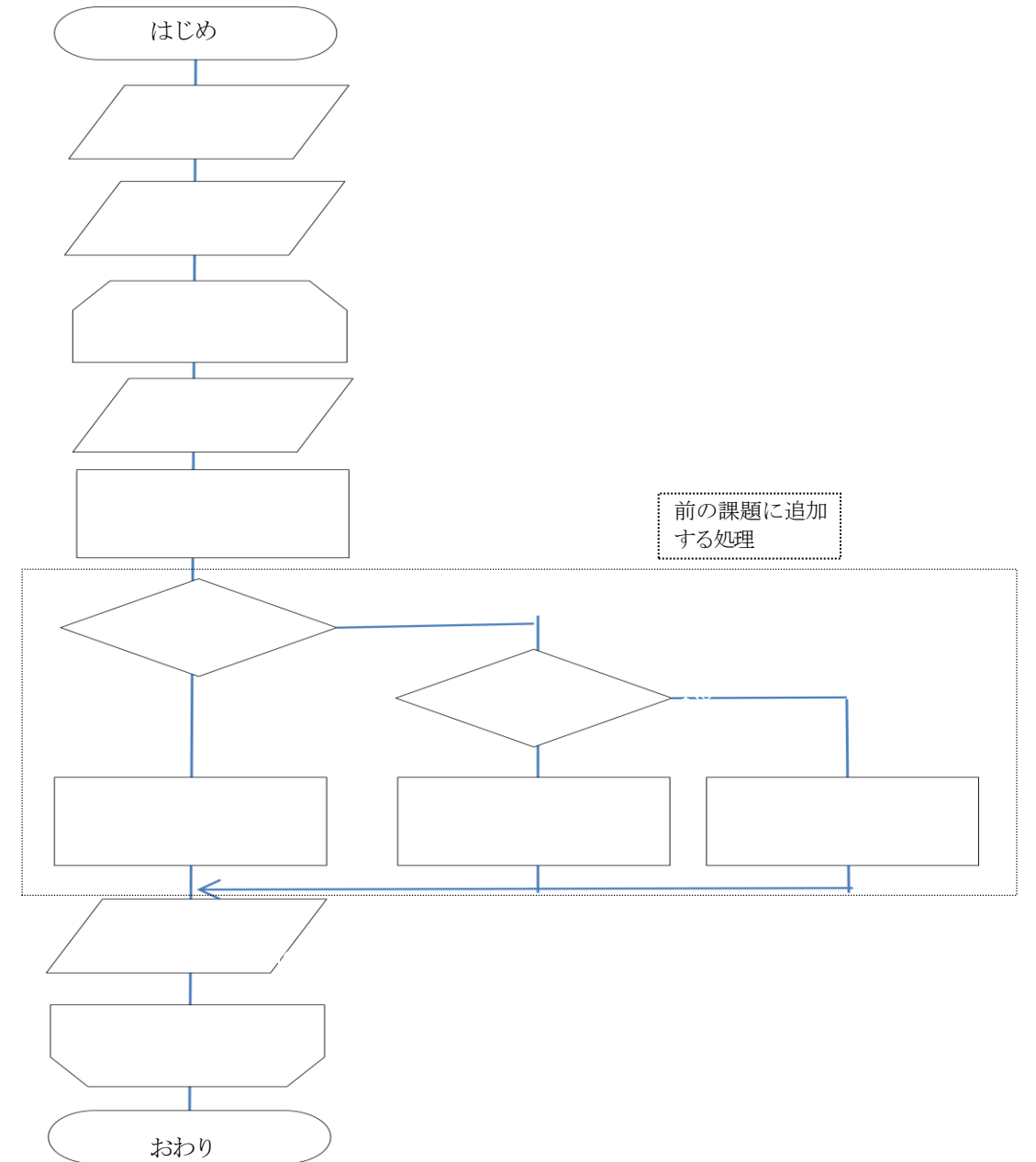
4 出力設計

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200
201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300
301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400
401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500
501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	52																																																																													

5 変数

[illegible]

6 流れ図とコーディング



追加部分のコーディング

[illegible]

支店ごとの2カ月分の売上データを読み、1月と2月の合計を求め、実行結果のように表示する。
データを読み終わったら月ごとの合計と平均を表示する。

番号	支店名	1月売上	2月売上
×	×××××	×××	×××

売上一覧表		作成者×××		
No	店舗名	1 月	2 月	合計 判定
1	曙支店	100	90	190 減少
2	神楽支店	80	100	180 増加
8	忠和支店	100	100	200 同じ
合計		685	740	1,425
平均		85	92	178

- (1) 売上データ (uriage.csv) を読み込んで変数に記憶する
- (2) 1月と2月の売上から合計を計算し、売上一覧表を表示する
- (3) データにエラーはないものとする
- (4) 「判定」は1月より2月が多ければ「増加」、1月より2月が少なければ「減少」
1月と2月が同額の場合は「同じ」と表示する
- (5) データを読み終わったら、「1月」「2月」「合計」の「全支店の合計」と「全支店の平均」を表示する

Figure 1 displays a grid of 300 small plots (6 rows by 50 columns) showing the evolution of the number of nodes in a network over 50 time steps for different values of α (1 to 50). Each plot has a y-axis from 0 to 100 and an x-axis from 0 to 50. The plots show a general upward trend in the number of nodes over time, with some fluctuations. The plots are arranged in 6 rows and 50 columns. The first row shows α values from 1 to 50. The second row shows α values from 1 to 50. The third row shows α values from 1 to 50. The fourth row shows α values from 1 to 50. The fifth row shows α values from 1 to 50. The sixth row shows α values from 1 to 50.

---	---

The image shows a blank flowchart template with two vertical paths. The left path starts with an oval labeled 'はじめ' (Start) and ends with an oval labeled 'A'. The right path starts with a circle labeled 'A', branches into two parallel paths, and ends with an oval labeled 'おわり' (End).

