

平成21年度前期 情報検定

<実施 平成21年9月6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）、ポケットベル、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35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. 合否通知の発送は平成21年10月中旬の予定です。
 - ①団体受験された方は、団体経由で合否の通知をいたします。
 - ②個人受験の方は、受験票に記載されている住所に郵送で合否の通知をいたします。
 - ③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 3	2 ページ～11 ページ
-------------------------	--------------

選択問題 A 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	14 ページ～15 ページ
・ 表計算の問題	16 ページ～17 ページ
・ アセンブラの問題	18 ページ～19 ページ

選択問題 B 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	20 ページ～26 ページ
・ 表計算の問題	27 ページ～30 ページ
・ アセンブラの問題	31 ページ～35 ページ

必須問題

問題 1 次の二分探索木に関する説明を読み、設問に答えよ。

二分木とは、各ノードが持つ子のノードの数が最大 2 である木構造のことを表す。
あるデータを二分木に格納するときの条件は、ノード X に対して左部分木に格納される値はノード X より小さく、右部分木に格納される値はノード X より大きいものとする。これを二分探索木という。(図 1 参照) なお、同じ値は格納されないものとする。

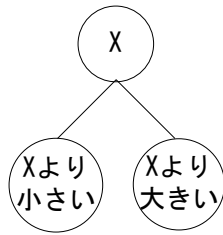
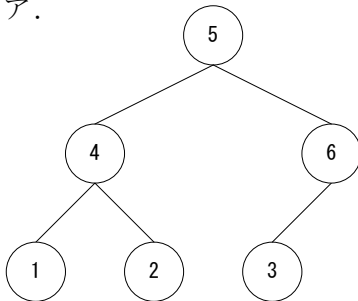


図 1 二分探索木例

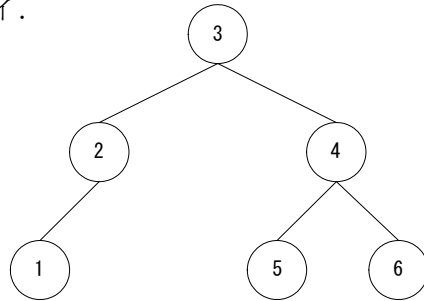
<設問 1> データを二分木に格納する。二分探索木の条件に合致しているのはどれか。
解答群より選べ。

(1) の解答群

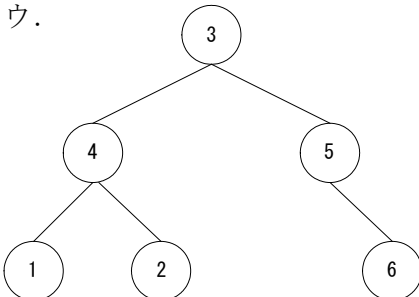
ア.



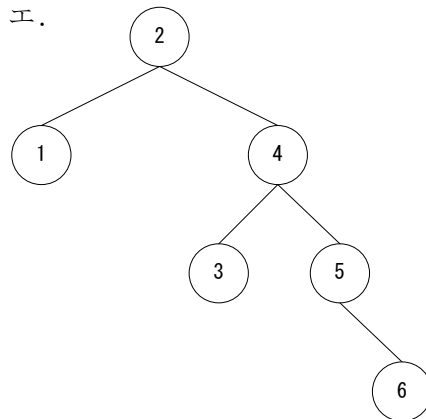
イ.



ウ.



エ.



<設問 2> 次の二分探索木に対してデータの挿入や削除を行う記述中の に入れるべき適切な字句または図を解答群から選べ。

二分探索木に対して新たな値を挿入する場合、まずルート(根)と挿入する値を比較し、ルートの値より小さければ左子ノード、ルート値以上であれば右子ノードが次の比較対象になる。これを比較対象のノードが無い状態になるまで繰り返し、その位置に新しいノードを挿入する。図 2 の二分探索木に対して値 12 を挿入すると (2) のようになる。

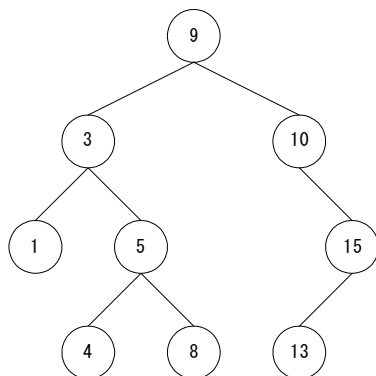
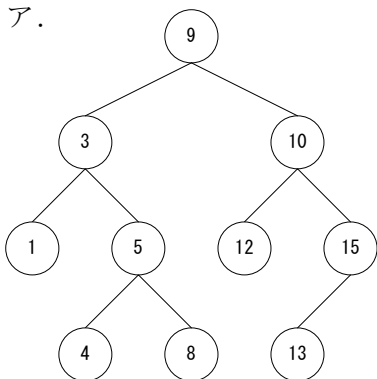


