

平成25年度前期 情報検定

<実施 平成25年9月8日（日）>

プログラミングスキル

（説明時間 10：00～10：10）

（試験時間 10：10～11：40）

- ・ 試験問題は試験開始の合図があるまで開かないでください。
- ・ 解答用紙（マークシート）への必要事項の記入は、試験開始の合図と同時に行いますので、それまで伏せておいてください。
- ・ 試験開始の合図の後、次のページを開いてください。＜受験上の注意＞が記載されています。必ず目を通してから解答を始めてください。
- ・ 試験問題は、すべてマークシート方式です。正解と思われるものを1つ選び、解答欄の○をHBの黒鉛筆でぬりつぶしてください。2つ以上ぬりつぶすと、不正解になります。
- ・ 辞書、参考書類の使用および筆記用具の貸し借りは一切禁止です。
- ・ 電卓の使用が認められます。ただし、下記の機種については使用が認められません。

<使用を認めない電卓>

1. 電池式（太陽電池を含む）以外の電卓
2. 文字表示領域が複数行ある電卓（計算状態表示の一行は含まない）
3. プログラムを組み込む機能がある電卓
4. 電卓が主たる機能ではないもの
 - * パソコン（電子メール専用機等を含む）、携帯電話（PHS）、スマートフォン、タブレット、電子手帳、電子メモ、電子辞書、翻訳機能付き電卓、音声応答のある電卓、電卓付腕時計等
5. その他試験監督者が不適切と認めるもの

<受験上の注意>

1. この試験問題は28ページあります。ページ数を確認してください。
乱丁等がある場合は、手をあげて試験監督者に合図してください。
※問題を読みやすくするために空白ページを設けている場合があります。
2. 解答用紙（マークシート）に、受験者氏名・受験番号を記入し、受験番号下欄の数字をぬりつぶしてください。正しく記入されていない場合は、採点されませんので十分注意してください。
3. 試験問題についての質問には、一切答えられません。自分で判断して解答してください。
4. 試験中の筆記用具の貸し借りは一切禁止します。筆記用具が破損等により使用不能となった場合は、手をあげて試験監督者に合図してください。
5. 試験を開始してから30分以内は途中退出できません。30分経過後退出する場合は、もう一度、受験番号・マーク・氏名が記載されているか確認して退出してください。なお、試験終了5分前の合図以降は退出できません。試験問題は各自お持ち帰りください。
6. 試験後にお知らせする合否結果（合否通知）、および合格者に交付する「合格証・認定証」はすべて、Web ページ（PC、モバイル）での認証によるデジタル「合否通知」、デジタル「合格証・認定証」に移行しました。
 - ①団体宛にはこれまでと同様に合否結果一覧ほか、試験結果資料一式を送付します。
 - ②合否等の結果についての電話・手紙等でのお問い合わせには、一切応じられませんので、ご了承ください。

<問題の構成>

必須問題 全員解答

問題 1 ～ 問題 4	2 ページ～14 ページ
-------------------------	--------------

選択問題 次の問題から 1 問選択し解答せよ。

(選択した問題は解答用紙「選択欄」に必ずマークすること)

※選択欄にマークがなく、解答のみマークした場合は採点を行いません。

・ C 言語の問題	16 ページ～20 ページ
・ 表計算の問題	21 ページ～25 ページ
・ アセンブラの問題	26 ページ～28 ページ

必須問題

問題 1 次のヒープソートに関する記述を読み、各設問に答えよ。

要素数 7 の 1 次元配列 A (図 1) をヒープソート法で整列する。

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	←配列の添字
A	5	2	7	1	8	9	4	

図 1 1 次元配列 (未整列の初期データ)

1 次元配列の並びは、図 2 に示すように、配列の各要素を節 (ノード) とする 2 分木に見立てることができる。

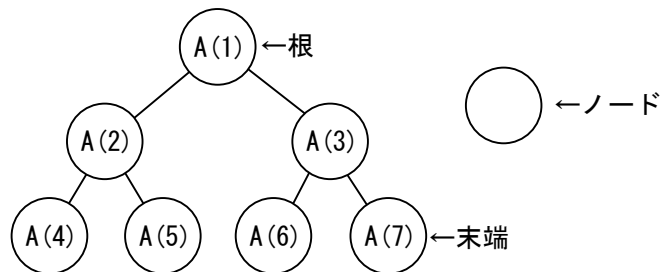


図 2 配列に対応する 2 分木

また、ヒープの条件を満たすとは、2 分木を構成するどのノードにおいても、親は子以上の値 (親 \geq 子) であることを示している

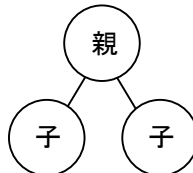


図 3 ヒープ

[ヒープソートの説明]

手順 1: 配列データから、ヒープを作成する。

手順 2: ヒープの根と末端のノードを交換し、末端のノードをヒープから切り離す。

つまり、ノード数を一つ減らし、残りを次の 2 分木とする。

手順 3: 残った 2 分木はヒープと呼べない場合が多いので、ヒープの条件を満たすように、親子関係を再構築する。

手順 4: 手順 2 と手順 3 を、ヒープからすべてのノードが切り離されるまで繰り返す。

以上の手順を繰り返すことにより、切り離されたデータが整列される。

<設問 1> 次の手順 1 に関する記述中の に入れるべき適切な字句を解答群から選べ。

ヒープはノードが一つもない状態を初期値とし、図 1 で示される配列の先頭から順に、ヒープの条件を満たすように、1 要素ずつ追加する。このとき 2 分木は、ヒープの条件を満たしている。

ヒープが作成される過程を 5 回目の追加まで図 4 に示す。

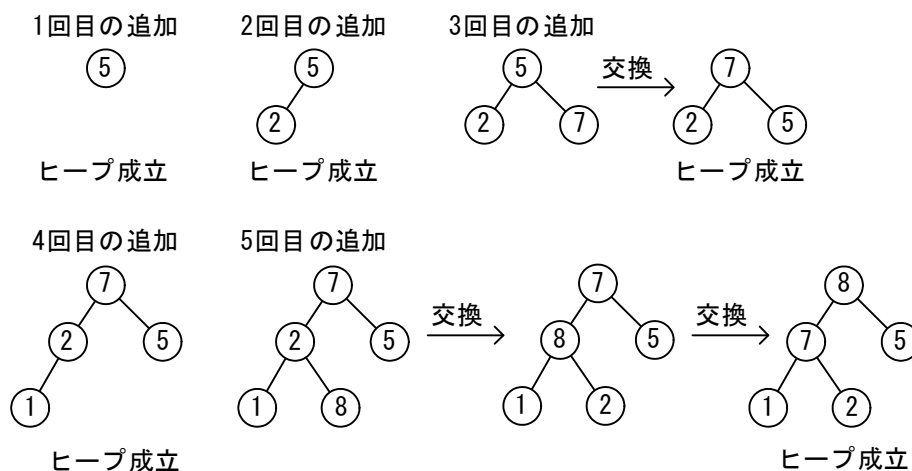
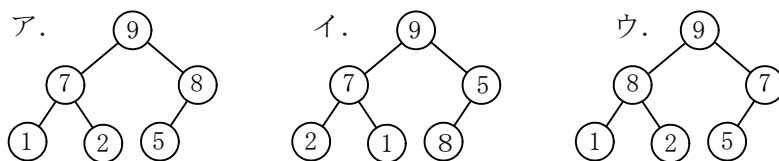