実習課題9 売上一覧表 (最大値と最小値)

支店ごとの2カ月分の売上データを読み、1月と2月の合計を求め、実行結果のように表示する。
データを読み終わったら月ごとの合計と平均と、1月と2月の合計の最高額と最低額を表示する。

1 入力データ (ファイル名: uriage.csv)

番号 ×	支店名 ×××××	1 月売上 ×××	2 月売上 ×××
---------	--------------	--------------	--------------

2 実行結果

売上一覧表		作成者××××			
No	店舗名	1 月	2 月	合計	判定
1	曙支店	100	90	190	減少
8	忠和支店	100	100	200	同じ
合計		685	740	1,425	
平均		85	92	178	
最高				240	
最低				120	

3 处理条件

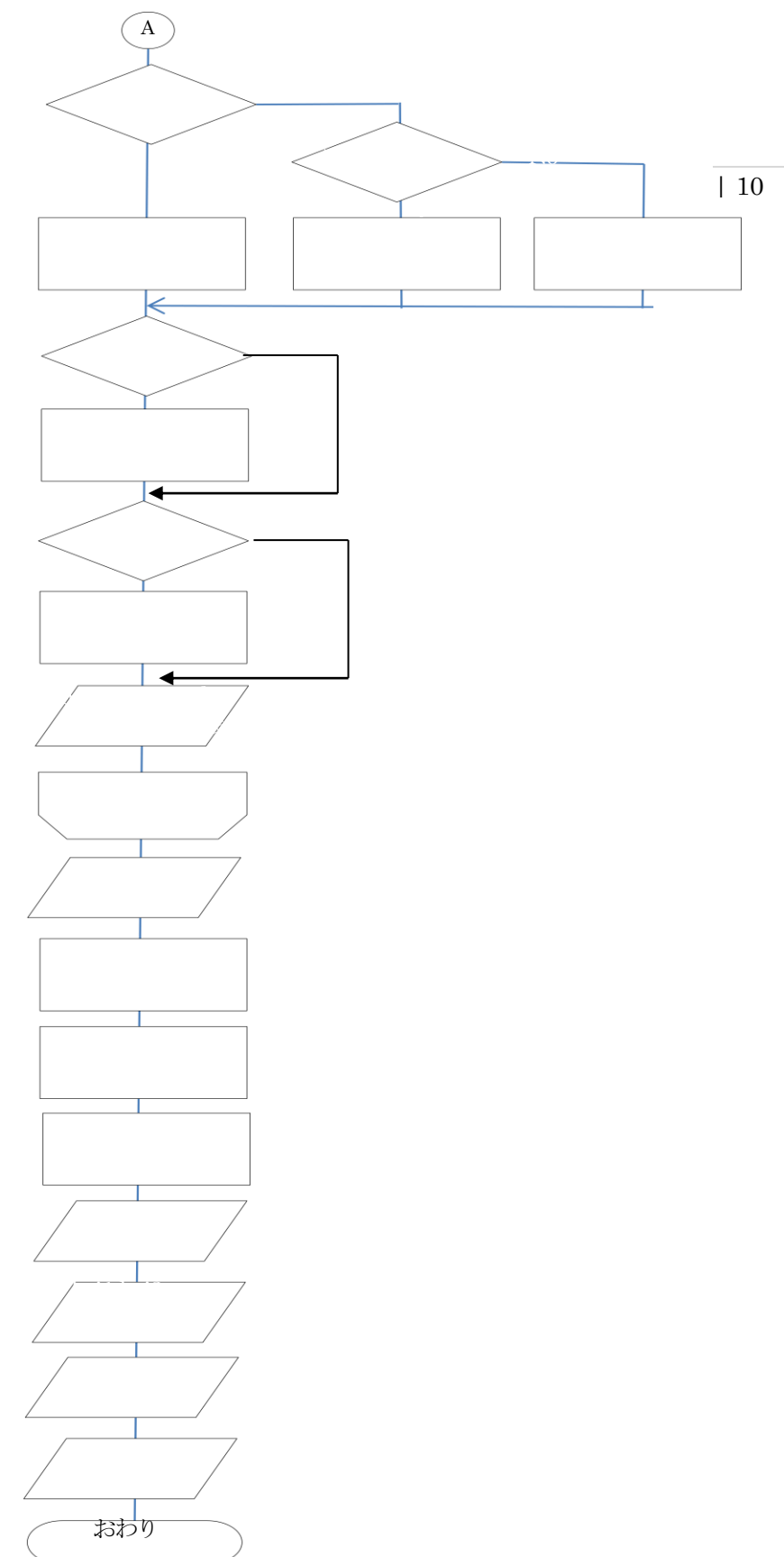
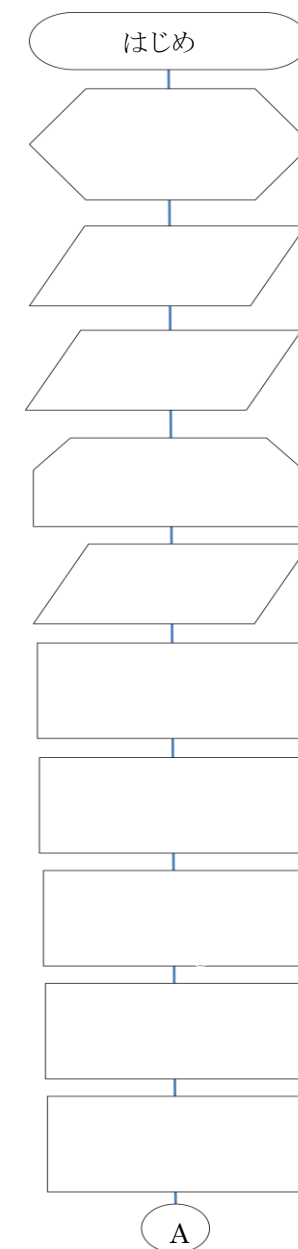
- (1) 売上データ (uriage.csv) を読み込んで変数に記憶する
- (2) 1月と2月の売上から合計を計算し、売上一覧表を表示する
- (3) データにエラーはないものとする
- (4) 「判定」は1月より2月が多ければ「増加」、1月より2月が少なければ「減少」
1月と2月が同額の場合は「同じ」と表示する
- (5) データを読み終わったら、「1月」「2月」「合計」の「全支店の合計」と「全支店の平均」を表示する
- (6) 「各支店の1月と2月の合計」の最高額と最低額を求めて表示する

4 出力設計

[illegible]