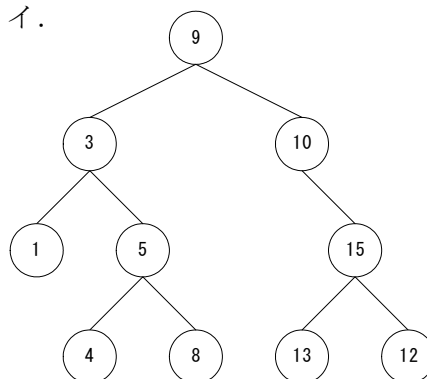
図 2 挿入前二分探索木

(2) の解答群

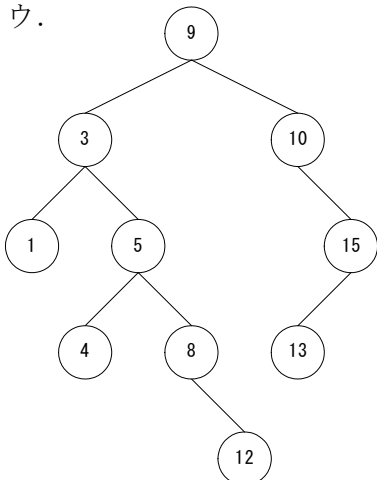
ア.



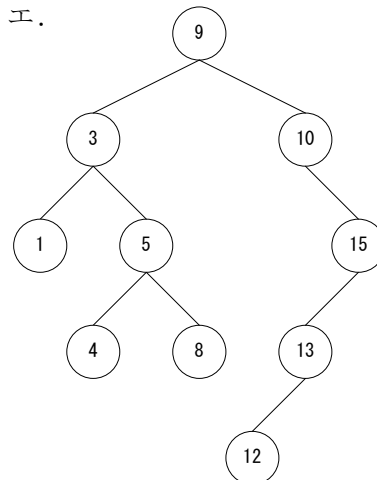
イ.



ウ.



エ.



また、ノードの削除をする場合も同様にルートから比較をしていき、削除するノードを探索する。ただし、削除するノードに子ノードがあるかどうかで、処理が変わる。

表 1 削除するノードと子ノードの関係と処理内容

子ノードの状態	処理内容
子ノードがない	そのまま削除する。
子ノードが 1 つ	子ノードで置き換えて対象のノードを削除する。
子ノードが 2 つ	削除するノードの左の子ノード群で (3) を持つノードか、または、右の子ノード群で (4) を持つノードを探索し、削除対象のノードと置き換える。

なお、置き換えの対象となる子ノードに、さらに子ノードがある場合は、表 1 の考え方に従って、置き換えの対象となる子ノードを削除すると考える。

図 3 の二分探索木に対して値 7 のノードを削除すると、図 4 のようになる。

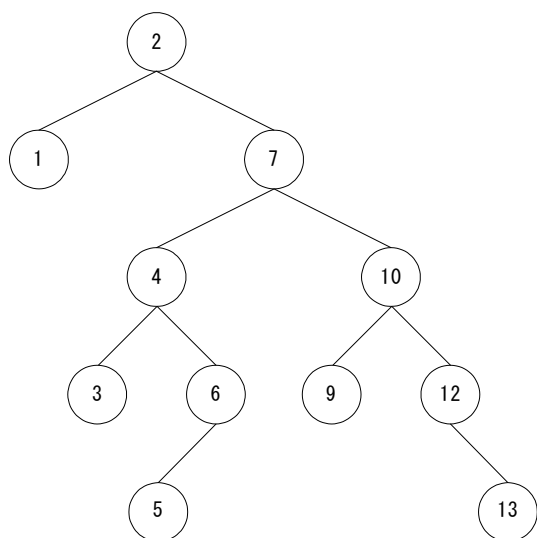


図 3 削除前二分探索木

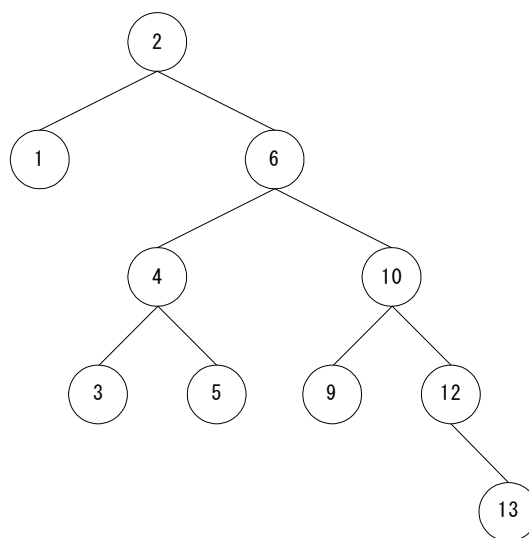


図 4 削除後二分探索木

(3) , (4) の解答群

- ア. 最大の値
- イ. 最小の値
- ウ. 削除するノードより小さい値
- エ. 削除するノードより大きい値

問題2 次の流れ図の説明を読み、設問に答えよ。

[流れ図の説明]