図 4 ヒープの作成過程

図 4 に従い、次の 6 回目のデータが追加され、ヒープの条件を満たしたときのヒープの構造は (1) である。

また、1 回目の追加から 7 回目の追加まで、すべてのデータが追加されヒープが完成するまでに、交換は (2) 回行われる。

(1) の解答群



(2) の解答群

- ア. 3 イ. 4 ウ. 5

<設問 2> 次の手順 2 に関する記述中の に入れるべき適切な字句を解答群から選べ。

手順 2 が最初に実行された後の、ヒープと配列の関係を図 5 に示す。図 5 の配列の内容は (3) である。

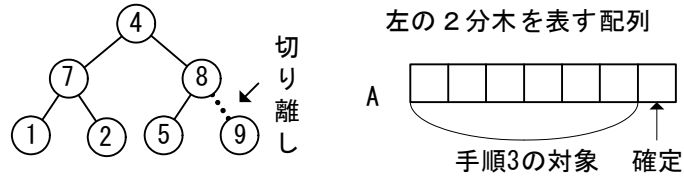


図 5 手順 2 実行後のヒープと配列の関係

また、 (3) をヒープの条件を満たすように再構築するために、根の子である 7 と 8 を比較し、大きい方の 8 と根の 4 を交換する。交換した 4 を新たな親とし、その子である (4) と比較し、子の方が大きいので、4 と (4) を交換する。その結果、 (5) の形になる。

(3) の解答群

ア. A

4	7	1	2	8	5	9
---	---	---	---	---	---	---

イ. A

4	7	8	1	2	5	9
---	---	---	---	---	---	---

ウ. A

4	8	5	7	1	2	9
---	---	---	---	---	---	---

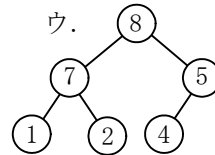
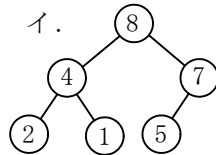
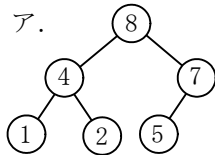
(4) の解答群

ア. 2

イ. 5

ウ. 8

(5) の解答群



問題を読みやすくするために、
このページは空白にしてあります。

問題2 次の流れ図の説明を読み、各設問に答えよ。

[流れ図の説明]

1次元配列の中に格納されたデータの中から二分探索法によりデータを探索する流れ図である。二分探索法とは、昇順または降順に整列済みである配列を利用した探索方法で、探索しようとする値と1次元配列の中央の値を比べ、その大小関係によって探索範囲を狭くして目的のデータを探索するものである。

次の流れ図は、N件の数値データからなる1次元配列Tから二分探索法によりデータを検索するものである。探索すべきデータの値としてXが与えられたとき、 $X=T[S]$ となるような添字Sを求める。ただし、見つからなかった場合には、Sに-1を返すものとする。また、処理中の除算の商は小数点以下切り捨てとする。