5 変数

6 流れ図



番号、氏名、ビ基、簿記の点数データを読み、実行結果のように表示する。
データを読み終わったら科目ごとの合計と平均と、最高点と最低点をを表示する。

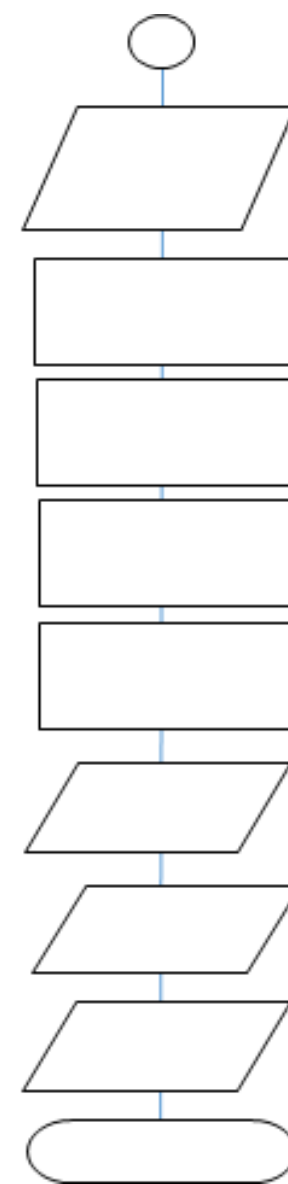
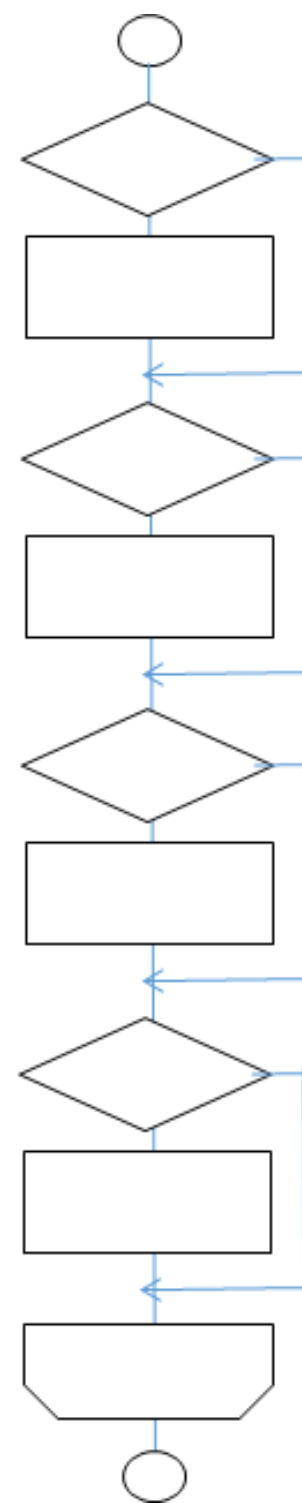
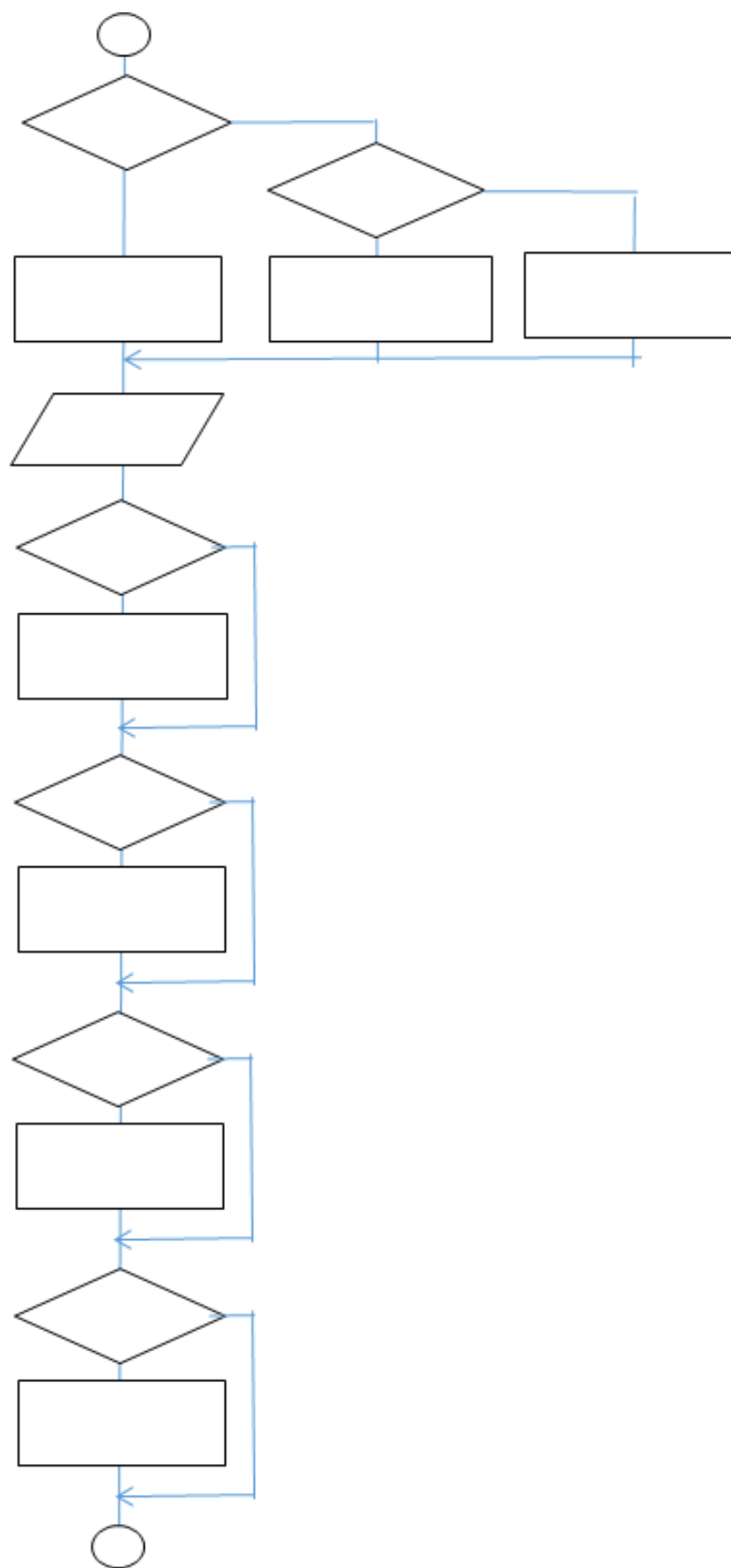
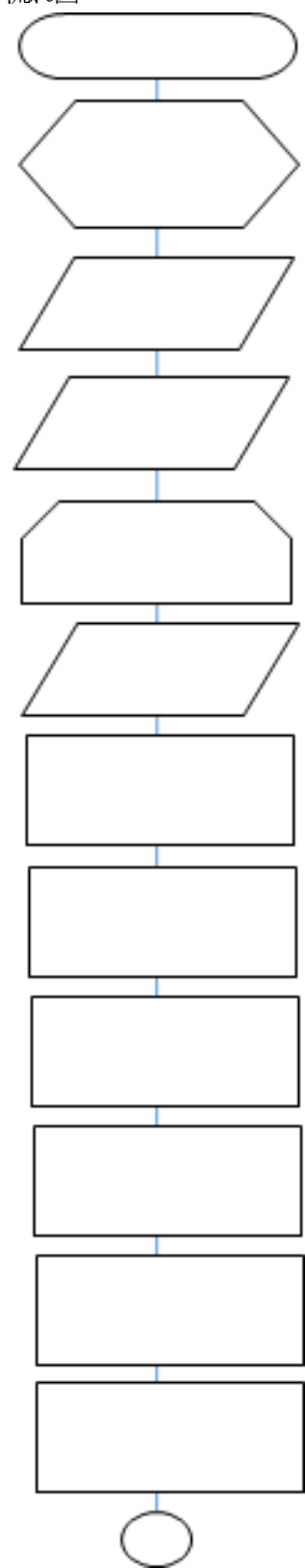
番号	氏名	ビ基	簿記	情処
×	×××××	×××	×××	×××