マスタファイルとトランザクションファイルを突き合わせ、マスタファイルの内容を更新し、新マスタファイルを生成する処理である。

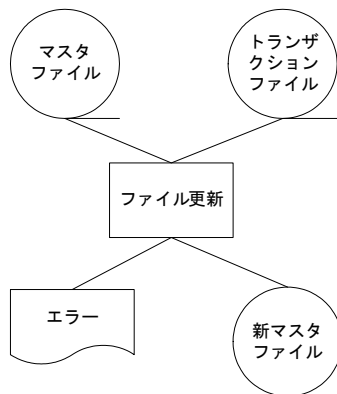


図1 プロセスチャート

マスタファイル、トランザクションファイルともにレコードキーの昇順に整列済みであり、以下の図2の形式になっている。

マスタファイルのレコード形式

レコードキー	数量
M-KEY	M 数量

トランザクションファイルのレコード形式

レコードキー	数量
T-KEY	T 数量

新マスタファイルのレコード形式

レコードキー	数量
新 M-KEY	新 M 数量

図2 レコードの形式

更新の内容は、T 数量と M 数量を加え新 M 数量とする。ただし、トランザクションレコードのキーの値がマスタファイル中に存在しない場合は、エラー印字する。

なお、マスタレコードのキーの値1件に対して、同じキーを持つトランザクションレコードが複数件存在する場合がある。

また、流れ図中の「最大値」とは、コンピュータ上で扱う値で一番大きい値のことであり、レコードキーには含まれていない。

<設問 1 > 次の流れ図中の に入れるべき適切な字句を解答群から選べ。

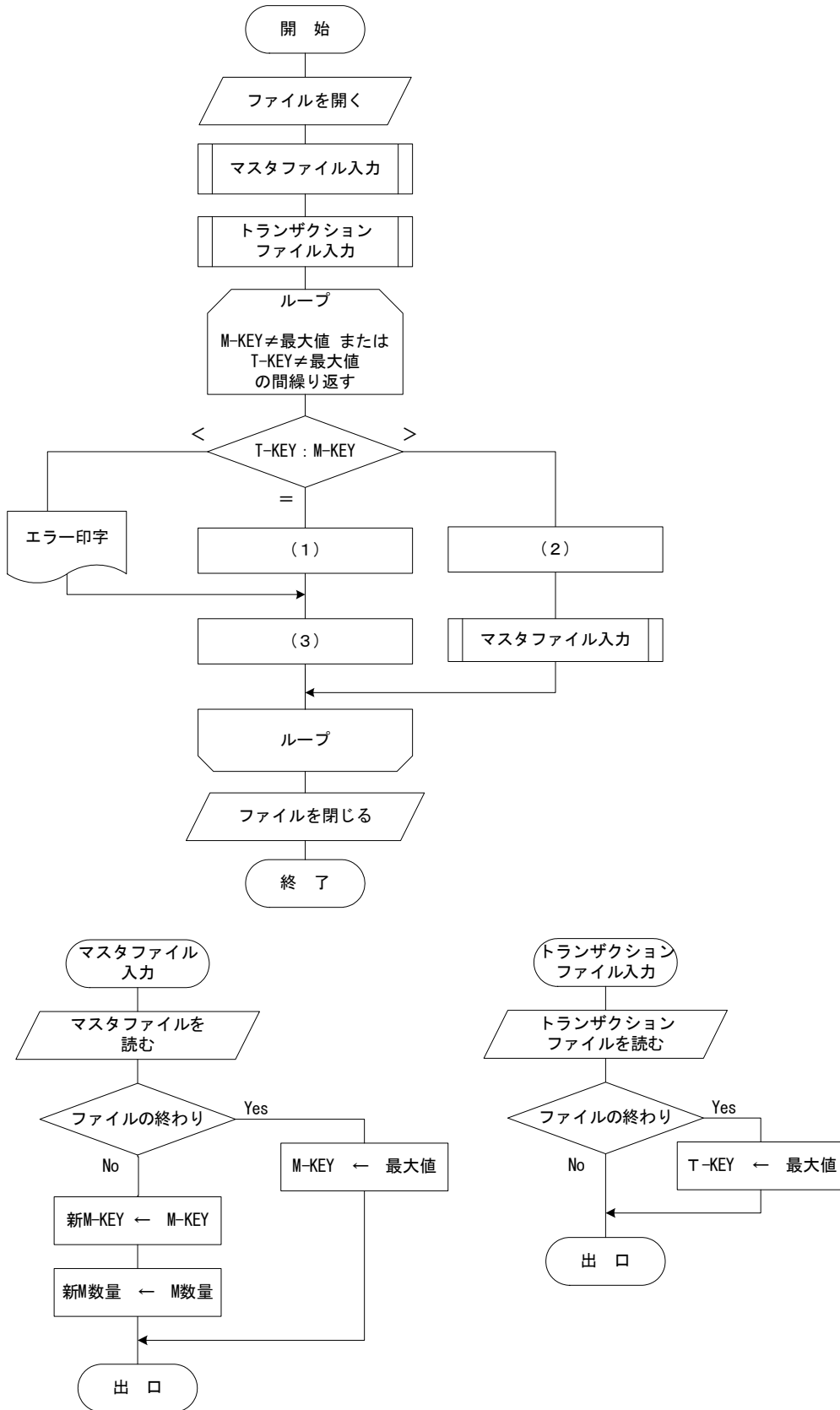


図 3 新マスタファイルを生成する流れ図

(1) ~ (3) の解答群

- ア. トランザクションファイル入力
- イ. マスタファイル入力
- ウ. 新 M 数量 ← T 数量
- エ. 新 M 数量 ← 新 M 数量 + T 数量
- オ. 新 M-KEY と新 M 数量を新マスタファイルへ出力
- カ. M-KEY と M 数量を新マスタファイルへ出力

<設問 2> 次の新マスタファイルの生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。

図 3 の流れ図に従って処理を行うが、トランザクションファイルの内容が誤ってレコードキーの昇順に整列されていない場合を考える。

マスタファイルとトランザクションファイルのレコードキーが図 4 のような順番であるときに、マスタファイルのレコードのうち、数量がトランザクションファイルの値と加算されて新マスタファイルに出力されるレコード数は (4) 件である。また、トランザクションファイルのレコードのうち、エラー印字されるレコードキーは (5) である。