なお、1次元配列Tの内容は昇順に整列済みであり、配列の添字は0から始まる。

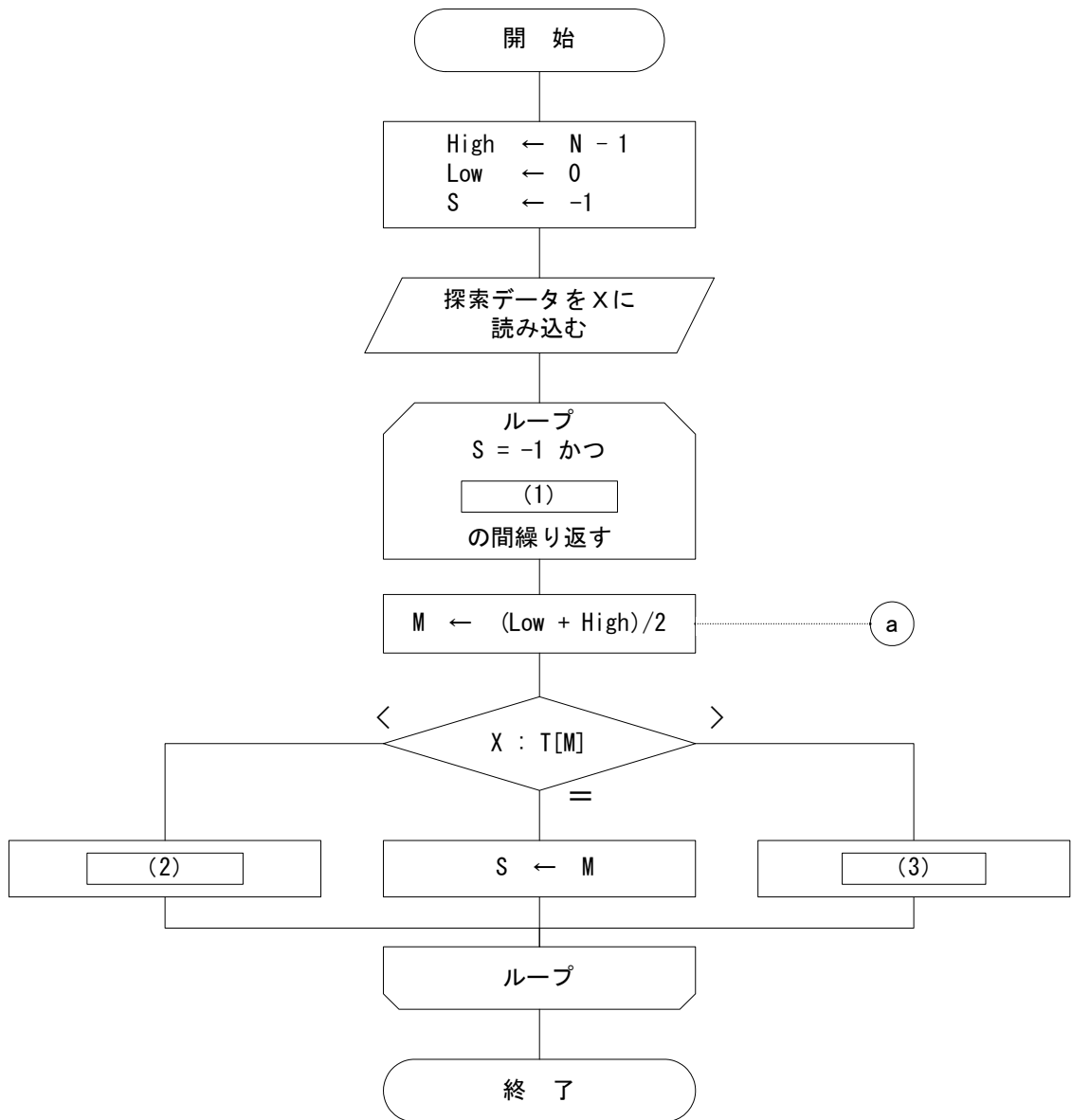


図1 流れ図

<設問 1> 流れ図中の に入れるべき適切な字句を解答群から選べ。

(1) の解答群

ア. $Low < High$

イ. $Low \leq High$

ウ. $Low > High$

エ. $Low \geq High$

(2) , (3) の解答群

ア. $High \leftarrow M - 1$

イ. $High \leftarrow M + 1$

ウ. $Low \leftarrow M - 1$

エ. $Low \leftarrow M + 1$

<設問 2> 流れ図をトレースした次の記述中の に入れるべき適切な字句を解答群から選べ

配列 T が図 2 のようであり、X に 13 を入力して実行した場合、流れ図中の (a) の実行回数は (4) 回であり、S の内容は (5) となる。

	0	1	2	3	4	5	6	7	8	9
配列 T	1	3	5	7	9	11	13	15	17	19

図 2 配列 T に格納されている値

(4) , (5) の解答群

ア. -1

イ. 3

ウ. 4

エ. 5

オ. 6

問題3 次の流れ図の説明を読み、各設問に答えよ。

[流れ図の説明]

在庫マスタファイルをトランザクションファイルの内容で更新し、新マスタファイルを作成する流れ図である。

なお、流れ図中で用いている最高値とは、システムで表現できる最大の値であり、どの入力データと比較しても大きいと判定される値である。

[在庫マスタファイルについて]

在庫マスタファイルのレコード形式は次の図1のようになっている。商品コードの昇順に整列済みであり、同じ商品コードを持つレコードは存在しない。

削除フラグは、取り扱い中の商品であれば0、取り扱わなくなった商品であれば1が設定される。

商品コード	商品名	在庫量	削除フラグ
MCODE	MNAME	ZAICO	FLG

図1 在庫マスタファイルのレコード形式

[トランザクションファイルについて]

トランザクションファイルのレコード形式は次の図2のようになっている。商品コード、日付、時間の昇順に整列済みであり、同じ商品コードを持つレコードは複数存在する可能性がある。

商品コード	日付	時間	処理区分	数量	商品名
TCODE	HIZUKE	JIKAN	KUBUN	SURYO	TNAME

図2 トランザクションファイルのレコード形式

図2の処理区分は次のような値と意味を持つ。

表 トランザクションファイルの処理区分の値と意味

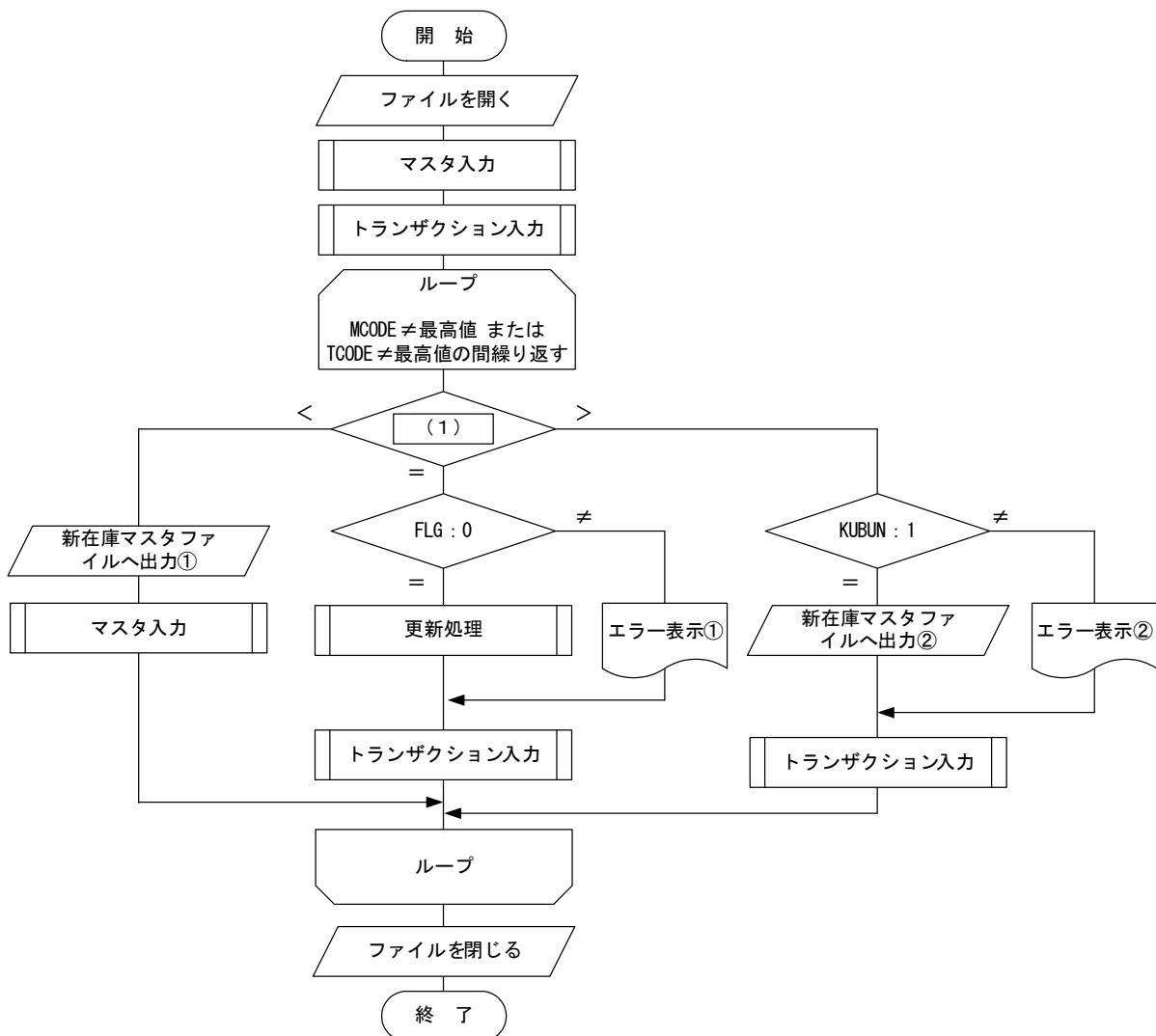
値	意味
1	新規レコードとして在庫マスタファイルに追加する
2	数量を在庫量に加える
3	在庫量から数量を減じる
4	削除フラグに1を設定する