成績一覽表			作成者×××					
番号	氏 名	ビ基	簿記	情処	合計	平均	判定	
1	井上 奈月	87	94	99	280	93.3	A	
2	大久保 知史	47	45	43	135	45.0	C	
3	大坪 由宇	51	61	87	199	66.3	C	
4	城戸 広司	78	46	74	198	66.0	C	
5	倉持 真希	44	54	57	155	51.7	C	
6	小寺 えみ	72	64	99	235	78.3	B	
7	篠田 くるみ	54	76	63	193	64.3	C	
8	柴崎 さやか	61	44	87	192	64.0	C	
9	島 しぼり	51	54	85	190	63.3	C	
10	杉浦 朝香	88	80	51	219	73.0	B	
11	田口 ジョージ	77	59	57	193	64.3	C	
12	西野 恵子	71	86	82	239	79.7	B	
13	広末 麗奈	54	100	89	243	81.0	A	
14	藤井 まさみ	98	55	54	207	69.0	C	
15	柳田 光博	47	54	60	161	53.7	C	
	合 計	980	972	1,087	3,039			
	平 均	65.3	64.8	72.5	202.6			
	最高点	98	100	99	280			
	最低点	44	44	43	135			

- (1) 成績データ (seiseki.csv) を読み込んで変数に記憶する
- (2) 3科目の点数から合計と平均を計算し、成績一覧表を表示する
- (3) データにエラーはないものとする
- (4) 「判定」は平均が80点以上ならば「A」、70点以上ならば「B」、それ以外は「C」とする
- (5) データを読み終わったら、科目ごとの「合計」「平均」「最高点」「最低点」を表示する

[illegible]

6 流れ図



実習課題 11 アパートの入居状況一覧表の作成 (判定、最小値)

賃貸アパートの入居状況が記録された CSV データを読み、実行結果のように表示する。

1 入力データ（ファイル名：ap.csv）

建物	戸数	空戸数
----	----	-----

2 実行結果

建物	戸数	空戸数	入居率	備考
グリーンコーポ	10	2	80.0	
1条ハイツ	12	1	91.7	*
ファミリー曙	16	3	81.3	
サニーコート	16	1	93.8	*
メゾン・グランデ	20	5	75.0	
インベリアルコート	24	2	91.7	*
シャルール・メゾン	64	3	95.3	*
合計	162	17		
最低入居率	75.0			
最低入居率の建物名	メゾン・グランデ			

3 处理条件

(1) 入力データを読み、入居率(%)を次の計算式で求め、実行結果のように表示する。
 なお、備考は入居率(%)が90以上の場合は*を表示する。

$$\text{入居率}(\%) = (\text{戸数} - \text{空戸数}) \times 100 \div \text{戸数}$$

(2) 入力データが終了したら、戸数と空戸数の合計、最低入居率(%)、最低入居率(%)の建物名を表示する。
 なお、最低は同じ入居率(%)があった場合、先に入力されたデータを優先する。

(3) データにエラーはないものとする。

4 出力設計

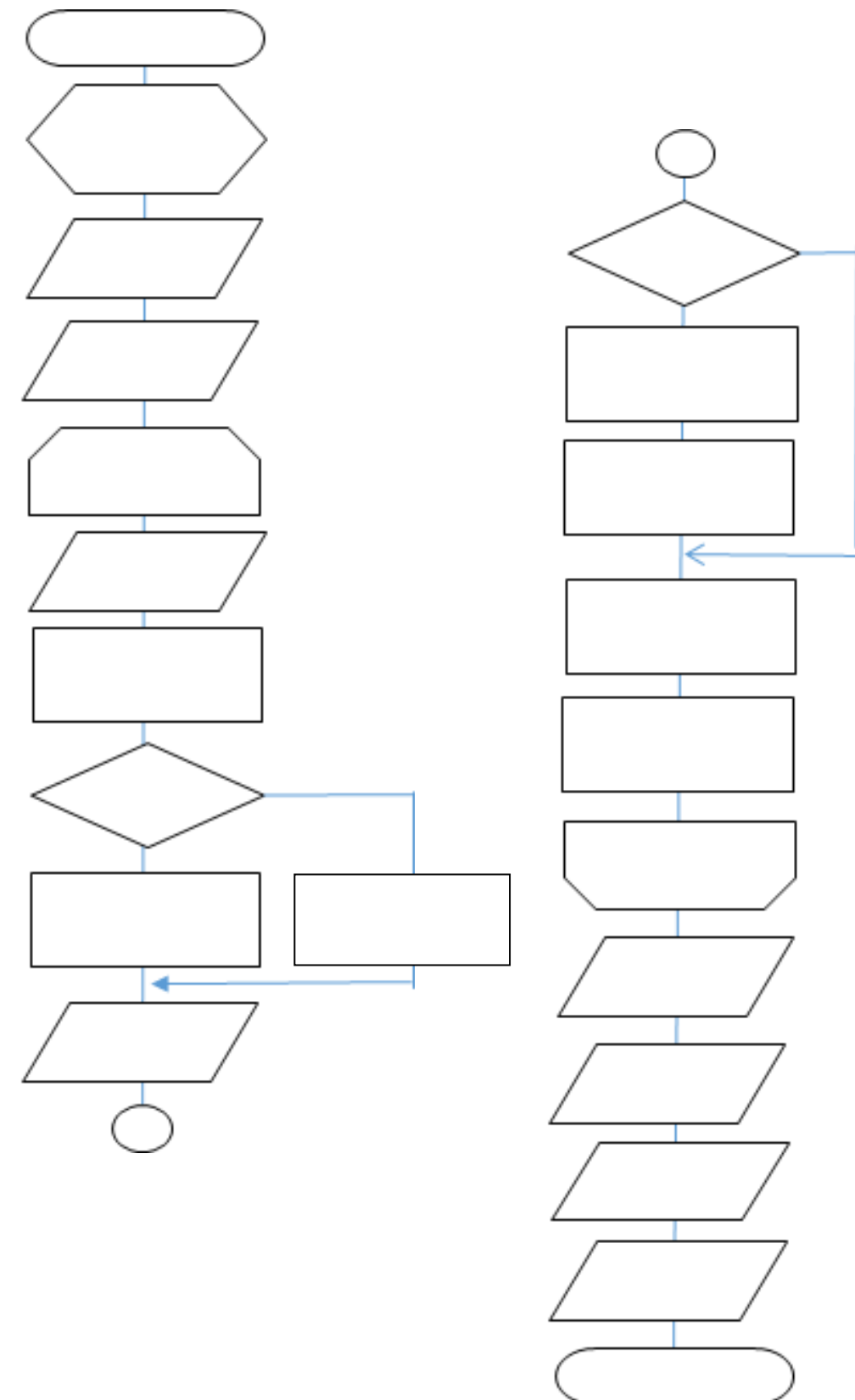
The figure displays a 6x10 grid of 60 small bar charts, each representing a different country. Each chart has a vertical axis with a scale from 0 to 100 and a horizontal axis with a single bar representing the country's value. The bars are colored in a repeating pattern of blue, green, and red. The countries are listed in the following order (row by row):