マスタファイルのレコードキー

M-KEY
10110
10120
10210
20010
20100
22100
23120
25110
32120
32130

トランザクションファイルのレコードキー

T-KEY
10110
10110
23120
22100
25110

図 4 マスタファイルとトランザクションファイルのレコードキーの並び

(4) の解答群

- ア. 2
- イ. 3
- ウ. 4
- エ. 5

(5) の解答群

ア. 10110

イ. 22100

ウ. 23120

エ. 25110

オ. 10110 と 22100

カ. 10110 と 23120

キ. 10110 と 25110

ク. 22100 と 23120

ケ. 22100 と 25110

コ. 23120 と 25110

問題3 次の探索に関する説明を読み、設問に答えよ。

[探索アルゴリズム1]

配列の先頭から順に探索していく方法を線形探索という。図1の流れ図は一次元配列T ($T[i] : i=0, 1, \dots, 9$) を先頭から探索していき、探索値xと同じ値を見つけると、その要素番号を表示する。

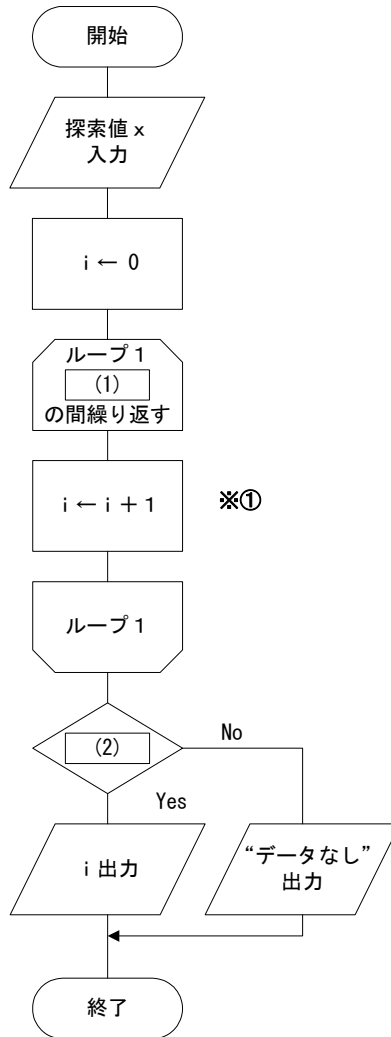


図1 線形探索

<設問1> 図1の に入れるべき適切な字句を解答群から選べ。

(1), (2) の解答群

ア. $i < 10 \text{ or } x \neq T[i]$

イ. $i < 10 \text{ and } x \neq T[i]$

ウ. $i < 0$

エ. $i < 10$

オ. $i \leq 10$

カ. $i > 0$

キ. $x \neq T[i]$

[探索アルゴリズム 2]

整列済みの配列に対して、配列の中央の値と探索する値の大小関係を用いて探索したい値が配列の前後のどちらにあるかを判断し探索していく方法を二分探索という。図 2 の流れ図は昇順に整列済みの一次元配列 $T(T[i] : i=0, 1, \dots, 9)$ を二分探索していき、探索値 x と同じ値を見つけると、その要素番号を表示する。