処理区分が1の場合は全ての項目に値が設定されるが、処理区分が2と3の場合は商品コードと数量、処理区分が4の場合は商品コードに必ず値が設定される。

[更新処理について]

1. 在庫マスタファイルの商品コードがトランザクションファイル中に存在しないレコードは、そのまま新マスタファイルに出力する。
2. トランザクションファイルの商品コードが在庫マスタファイル中に存在しないレコードは、処理区分が1の場合はトランザクションファイルのレコードの内容を新マスタファイルに出力するが、処理区分が1以外の場合はエラーとする（更新に該当するレコードが存在しない）。
3. 在庫マスタファイルの商品コードとトランザクションファイルの商品コードが一致する場合、表にあるような処理を行うが、在庫量より多い数量を減じようとする場合と削除済みのレコードに対する更新はエラーとする。
4. 何らかのエラーがあるトランザクションファイルのレコードの処理は行わない。

[流れ図]



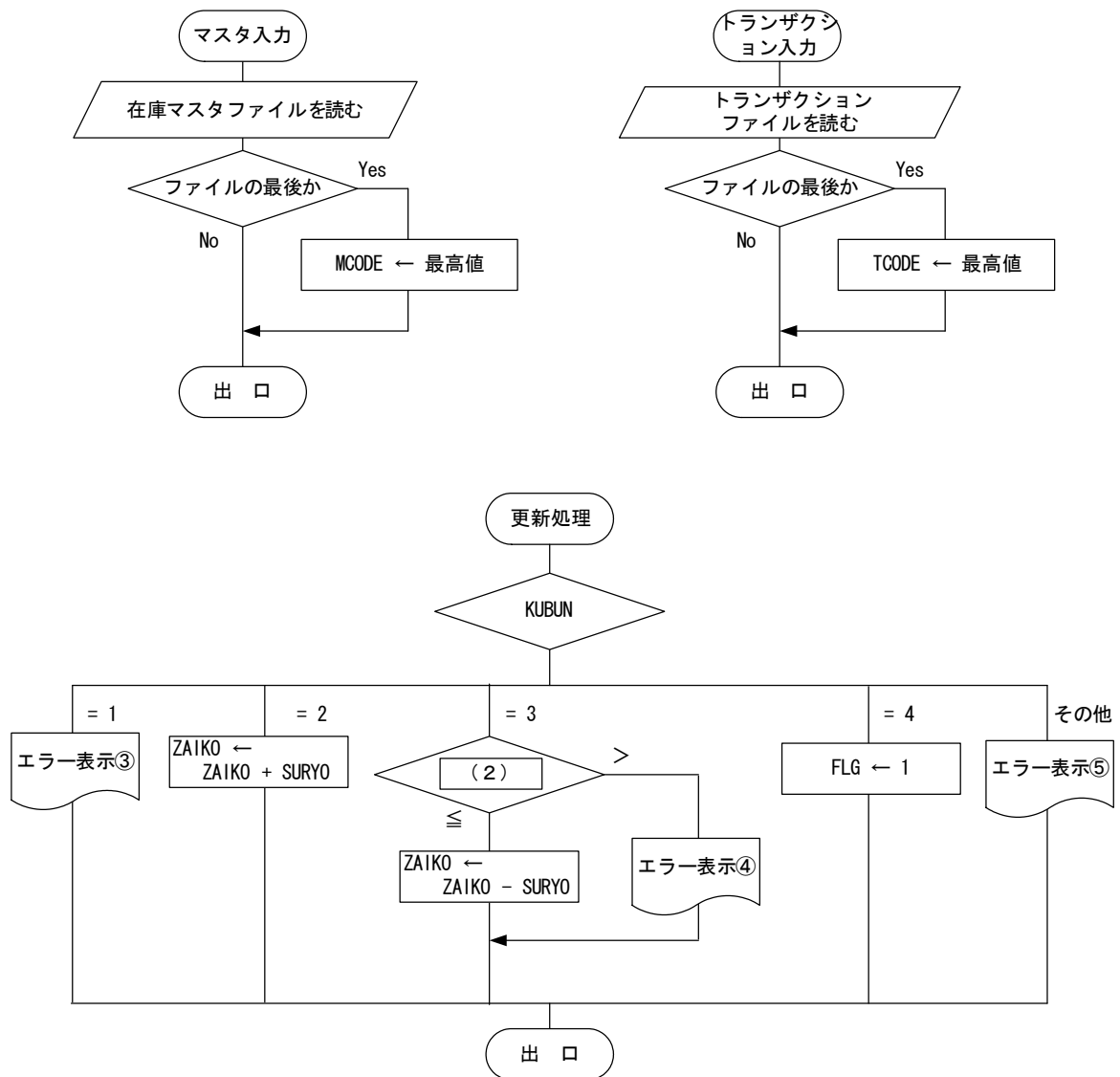


図3 在庫マスタファイルを更新する流れ図

<設問1> 流れ図中の に入るべき適切な字句を解答群から選べ。

(1), (2) の解答群

ア. FLG : KUBUN

イ. KUBUN : FLG

ウ. MCODE : TCODE

エ. TCODE : MCODE

オ. SURYO : ZAIKO

カ. ZAIKO : SURYO

<設問 2> 流れ図中の“新在庫マスタファイルへ出力①”と“新在庫マスタファイルへ出力②”で出力すべき内容を解答群から選べ。

新在庫マスタファイルへ出力①で出力すべき内容：

新在庫マスタファイルへ出力②で出力すべき内容：

(3) , (4) の解答群

ア. TCODE, TNAME, SURYO, 0

イ. TCODE, MNAME, ZAIKO, FLG

ウ. TCODE, TNAME, SURYO, KUBUN

エ. MCODE, MNAME, ZAIKO, FLG

<設問 3> 流れ図中の“エラー表示①”から“エラー表示⑤”の部分で表示すべき内容として適切な字句を解答群から選べ。

エラー表示①の内容：

エラー表示②の内容：

エラー表示③の内容：

エラー表示④の内容：

エラー表示⑤の内容：

(5) ~ (9) の解答群

ア. この商品コードはトランザクションファイル中に存在しません

イ. この商品コードは在庫マスタファイル中に存在しません

ウ. 在庫量が不足しています

エ. 削除済の商品を更新しようとした

オ. 削除できません

カ. 処理区分が1～4ではありません

キ. 新規で追加しようとした商品コードは既に使われています

問題 4 次のプログラムの説明を読み、プログラム中の に入れるべき適切な字句を解答群から選べ。

[プログラムの説明]