- Row 1: Afghanistan, Albania, Algeria, Argentina, Armenia, Australia, Austria, Azerbaijan, Bahrain, Bangladesh, Belgium, Bolivia
- Row 2: Bosnia and Herzegovina, Brazil, Bulgaria, Cambodia, Canada, Chile, China, Colombia, Costa Rica, Czech Republic, Denmark, Dominican Republic
- Row 3: Ecuador, Egypt, El Salvador, Estonia, Finland, France, Germany, Greece, Guatemala, Honduras, Hungary, India
- Row 4: Indonesia, Iran, Iraq, Israel, Italy, Japan, Jordan, Kazakhstan, Kenya, Korea, Kuwait, Kyrgyzstan
- Row 5: Laos, Latvia, Lebanon, Lithuania, Luxembourg, Macedonia, Malaysia, Maldives, Mali, Malta, Mexico, Moldova
- Row 6: Monaco, Mongolia, Montenegro, Morocco, Myanmar, Nepal, Netherlands, New Zealand, Nicaragua, Norway, Oman, Pakistan

5 変数

---	---

6 流れ図



実習課題 12 人口変動一覧表の作成 (判定、最大値)

北海道内の市町村人口の CSV データを読み、実行結果のように表示する。

1 入力データ (ファイル名 : jinko.csv)

市町村名	2014 年の人口	2019 年の人口
------	-----------	-----------

2 実行結果

人口変動一覧			
市町村名	2014 年	2019 年	増加率
ニセコ町	4, 857	5, 298	9. 1
倶知安町	15, 816	16, 642	5. 2
S			
東神楽町	10, 050	10, 321	2. 7
東川町	7, 948	8, 382	5. 5
留寿都村	1, 905	2, 047	7. 5

増加率が最大の市町村名 占冠村
 最大増加率 23. 6
 市町村数 10

3 处理条件

(1) 入力データを読み, 増加率 (%) を次の計算式で求め, 実行結果のように表示する。

$$\text{増加率(\%)} = (\text{2019 年} - \text{2014 年}) \times 100 \div \text{2014 年度}$$

(2) 入力データが終了したら, 増加率(%)が最大の市町村名, 増加率(%)の最大, 市町村数を表示する。

なお、最大は同じ増加率(%)があった場合、先に入力されたデータを優先する。

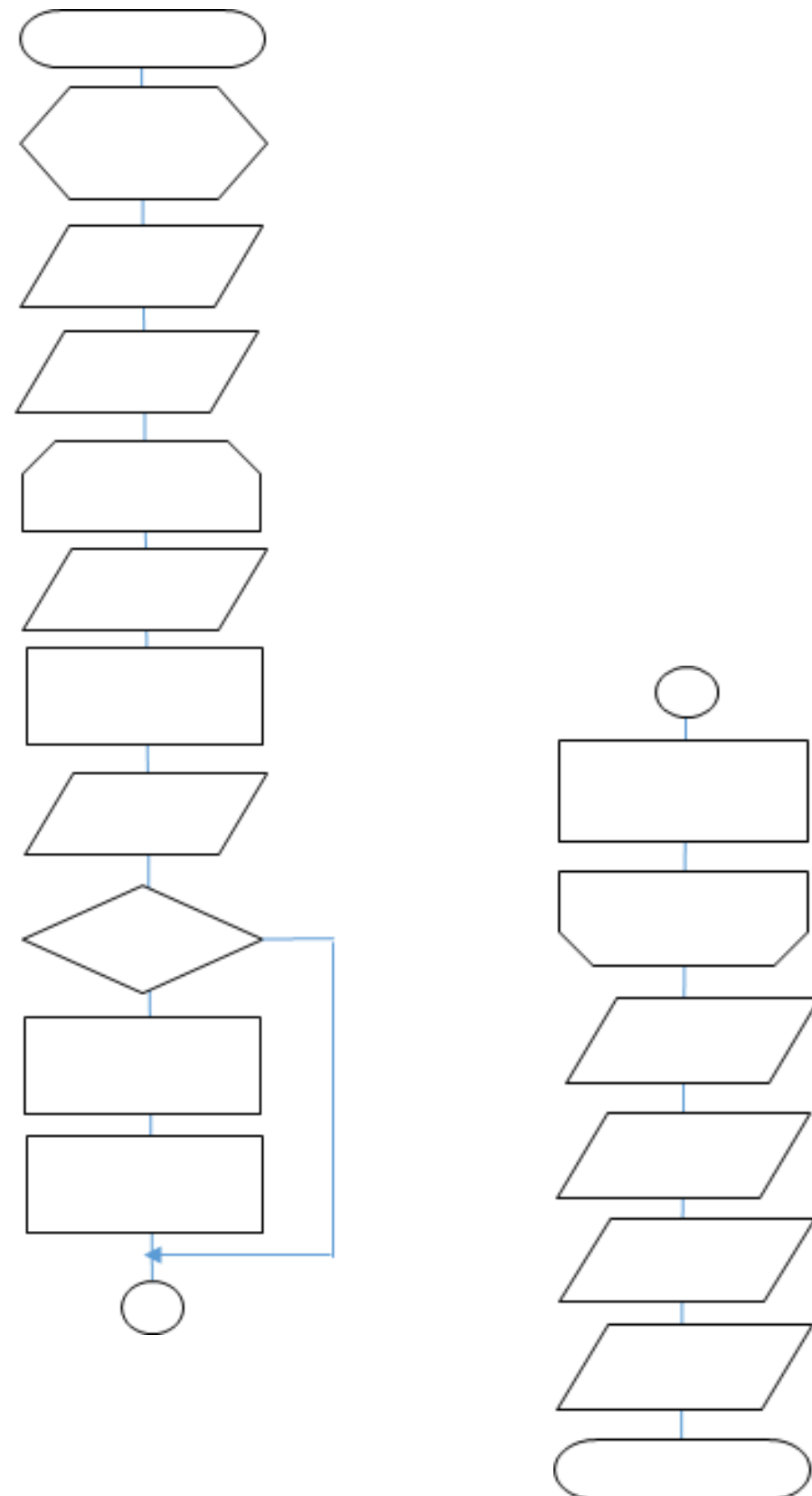
(3) データにエラーはないものとする。

4 出力設計

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200
201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300
301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400
401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500
501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	52																																																																													