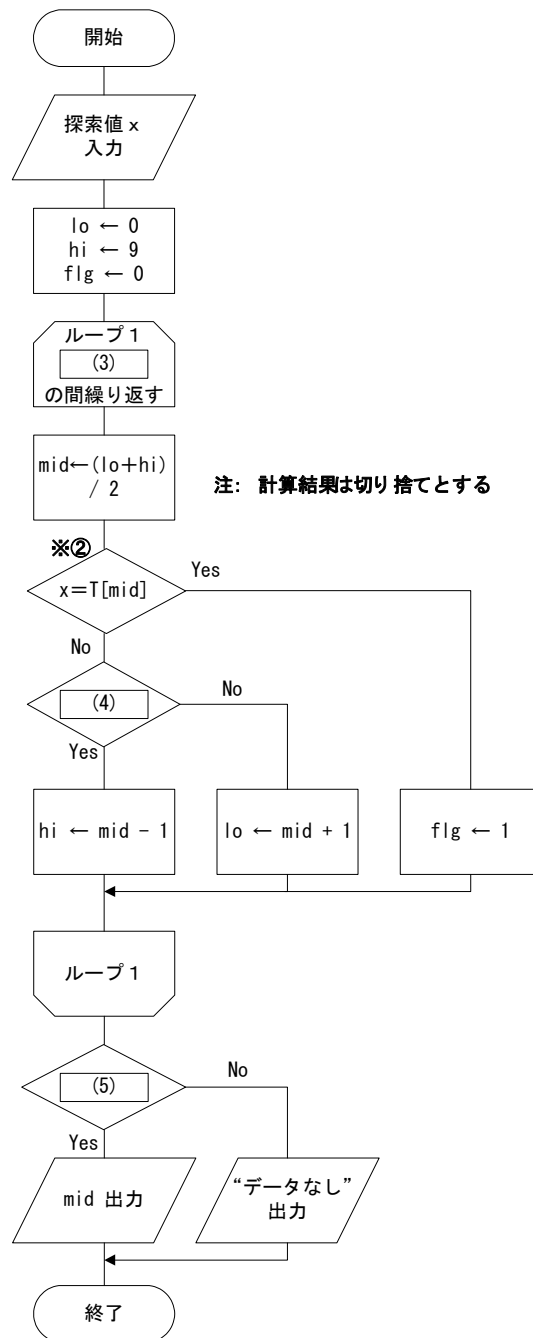


図 2 二分探索

<設問 2 > 図 2 の に入れるべき適切な字句を解答群から選べ。

(3) ~ (5) の解答群

- ア. $flg=0$ イ. $flg=1$ ウ. $x < T[mid]$
エ. $X \neq T[mid]$ オ. $x > T[mid]$ カ. $lo \leq hi$
キ. $lo \geq hi$ ク. $lo \leq hi$ and $flg=0$ ケ. $lo=hi$ or $flg=0$
コ. $lo \geq hi$ and $flg=1$

<設問 3 > [探索アルゴリズムの比較]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。

[探索アルゴリズムの比較]

図 1 と図 2 の流れ図に図 3 のデータを用いて探索して結果がでたとき、図 1 では※①の処理を (6) 回繰り返すのに対して、図 2 のアルゴリズムでは※②の処理を (7) 回繰り返す。

また、配列の要素数が 10 個のとき、図 1 では配列要素と探索値の最大比較回数は (8) 回なのに対して、図 2 では、最大比較回数は (9) 回になる。

配列 T

1	3	5	6	11	16	18	25	33	45
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]

探索値 x

16

図 3 配列 T と探索値 x

(6) ~ (9) の解答群

- ア. 1 イ. 2 ウ. 3 エ. 4 オ. 5
カ. 6 キ. 7 ク. 8 ケ. 9 コ. 10

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。

これより

< 選 択 問 題 >

選択問題はA，Bから，それぞれ1問選択し解答せよ。
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 A

- | | |
|------------|---------------|
| ・ C言語の問題 | 14 ページ～15 ページ |
| ・ 表計算の問題 | 16 ページ～17 ページ |
| ・ アセンブラの問題 | 18 ページ～19 ページ |

選択問題 B

- | | |
|------------|---------------|
| ・ C言語の問題 | 20 ページ～26 ページ |
| ・ 表計算の問題 | 27 ページ～30 ページ |
| ・ アセンブラの問題 | 31 ページ～35 ページ |

選択問題A C言語の問題

次のC言語プログラムに関する説明を読み、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]

ある文字列の中から特定の文字列を検索し、別の文字列に置き換える関数rplである。なお、置換後の文字列は255文字以下であるものとする。

実行例：

文字列	T	O	K	Y	O	-	K	Y	O	T	O	¥0
検索文字列	K	Y	O	¥0								
置換文字列	M	Y	A	O	¥0							
置換後文字列	T	O	M	Y	A	O	-	M	Y	A	O	¥0

[関数の説明]

rpl 関数

引 数：文字列，検索文字列，置換文字列

機 能：文字列の中から検索文字列を探し，置換文字列で置き換える

戻り値：置換後の文字列

[プログラム]