1次元配列 `moji` の中に長さ `len_m` の文字列が格納されている。この文字列の先頭から配列 `pre` (長さ `len_p`) に格納されている文字列を探索し、配列 `fol` (長さ `len_f`) に格納されている文字列で置換するプログラム `replace` である。

各配列の添字は 0 から始まり、処理に十分な大きさを持っているものとする。

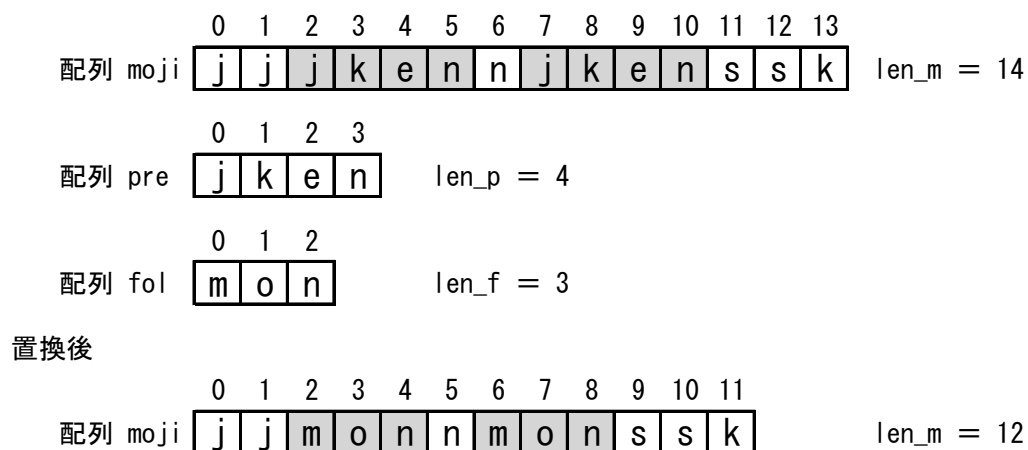


図 1 実行例

[疑似言語の記述形式の説明]

記述形式	説明
○	手続き、変数などの名前、型などを宣言する
・変数 ← 式	変数に式の値を代入する
{文}	注釈を記述する
↑ 条件式 ・処理 1 ↓ ・処理 2	選択処理を示す。 条件式が真の時は処理 1 を実行し、 偽の時は処理 2 を実行する。
■ 条件式 ・処理 ■	前判定繰り返し処理を示す。 条件式が真の間、処理を実行する。

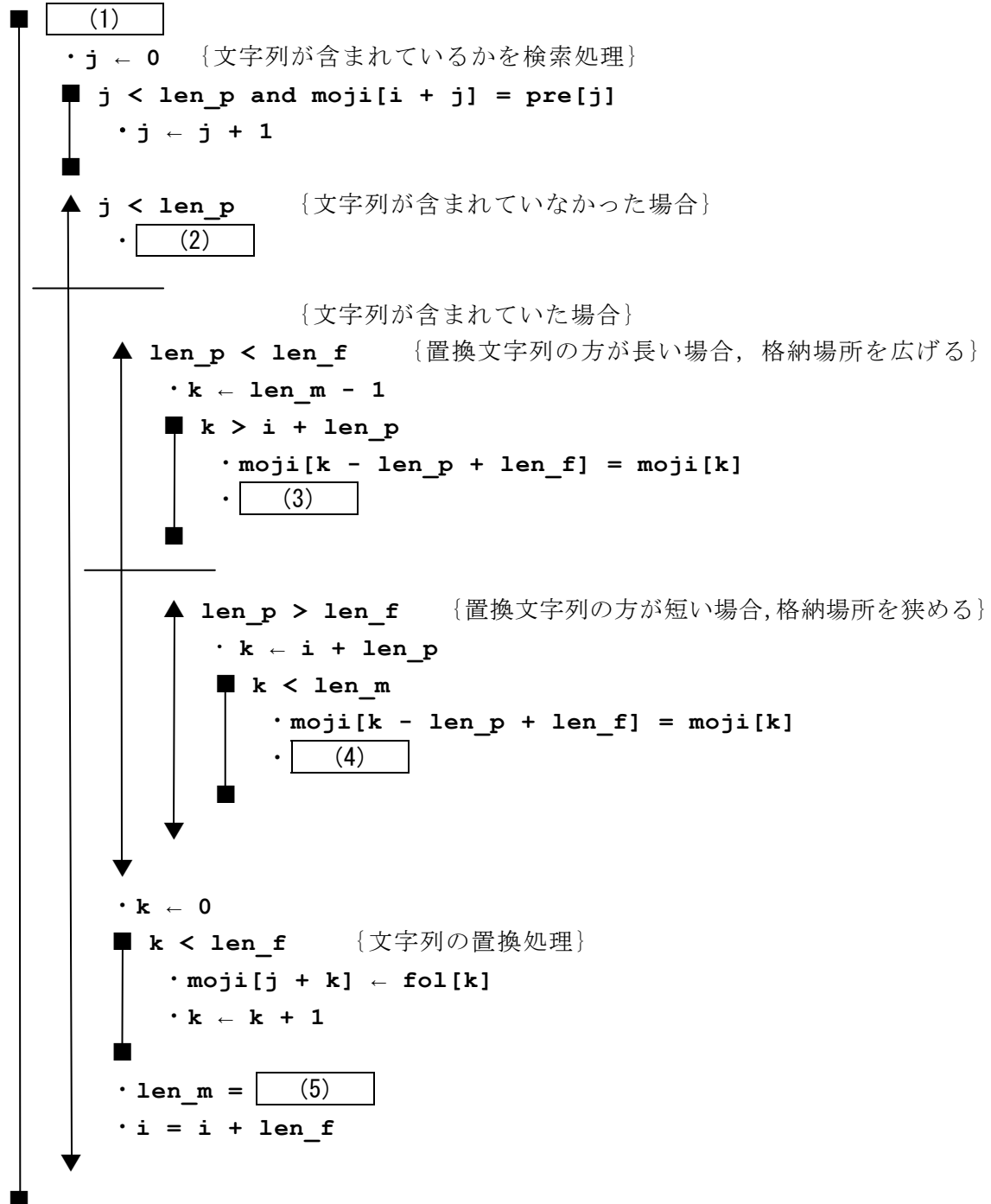
[プログラム]