5 変数

6 流れ図



実習課題 13 旭川空港の就航率一覧表の作成 (判定、最大値、最小値)

旭川空港の就航状況が記録された CSV データを読み、実行結果のように表示する。

1 入力データ (ファイル名 : syuuko.csv)

年	便数	欠航
---	----	----

2 実行結果

旭川空港の就航率				
年	便数	運航	欠航	就航率
H20	9,810	9,694	116	98.8
H21	8,982	8,936	46	99.5 ★
H22	10,734	10,631	103	99.0 ★
H23	9,418	9,324	94	99.0 ★
H24	8,600	8,526	74	99.1 ★
H25	7,850	7,750	100	98.7
H26	7,180	7,034	146	98.0

就航率が最高の年 H21 就航率 99.5
 就航率が最低の年 H26 就航率 98.0
 就航率99%以上の年 4回

3 处理条件

(1) 入力データを読み,就航率 (%)を次の計算式で求め,実行結果のように表示する。

$$\text{就航率}(\%) = \text{運航} \times 100 \div \text{便数}$$

(2) 就航率が 99%以上の年には★印を表示する

(3) 入力データが終了したら、就航率(%)が最高の年とその年の就航率(%)、就航率(%)が最低の年とその年の就航率(%)を表示する。

なお、最大は同じ増加率(%)があった場合、先に入力されたデータを優先する。

(4) 最後に就航率が 99%以上の年をカウントして表示する。

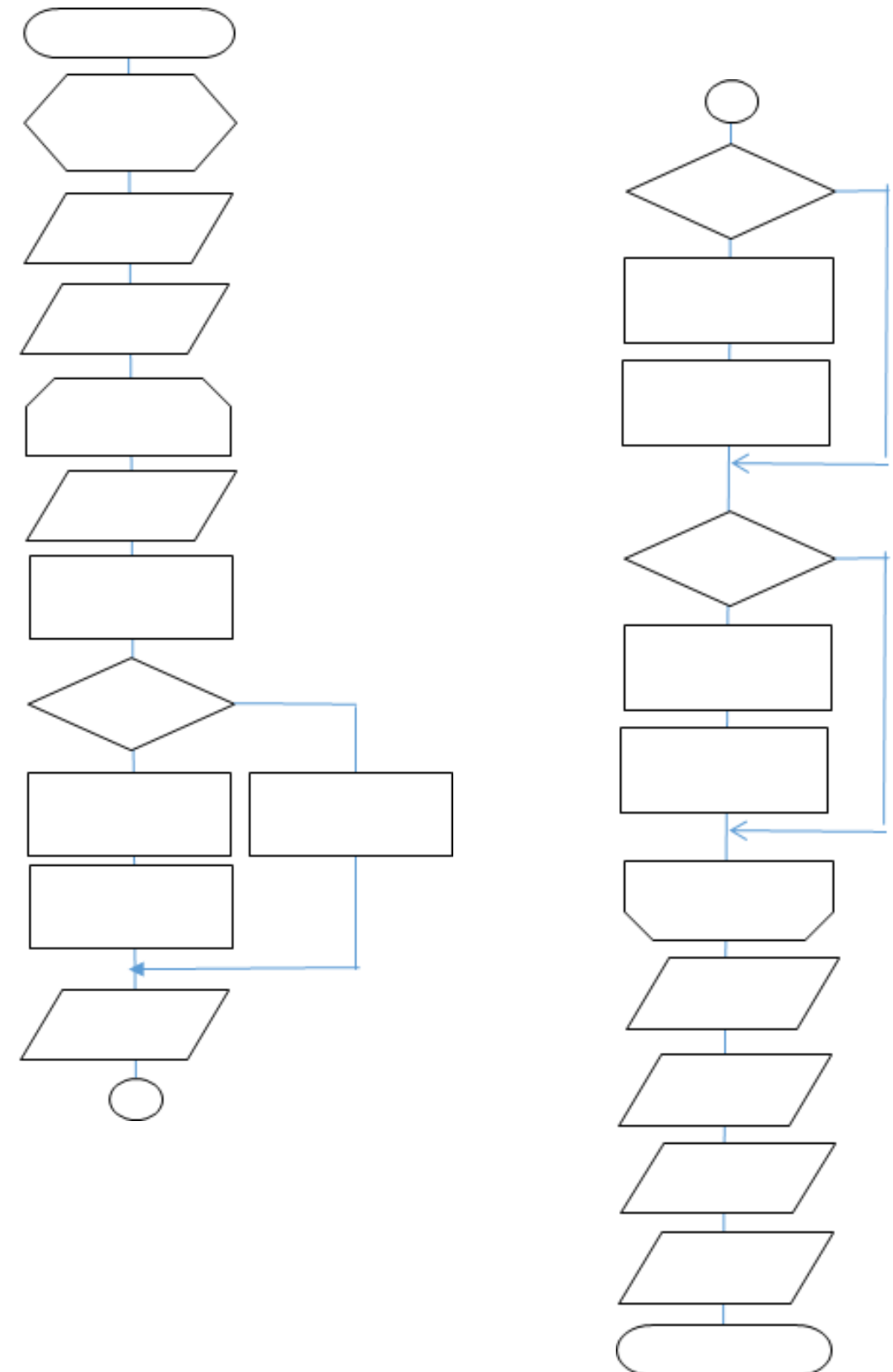
(5) データにエラーはないものとする。

4 出力設計

[illegible]

5 変数

6 流れ図



実習課題 14 ビジネス文書検定一覧表の作成 (判定、集計)

ビジネス文書検定の成績が記録された CSV データを読み、実行結果のように表示する。

1 入力データ（ファイル名：wpdata.csv）

番号	氏名	筆記	文書	速度
----	----	----	----	----

2 実行結果

ビジネス文書検定成績				作成者×××	
番号	氏名	筆記	文書	速度	判定
1	井上 奈月	86	95	460	2級合格
2	大久保 知史	82	95	460	2級合格
3	大坪 由宇	78	100	460	2級合格
S					
13	広末 麗奈	90	80	460	2級合格
14	藤井 まさみ	96	100	459	2級合格
15	柳田 光博	76	40	358	不合格