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char *rpl(char *text, char *from, char *to) {
    int    i, p, q, r, n1, n2, n3, wk, sw;
    char   *ret;
    ret = (char *)malloc(256); /* 置換後の文字列を格納する領域を確保 */
    n1 = strlen(text);        /* 文字列の長さ */
    n2 = strlen(from);       /* 検索文字列の長さ */
    n3 = strlen(to);         /* 置換文字列の長さ */
    p = 0;                    /* 元の文字列の添字を初期化 */
    r = 0;                    /* 置換後文字列の添字を初期化 */
    while ( p < n1 - n2 + 1 ) {
        q = 0;                /* 検索文字列の添字を初期化 */
        sw = 0;               /* 判定スイッチを初期化 */
         (1) ;
```



```

while ( q < n2 && (2) ) { /* 検索文字列がある場所を探す */
    if ( *(text + p) == *(from + q) ) {
        (3)
    } else
        sw = 1;
}
if ( (4) ) { /* 不一致の時は1文字を複写 */
    p = wk;
    *(ret + r) = *(text + p);
    (5)
} else { /* 一致した時は置換文字列を複写 */
    for ( i = 0; i < n3; i++ ) {
        *(ret + r) = *(to + i);
        r++;
    }
}
}
while ( p < n1 ) { /* 残っている文字列を転送 */
    *(ret + r) = *(text + p);
    (5)
}
*(ret + r) = '\0';
return ret;
}

```

(1) の解答群

- | | |
|-----------|-----------|
| ア. wk = 0 | イ. wk = p |
| ウ. ret++ | エ. *ret++ |

(2) の解答群

- | | |
|------------------------|------------------------|
| ア. sw == 0 | イ. sw == 1 |
| ウ. *(text + p) == '\0' | エ. *(from + q) != '\0' |

(3) , (5) の解答群

- | | | |
|---------|---------|---------|
| ア. p++; | イ. q++; | ウ. r++; |
| エ. p++; | オ. p++; | カ. q++; |
| q++; | r++; | r++; |

(4) の解答群

- | | |
|------------|------------|
| ア. p > n1 | イ. q > n1 |
| ウ. sw == 0 | エ. sw == 1 |

選択問題 A 表計算の問題

関数、式の仕様は次のとおりである。

AVERAGE 関数

指定された範囲に含まれる数値の入ったセルの平均を求める。

書式：AVERAGE(範囲)

AVERAGEIF 関数

指定された検索範囲内の検索条件に一致するセルと対応する平均対象範囲のセルの平均を求める。

書式：AVERAGEIF(検索範囲, 検索条件, 平均対象範囲)

COUNT 関数

指定された範囲に含まれるセルのうち、数値の入っているセルの個数を求める。

書式：COUNT(範囲)

COUNTIF 関数

指定された範囲に含まれるセルのうち、検索条件に一致するセルの個数を求める。

書式：COUNTIF(範囲, 検索条件)

J 学院では、ある試験の結果を集計するのに表計算ソフトを用いることにした。試験の実施概要は、以下のとおりである。

- ・科目 1 から科目 4 の 4 科目からなる試験である。
- ・1 科目 25 点満点で 16 点以上が合格となる。
- ・4 科目すべて合格すると資格取得となる。
- ・科目合格制度があり、過去に合格した科目は受験しなくてもよい。
- ・入力されているデータに誤りはないものとする。

本年 12 月に実施した試験の結果を以下の表 1 にまとめた。

表 1 成績表

	A	B	C	D	E	F	G	H
1	学籍番号	名前	科目1	科目2	科目3	科目4	試験結果	平均点
2	J201101	相沢一秋	20	18	19	25	合格	20.5
3	J201102	伊藤春名	免除	24	15	18	不合格	19.0
4	J201103	上野みゆき	18	21	25	16	合格	20.0
5	J201104	遠藤仁美	免除	10	免除	8	不合格	9.0
6	J201105	小野寺信二	免除	免除	免除	欠席	不合格	
7	J201106	奥寺武明	12	17	21	18	不合格	17.0
8	J201107	木村太郎	15	18	21	免除	不合格	18.0
9	J201108	草野誠	22	16	25	19	合格	20.5
10	J201109	千葉雅和	欠席	欠席	欠席	欠席	不合格	
11	J201110	広沢純一	18	21	18	免除	合格	19.0
12	J201111	山田幸谷	21	24	12	欠席	不合格	19.0
13	J201112	渡辺仁	20	25	24	19	合格	22.0
14							合格者平均	20.4

科目 1 から科目 4 のそれぞれの項目に点数を入力した。ただし、過去に科目合格済みである場合は「免除」、欠席した場合は「欠席」と入力した。

<設問> 成績表に関する次の記述の に入れるべき適切な式を解答群から選べ。

試験結果を表示するために、以下の式をセル G2 に入力した。すべての科目が合格または科目合格済みである場合は合格と表示、そうでない場合は不合格と表示する。なお、この式はセル G3 から G13 に複写するものとする。

=IF((1) + (2) = 4, "合格", "不合格")

(1) の解答群

- | | |
|--------------------------|---------------------------|
| ア. COUNTIF(C2:F2, ">16") | イ. COUNTIF(C2:F2, ">=16") |
| ウ. COUNTIF(C2:F2, "<16") | エ. COUNTIF(C2:F2, "<=16") |

(2) の解答群

- | | |
|--------------------------|---------------------------|
| ア. COUNTIF(C2:F2, "=欠席") | イ. COUNTIF(C2:F2, "<>欠席") |
| ウ. COUNTIF(C2:F2, "=免除") | エ. COUNTIF(C2:F2, "<>免除") |

学生ごとの平均を求めるために、以下の式をセル H2 に入力した。ただし、今回 1 科目も受験していない場合には空欄とする。なお、この式はセル H3 から H13 に複写するものとする。

=IF((3), "", AVERAGE(C2:F2))

(3) の解答群

- | | |
|-------------------|-------------------|
| ア. COUNT(C2:F2)=0 | イ. COUNT(C2:F2)=1 |
| ウ. COUNT(C2:F2)>0 | エ. COUNT(C2:F2)<4 |

合格者の平均を求めるために、以下の式をセル H14 に入力した。ただし、合格者が 1 名もない場合には「合格者なし」を表示する。

=IF((4), "合格者なし", (5))

(4) の解答群

- | | |
|-----------------------------|-----------------------------|
| ア. COUNTIF(G2:G13, "=合格")>0 | イ. COUNTIF(G2:G13, "=合格")=0 |
| ウ. COUNTIF(G2:G13, "=合格")<4 | エ. COUNTIF(G2:G13, "=合格")=4 |

(5) の解答群

- ア. AVERAGE(G2:G13)
- イ. AVERAGE(H2:H13)
- ウ. AVERAGEIF(H2:H13, "=合格", G2:G13)
- エ. AVERAGEIF(G2:G13, "=合格", H2:H13)

選択問題A アセンブラの問題

次のCASL IIプログラムに関する説明を読み、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]

2つの整数AとBの最大公約数をユークリッドの互除法により求めるプログラムである。なおAとBともに0より大きい値とし、計算結果によるあふれは起きないものとする。

ユークリッドの互除法により最大公約数を求める手順は、次のようになる。

- ① AをBで割った余りをRとする。
- ② Rが0ならばBが最大公約数である。
- ③ Rが0でなければ、Bの値をAへ、Rの値をBへ代入して①へ戻る。

プログラムでは、最大公約数をGCM番地へ格納して終了する。

[プログラム]