○プログラム名 : `replace`

○文字型配列 : `moji, pre, fol`

○整数型 : `len_m, len_p, len_f, i, j, k`

・ `i ← 0`



(1) の解答群

ア. $i < \text{len}_f$

イ. $i < \text{len}_p$

ウ. $i < \text{len}_m - \text{len}_p + 1$

(2) ~ (4) の解答群

ア. $i \leftarrow i + 1$

イ. $i \leftarrow i - 1$

ウ. $k \leftarrow k + 1$

エ. $k \leftarrow k - 1$

(5) の解答群

ア. $\text{len}_m + \text{len}_p + 1$

イ. $\text{len}_m + \text{len}_f + 1$

ウ. $\text{len}_m + \text{len}_p - \text{len}_f$

エ. $\text{len}_m - \text{len}_p + \text{len}_f$

< 選 択 問 題 >

選択問題は問題から1つ選択し解答せよ。

選択した問題は必ず、解答用紙「選択欄」にマークすること。

※選択欄にマークがなく、解答のみの場合は採点を行いません。

各構成は以下のとおり。

選択問題

- | | |
|------------|---------------|
| ・ C言語の問題 | 16 ページ～20 ページ |
| ・ 表計算の問題 | 21 ページ～25 ページ |
| ・ アセンブラの問題 | 26 ページ～28 ページ |

次のC言語プログラムの説明を読み、各設問に答えよ。

[プログラムの説明]

m進数の数値をn進数に基数変換するconv関数である。

conv 関数

引 数：m (変換前の基数), m 進数の文字配列, n (変換後の基数), 結果の文字配列

機 能：m 進数で表現された文字配列を n 進数に変換し, 結果の文字配列に格納する。ただし, m 進数の文字配列に使用できない文字が入っている場合はエラーメッセージを結果の文字配列に格納する。

戻り値：エラーが無ければ 0, エラーがあれば-1

- ① mおよびnは, 必ず2~16の整数値が渡される。
- ② 基数変換で使用する文字は, '0' ~ '9' および 'A' ~ 'F' である。ただし, conv 関数に渡されるm進数の文字配列に, これらの文字以外が存在すれば, エラーメッセージを結果の文字配列に格納する。
- ③ conv関数は, m進数で使われる1文字を整数値に変換するto_bin関数を呼び出す。

to_bin 関数

引 数：文字

機 能：'0' ~ '9' および 'A' ~ 'F' の文字を 0~15 に変換する。なお, これらの文字以外であれば-1 を返す。

表 文字と整数値の対応表

文字	'0'	'1'	'2'	'3'	'4'	'5'	'6'	'7'
整数値	0	1	2	3	4	5	6	7
文字	'8'	'9'	'A'	'B'	'C'	'D'	'E'	'F'
整数値	8	9	10	11	12	13	14	15

戻り値：0~15 の整数値または-1


```

/* m進数文字を整数値へ変換 */
static int to_bin(char ch) {
    int i;
    for(i=0; i<16; i++) {
        if ( ch == NUM[i] ) (6);
    }
    return ERROR;
}

```

(1) の解答群

ア. <code>from[i] == '¥0'</code>	イ. <code>from[i] != '¥0'</code>
ウ. <code>from[m] == '¥0'</code>	エ. <code>from[m] != '¥0'</code>

(2) の解答群

ア. <code>dec = k</code>	イ. <code>dec = dec * k + 10</code>
ウ. <code>dec = dec * m + k</code>	エ. <code>dec = dec * n + k</code>

(3) の解答群

ア. <code>dec > 0</code>	イ. <code>dec < 0</code>
ウ. <code>dec == 0</code>	エ. <code>dec > n</code>

(4) の解答群

ア. <code>to[k] = to[k-1]</code>	イ. <code>to[k] = to[m]</code>
ウ. <code>to[k] = to[n-1]</code>	エ. <code>to[k] = to[n]</code>

(5) の解答群

ア. <code>to[0] = NUM[k]</code>	イ. <code>to[k] = NUM[k]</code>
ウ. <code>to[m] = NUM[k]</code>	エ. <code>to[n] = NUM[k]</code>

(6) の解答群

ア. <code>return i - 1</code>	イ. <code>return i</code>
ウ. <code>return i + 1</code>	エ. <code>return SUCCESS</code>

<設問 2 > `conv` 関数に次の図 1 のようなデータを渡された場合、`conv` 関数が返す値と `to` を出力した場合に表示される文字列の組み合わせを解答群から選べ。

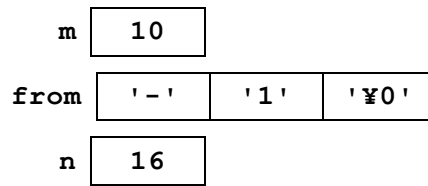


図 1 `conv`関数に渡された値

(7) の解答群

	<code>conv</code> が返す値	<code>to</code> を出力した文字列
ア.	-1	FFFF
イ.	0	FFFF
ウ.	-1	使えない文字があります
エ.	0	使えない文字があります

<設問 3 > 次のプログラムの修正に関する記述中の に入れるべき適切な字句を解答群から選べ。

このプログラムは、文字配列 `from` の中に理論上あり得ない値が入力された場合でも処理してしまう。例えば、次の図のように 8 進数でありながら、'8' 以上の文字が文字配列 `from` に含まれているような場合である。

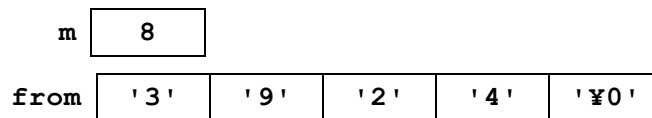


図 2 誤った値のパターン

そこで、このように誤った値が入力された場合もエラーとするため、`to_bin` 関数を次のように変更する（下線部分が変更箇所）。

[変更後の `to_bin` 関数]

```
static int to_bin(char ch, int x) {
    int i;
    for(i=0; i<x; i++) {
        if ( ch == NUM[i] )  (6);
    }
    return ERROR;
}
```

また、`to_bin`関数を呼び出すプログラム中の⑧を次のように変更する。

変更前： `k = to_bin(from[i]);`

変更後： `k = to_bin(from[i],);`

(8) の解答群

ア. `dec`

イ. `i`

ウ. `m`

エ. `n`

次の表計算ソフトの記述を読み、各設問に答えよ。

この問題で使用する表計算ソフトの仕様は下記のとおりである。

COUNT 関数

指定した範囲内のセルから数値のセルの数を返す。

書式：COUNT(範囲)

COUNTIF 関数

範囲内のセルの中で条件に一致するセルの数を返す。条件は">10"のように等号や不等号と定数を組み合わせたものが使えるが、この条件を数式で作成することもできる。

書式：COUNTIF(範囲, 条件)

IF 関数

条件が真の時に「真の場合」、偽の時に「偽の場合」の計算結果や値を返す。

IF 関数の中にIF 関数を入れることができる。

書式：IF(条件式, 真の場合, 偽の場合)

SUM 関数

指定した範囲のセルに含まれる数値のセルの合計を求める。

書式：SUM(範囲)