2級合格者 10 名(合格率 66.7 パーセント)
2級文書部門合格者 2 名
2級速度部門合格者 1 名

3 处理条件

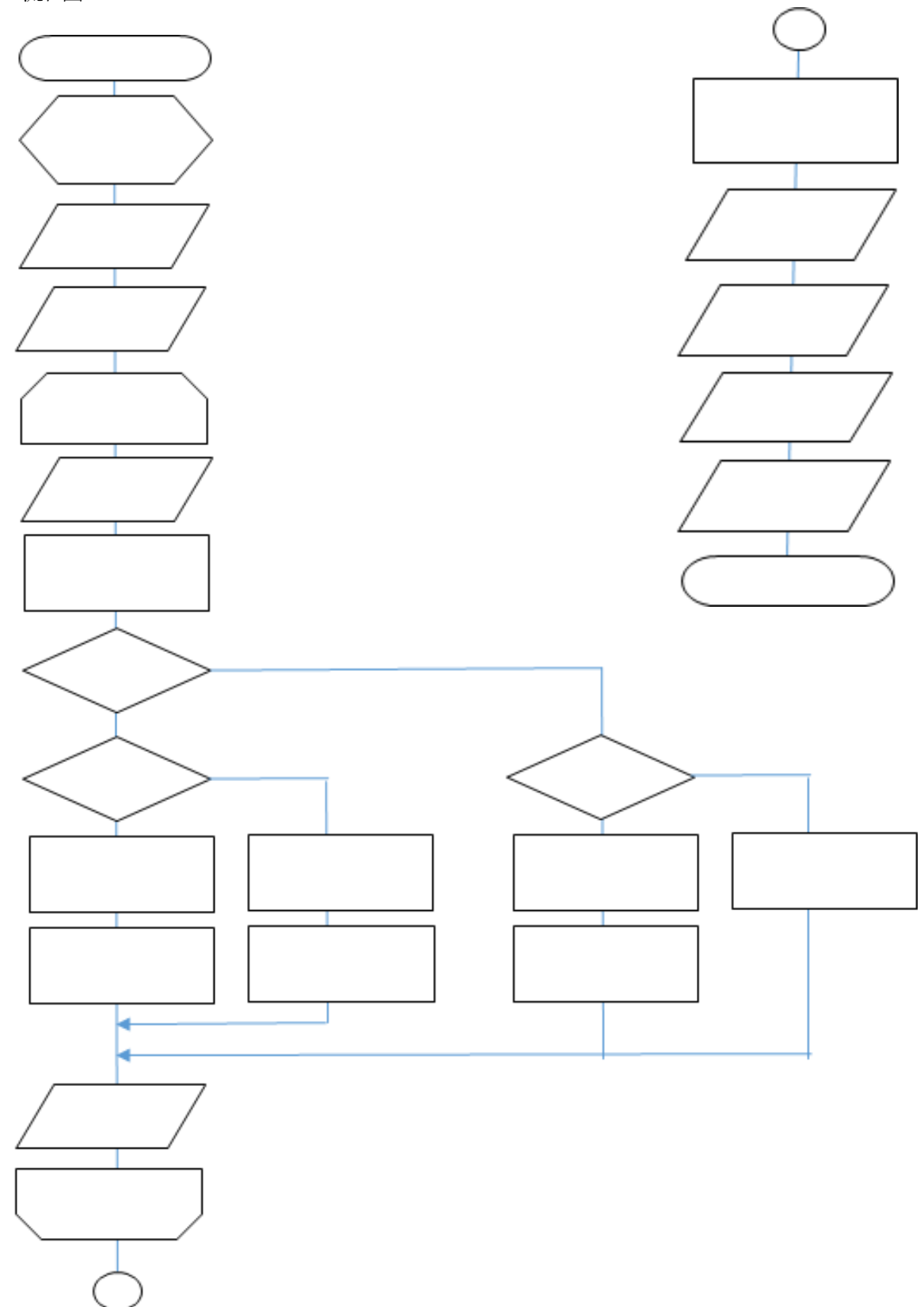
- (1) 入力データを読み、実行結果のように表示する。
- (2) 「判定」の基準は、次のとおりである。
 - ①筆記と文書がともに70点以上で文書部門合格
 - ②速度が450文字以上で速度部門合格
 - ③文書部門と速度部門が両方合格ならば、2級合格
- (3) 入力データが終了したら、2級合格者数および合格率、部門のみの合格者数を表示する。
- (4) データにエラーはないものとする。

4 出力設計

[illegible]

5 変数

6 流れ図



菓子店のスイーツ売上が記録された CSV データを読み、実行結果のように表示する。

月 ××	品番 ×	売上数量 ×××
---------	---------	-------------

商品別売上集計表（1～6月）		
	作成者	×××
品名	数量	金額
チーズケーキ	146	29,200
抹茶ムース	164	34,440
ザッハトルテ	175	43,750
ミルフィーユ	118	27,140
計	603	134,530

	(0)	(1)	(2)	(3)	(4)
hinMei		チーズケーキ	抹茶ムース	ザッハトルテ	ミルフィーユ
hinTan	0	200	210	250	300
sukei					
kinkei					

-
- The image displays a 6x100 grid of 600 small squares. Each square contains a number, and the numbers are arranged in a repeating sequence of 100 numbers per row. The sequence of numbers is as follows:
- Row 1: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100.
 - Row 2: 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200.
 - Row 3: 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300.
 - Row 4: 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400.
 - Row 5: 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500.
 - Row 6: 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600.