```
PROG      START
          LD      GR1, A
          LD      GR2, B
LOOPM     CALL    MOD
          CPA     GR0, =0
          JZE     FINISH      ;最大公約数が求めればFINISHへ分岐
          □□□□ (1)
          □□□□ (2)
          JUMP    LOOPM
FINISH    □□□□ (3)      ;最大公約数の保存
          RET
A         DS     1
B         DS     1
GCM       DS     1
          END
```

[副プログラム]

```
-----  
; GR1の値をGR2の値で割った余りをGR0へ求める副プログラム  
; 引き算の繰り返しで計算を行う  
-----  
MOD      START  
          LD      GR0, GR1  
LOOPS    SUBA   GR0, GR2  
          (4)  
          JZE   RETURN  
          (5)  
RETURN   RET  
        END
```

(1), (2) の解答群

ア. LD GR0, GR1 イ. LD GR0, GR2 ウ. LD GR1, GR0
エ. LD GR1, GR2 オ. LD GR2, GR0 カ. LD GR2, GR1

(3) の解答群

ア. ST GR0, GCM イ. ST GR1, GCM ウ. ST GR2, GCM

(4) の解答群

ア. JPL LOOPS イ. JMI LOOPS ウ. JNZ LOOPS

(5) の解答群

ア. ADDA GR0, GR0 イ. ADDA GR0, GR1 ウ. ADDA GR0, GR2

選択問題B C言語の問題

次のC言語プログラムに関する説明を読み、設問に答えよ。

[ダイクストラ法の説明]

ダイクストラ法とは、グラフ理論における最短経路を求めるためのアルゴリズムであり、カーナビゲーションシステムやルーティングプロトコルでも使用されている。

ダイクストラ法では、下の図1のように○印で表している地点と地点を結ぶ距離をもとに計算する。

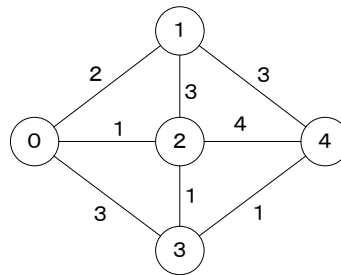


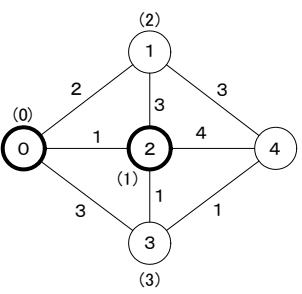
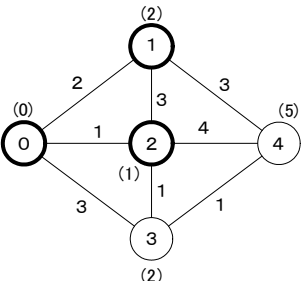
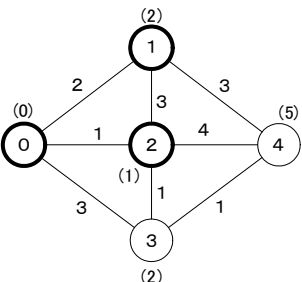
図1 地点と地点を結ぶ距離

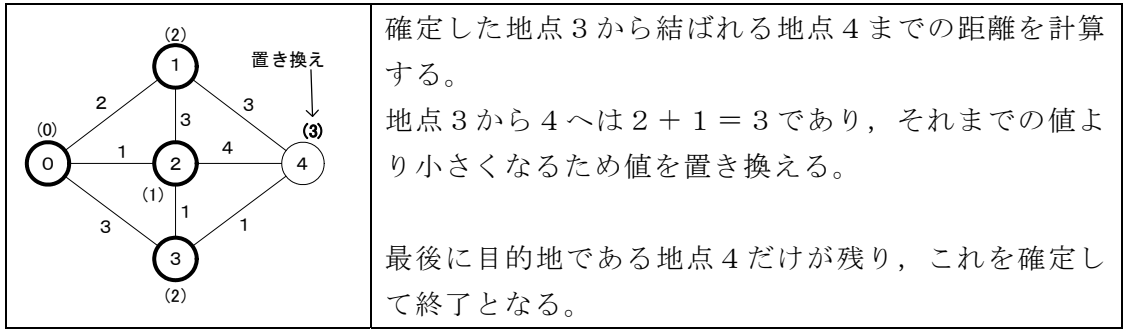
地点0から地点4までの最短経路を求めるには、次のような手順で行う。