式

=に続けて計算式や関数などを入力する。

&

文字列を結合する演算子。

セル番地の参照

セル番地に\$を付けることで、絶対番地（絶対参照）を表す。

他のワークシートの参照

「ワークシート名!セル番地」とすることで他のワークシートのセルを参照することができる。

J 専門学校の A 先生は、期末試験の結果を集計するために表計算ソフトを使用した。

A 先生の期末試験は、問題 1～問題 5 までの 5 問あり、各問題は 5 つの設問で構成される。学生は問題 1～問題 5 の中から 4 問選択して解答する。なお、5 問解答した場合は最初の 4 問（問題 1～問題 4）を集計の対象とする。

問題の解答はマークシートに記入させた。マークシートは OMR で読み取り、そのデータを図 1 のように「答案」ワークシートに取り込んだ。

A 列に学籍番号、B 列～Z 列は学生がマークした解答が 1～4 の数値で格納されている。B 列は問題 1 の設問 1、C 列は問題 1 の設問 2、…、Z 列は問題 5 の設問 5 の解答が格納されている。

なお、全部で 50 名の学生が受験した。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	132001	2	4	1	4	4	2	1	3	4	1	2	4	3	1	3	3	3	1	2	1					
2	132002	2	4	1	4	2	3	1	3	2	3	4	1	3	1	2						4	4	2	2	2
3	132003	2	4	1	4	2	2	1	3	2	3						3	3	1	2	3	4	4	2	2	2
4	132004	2	4	4	4	3						4	2	3	2	3	1	3	1	2	1	4	1	2	2	2
5	132005						2	3	3	1	1	4	2	1	1	4	3	3	1	2	3	4	4	2	4	2
6	132006	2	4	1	4	2						4	4	3	4	1	3	1	1	2	2	2	1	2	3	2
46	132046	2	4	1	4	2	2	1	3	4	2															
47	132047	2	4	2	4	2	2	1	3	2	3						3	3	3	2	3	4	4	2	2	2
48	132048	2	4	3	4	4	1	4	3	1	4						3	3	1	2	1	2	4	2	2	2
49	132049	1	4	3	4	2	2	1	3	4	4						3	3	1	2	1	4	4	2	2	2
50	132050	2	4	1	4	2	2	4	3	2	1	4	4	1	2	3	3	3	1	2	1					

⏟
⏟
⏟
⏟
⏟

問題1の解答
問題2の解答
問題3の解答
問題4の解答
問題5の解答

図 1 「答案」ワークシート

<設問 1 > 次の「採点」ワークシートの作成に関する記述中の に入れるべき適切な字句を解答群から選べ。

「答案」ワークシートを採点するため、図 2 のような「採点」ワークシートを作成した。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	正解	2	4	1	4	2	2	1	3	4	1	4	4	3	4	1	3	3	1	2	1	4	4	2	2	2
2	配点	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
3	学籍番号	問題1					問題2					問題3					問題4					問題5				
4	132001	5	5	5	5	0	5	5	5	5	5	0	5	5	0	0	5	5	5	5	5					
5	132002	5	5	5	5	5	0	5	5	0	0	5	0	5	0	0						5	5	5	5	5
6	132003	5	5	5	5	5	5	5	5	0	0						5	5	5	5	0	5	5	5	5	5
7	132004	5	5	0	5	0						5	0	5	0	0	0	5	5	5	5	5	5	0	5	5
8	132005						5	0	5	0	5	5	0	0	0	0	5	5	5	5	0	5	5	5	0	5
9	132006	5	5	5	5	5						5	5	5	5	5	5	0	5	5	0	0	0	5	0	5
10	132007	5	5	5	5	5											5	5	5	5	5	5	5	5	5	5
50	132047	5	5	0	5	5	5	5	5	0	0						5	5	0	5	0	5	5	5	5	5
51	132048	5	5	0	5	0	0	0	5	0	0						5	5	5	5	5	0	5	5	5	5
52	132049	0	5	0	5	5	5	5	5	5	0						5	5	5	5	5	5	5	5	5	5
53	132050	5	5	5	5	5	5	0	5	0	5	5	5	0	0	0	5	5	5	5	5					

図 2 「採点」ワークシート

1 行目は正解を入力し、2 行目は各設問の配点を入力した。

セル A4~A53 は「答案」ワークシートの学籍番号を表示するため次の式をセル A4 に入力し、セル A5~A53 に複写した。

= 答案!A1

セル B4~Z53 は、正解の場合に該当する設問の配点を表示し、不正解の場合は 0 を表示する。

セル B4 に次の式を入力し、セル C4~Z4 およびセル B5~Z53 に複写した。なお、解答していない設問は空欄とする。

= IF(答案!B1="", "", IF(答案!B1=(1), (2), 0))

(1), (2) の解答群

ア. B1

イ. B\$1

ウ. \$B1

エ. B2

オ. B\$2

カ. \$B2

<設問 2 > 次の「集計」ワークシートの作成に関する記述中の に入れるべき適切な字句を解答群から選べ。

「採点」ワークシートの採点結果を集計するため、図 3 のような「集計」ワークシートを作成した。

	A	B	C	D	E	F	G
1		問題1	問題2	問題3	問題4	問題5	得点
2	132001	20	25	10	25		80
3	132002	25	10	10		25	70
4	132003	25	15		20	25	85
5	132004	15		10	20	20	65
6	132005		15	5	20	20	60
7	132006	25		25	15	10	75
8	132046	25	20		20	25	90
...
49	132048	15	5		25	20	65
50	132049	15	20		25	25	85
51	132050	25	15	10	25		75

図 3 「集計」ワークシート

セル B2 は、「採点」ワークシートのセル B4~F4, セル C2 は「採点」ワークシートのセル G4~K4, …, セル F2 は「採点」ワークシートのセル V4~Z4 を集計する。

集計に SUM 関数のみを使うと集計するデータが 1 件も無い場合は 0 になる。これでは、選択した問題が全問不正解だった場合の 0 と区別ができない。そこで、1 つの問題も解答していない場合は空欄にする。

まず、セル B2 には次の式を入力し、セル B3~B51 まで複写した。

$$= \text{IF}(\text{ (3)}, \text{SUM}(\text{採点!B4:F4}), "")$$

次に、セル C2~F2 にセル B2 に入力した式の範囲を変更したものを入力し、3~51 行のセルに複写した。

G 列の得点は、問題ごとに集計した値を合計する。4 問選択ではあるが、全て解答した学生もいた。全て解答した場合は、最初の 4 問（問題 1~問題 4）を合計する。

セル G2 には次の式を入力し、セル G3~G51 まで複写した。

$$= \text{IF}(\text{COUNT}(\text{B2:F2}) < 5, \text{ (4)}, \text{ (5)})$$

(3) の解答群

- ア. COUNT(採点!B4:F4)>0
 イ. COUNT(採点!\$B\$4:\$F\$4)>0
 ウ. COUNT(採点!B4:F4)=0
 エ. COUNT(採点!\$B\$4:\$F\$4)=0

(4), (5) の解答群

- ア. 0
 イ. ""
 ウ. SUM(B2:E2)
 エ. SUM(B2:F2)
 オ. SUM(C2:E2)
 カ. SUM(C2:F2)