[illegible]

実習課題 16 北海道のおみやげ通販売上表の作成（1次元配列と線形探索）

ある通信販売サイトの北海道土産の売上が記録された CSV データを読み、実行結果のように表示する。

1 入力データ (ファイル名 : miyage.csv)

月 ××	商品コード ×××	売上数量 ×××
---------	--------------	-------------

2 実行結果

北海道のお土産通販 売上集計				
品名	数量	金額	割合	貢献度
白い恋人	363	552,486	40.0	A
マルセイバターサンド	207	269,100	19.0	C
じゃがポックル	427	377,895	27.0	B
バターケーキ	247	165,490	12.0	C
計	1,244	1,364,971		

3 处理条件

- (1) 月・品番・売上数量が記録された CSV ファイルを読み込む
- (2) 配列 hinBan に商品コード、hinMei に商品名、hinTan には単価が記憶されている
- (3) 配列 skei は商品ごとに売上数量を集計し、keinkei は商品ごと売上金額を集計する

	(0)	(1)	(2)	(3)	(4)
hinBan	0	101	115	208	255
hinMei		チーズケーキ	抹茶ムース	ザッハトルテ	ミルフィーユ
Tan	0	200	210	250	300
skei	(0)	(1)	(2)	(3)	(4)
kkei					

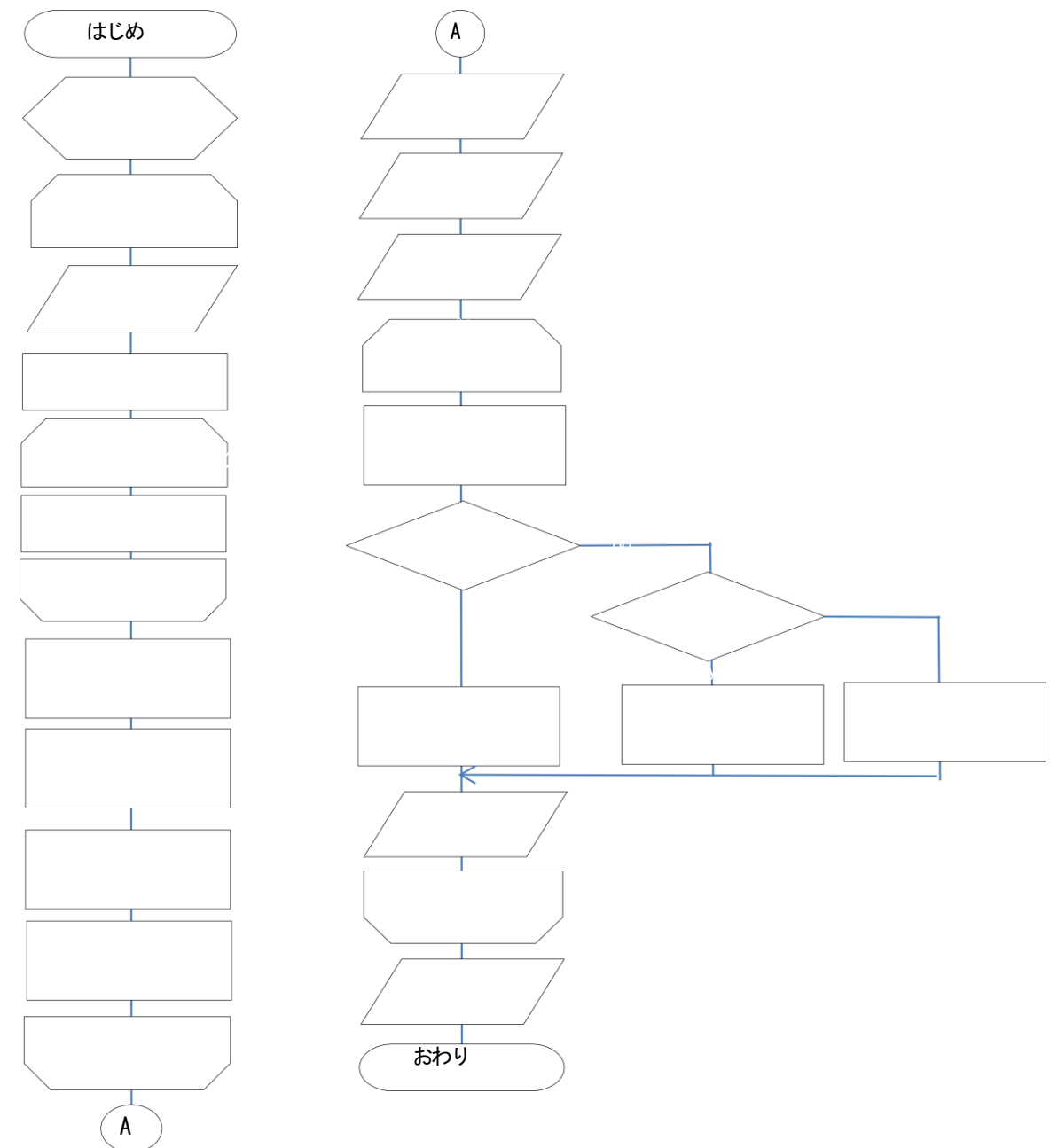
- (4) 割合は、次の式で求める **商品ごとの金額計**×100÷**金額の総合計**
 (5) 貢献度は、割合が40%以上ならA、20%以上ならB、それ以外はCとする

4 出力設計

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89</											

5 変数

6 流れ図



実習課題 17 観光ホテル予約状況 (1次元配列と線形探索)

観光ホテルの利用状況が記録された CSV データを読み、実行結果のように表示する。

1 入力データ (ファイル名 : hotel.csv)

施設コード	日帰り客数	宿泊客数
××	×××	×××

2 実行結果

施設名	日帰り客数計	宿泊客数計	宿泊割合 (%)	備考
朝日ホテル	19	33	63.0	◎
旅人の宿	6	7	53.0	◎
大雪ホテル	35	19	35.0	△
旭温泉	7	19	73.0	◎
ホテル憩館	16	23	58.0	◎
安らぎ旅館	18	28	60.0	◎
総計	101	129		

3 处理条件

(1) 配列 `cod` に施設コードを、配列 `Mei` に施設名を記憶する。なお、各配列は添字で対応している。

	(0)	(1)	(2)	(3)	(4)	(5)	(6)
code		101	201	301	401	501	601
mei		朝日ホテル	旅人の宿	大雪ホテル	旭温泉	ホテル憩館	安らぎ旅館

入力データを読み、次の処理を行う。

(2) 施設コードをもとに配列 `cod` を探索し、配列 `Hkei` に日帰り客数を、配列 `Skei` に宿泊客数を集計する。

(3) Skei(0)は曰帰り客数と宿泊客数の総計用に利用する。また、Hkei、Skeiはcodと添字で対応している。

	(0)	(1)	(2)	(3)	(4)	(5)	(6)
hkei							
skei							

(4) 入力データが終了したら、次の処理を行う。

①各施設の宿泊割合(%)を次の計算式で求める。

$$\text{利用客数計} = \text{日帰り客総計} + \text{宿泊客数計}$$
$$\text{宿泊割合(\%)} = \text{宿泊客総計} \times 100 \div \text{利用客総計}$$

②施設名から備考までを実行結果のように表示する。

なお、備考は、宿泊割合(%)が 60%以上の場合◎を、40%以上 60%未満の場合○を 40%未満の場合△を表示する。

(5) 最後に、総計を実行結果のように表示する。

(6) データにエラーはないものとする。

4 出力設計

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85															

5 変数