<設問3> 次の「ヒストグラム」ワークシートの作成に関する記述中の に
入れるべき適切な字句を解答群から選べ。

「集計」ワークシートの得点から図4のようなヒストグラムを作成するため、図5
のような「ヒストグラム」ワークシートを作成した。

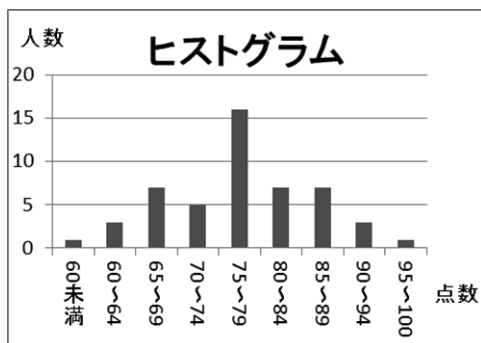


図4 ヒストグラム

	A	B	C	D
1	得点の範囲	下限	上限	人数
2	60未満	0	59	1
3	60~64	60	64	3
4	65~69	65	69	7
5	70~74	70	74	5
6	75~79	75	79	16
7	80~84	80	84	7
8	85~89	85	89	7
9	90~94	90	94	3
10	95~100	95	100	1

図5 「ヒストグラム」ワークシート

60点未満の人数を集計するため、セルD2に次の式を入力した。

= COUNTIF(集計!G\$2:G\$51, " (6) ")

次に、各点数範囲の人数を集計するため、セルD3に次の式を入力し、セルD4~D10
に複写した。

= COUNTIF(集計!G\$2:G\$51, "<="& (7)) - SUM((8))

(6) の解答群

ア. >59 イ. <59 ウ. >60 エ. <60

(7) の解答群

ア. B3 イ. B\$3 ウ. C3 エ. C\$3

(8) の解答群

ア. D2:D2 イ. D\$2:D2 ウ. \$D2:D2 エ. \$D2:\$D2

選択問題 アセンブラの問題

次のアセンブラ言語CASL II プログラムの説明を読み、各設問に答えよ。

[プログラムの説明]

要素数 N の1次元配列 DAT を、バブルソート法により昇順に整列するプログラムである。

配列 DAT は、 $DAT+0$ 番地以降に連続して格納済みであり、 N は N 番地に格納済みである。ただし、 N は 1 以上 100 以下とする。

[バブルソート法の説明]

手順 1 : J を $N-1$ から 1 ずつ減らしながら、手順 2 を $J = 2$ となるまで繰り返す。

手順 2 : 配列の先頭から、隣接する要素を順次比較し、最大値を $DAT+J$ 番地に求める。図に $N = 5$, $J = 4$ の例を示す。

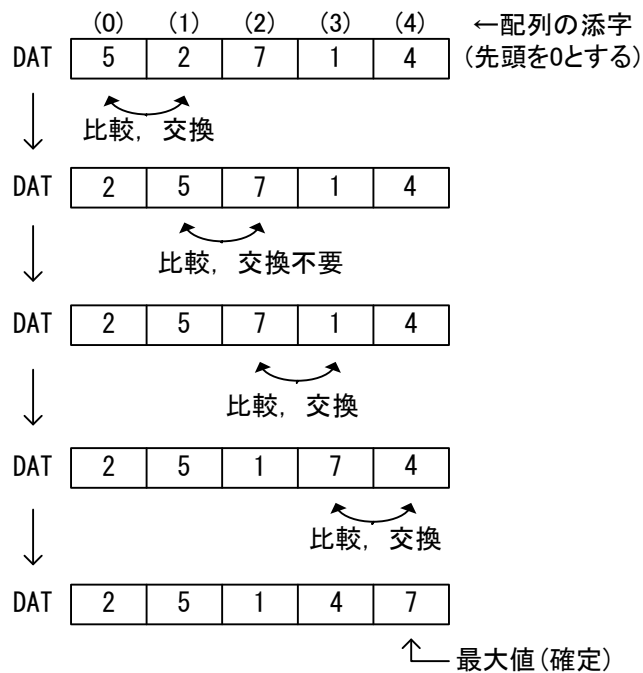


図 $N = 5$, $J = 4$ の例

[プログラム]

行番号	ラベル	命令	オペランド	コメント
100	BUBLLE	START		
110		LD	GR3,N	
120	LOOP1	SUBA	GR3,=1	
130		JZE	FINISH	
140		LD	GR1,=0	
150	LOOP2		(1)	
160		LD	GR0,DAT,GR1	
170		CPA	GR0,DAT,GR2	
180		JZE	NEXT	
190			(2)	
200		LD	GR4,DAT,GR2	
210		ST	GR4,DAT,GR1	
220			(3)	
230	NEXT	LAD	GR1,1,GR1	
240		CPA	GR1,GR3	
250			(4)	
260		JUMP	LOOP1	
270	FINISH	RET		
280	N	DS	1	
290	DAT	DS	100	
300		END		

<設問 1> プログラム中の に入れるべき適切な字句を解答群から選べ。

(1) の解答群

- | | |
|------------------|------------------|
| ア. LAD GR2,1,GR1 | イ. LAD GR2,1,GR3 |
| ウ. LD GR2,=0 | エ. LD GR2,N |

(2) , (4) の解答群

- | | |
|--------------|---------------|
| ア. JMI LOOP1 | イ. JMI LOOP2 |
| ウ. JMI NEXT | エ. JMI FINISH |

(3) の解答群

- | | |
|-------------------|-------------------|
| ア. ST GR0,DAT,GR1 | イ. ST GR0,DAT,GR2 |
| ウ. ST GR4,DAT,GR1 | エ. ST GR4,DAT,GR2 |

<設問 2> プログラムの拡張に関する記述を読み，表中の に入れるべき適切な字句を解答群から選べ。

手順 1 において，隣接する要素を比較していることにより，交換が一度も行われなかった場合，配列の全要素はすでに整列済みと判定できる。

そこで，手順 2 の途中であっても，交換が一度も行われなかったときに終了するため，次の命令を追加する。交換の有無は，GR5 をフラグとして使用し判定する。

[追加される命令]

行番号	ラベル	命令	オペランド	コメント
<input type="text"/>	(5)	LAD	GR5,=0	

行番号	ラベル	命令	オペランド	コメント
<input type="text"/>	(6)	LAD	GR5,=1	

行番号	ラベル	命令	オペランド	コメント
255			<input type="text"/>	(7)
256		JZE	FINISH	

(5) , (6) の解答群

ア. 115 イ. 135 ウ. 155 エ. 225

(7) の解答群

ア. CPA GR3,=0 イ. CPA GR3,=1
 ウ. CPA GR5,=0 エ. CPA GR5,=1

<メモ欄>