1. 出発地点の距離を0として確定する。
2. 確定地点と結ばれたすべての地点について、出発地点からそれぞれの地点までの距離を加算して、各地点までの距離とする。もし、結ばれた地点の距離が、それまでの距離より小さければ置き換える。ここで計算した結果は、未確定要素である。
3. 未確定要素の中から一番小さい値を選び確定要素とする。
4. すべての未確定要素がなくなるまで上記2と3を繰り返す。

具体的には次のようになる。

	<p>出発地点の0を距離0として確定する。地点にかっこで書かれた値は距離である。</p>
	<p>地点0から結ばれる地点1, 2, 3までの距離をそれぞれ計算する。この時点で地点1, 2, 3までの距離は未確定要素である。</p>

	<p>未確定要素である地点 1, 2, 3 の中で一番小さい距離は地点 2 の「1」なので、地点 2 が確定要素となる。</p>
	<p>確定した地点 2 から結ばれる地点 1, 3, 4 までの距離を計算する。 地点 2 から 1 へは $1 + 3 = 4$ であるので、以前に計算した値より大きいので距離は置き換えない。 地点 2 から 3 への距離は $1 + 1 = 2$ であるので、以前に計算した値より小さいから距離を置き換える。 なお、すでに確定している地点 0 との計算は行わない。</p>
	<p>未確定要素である地点 1, 3, 4 の中で一番小さい値を持つ地点は、地点 1 と 3 である。値が同じ場合はどちらを採用しても良いので、ここでは地点番号が小さい地点 1 を確定要素とする。</p>
	<p>確定した地点 1 から結ばれる地点 4 までの距離を計算する。 地点 1 から 4 へは $2 + 3 = 5$ であり、それまでの値以上となるので距離は置き換えない。</p>
	<p>未確定要素である地点 3, 4 の中で一番小さい値を持つ地点 3 を確定要素とする。</p>



距離の計算をして距離を置き替えた時に、どの地点から計算したものを記憶しておき、最後に記憶した地点を逆にたどることで最短経路が求められる。この例では、出発地点からの最短経路は $0 \rightarrow 2 \rightarrow 3 \rightarrow 4$ となる。

<設問1> 次のダイクストラ法による最短経路に関する記述中の に入れるべき適切な字句を解答群から選べ。

下の図2のような経路において、ダイクストラ法により地点1から地点5までの最短経路を求めると、その経路は (1) となり、その距離は (2) である。なお、○印で囲まれたものは地点であり、地点同士を結ぶ直線上の数値は距離を表している。

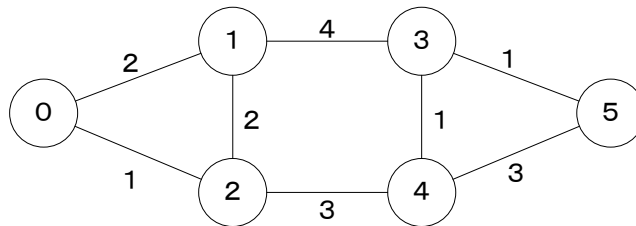


図2 経路図

(1) の解答群

- ア. $0 \rightarrow 1 \rightarrow 3 \rightarrow 5$
- イ. $0 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5$
- ウ. $0 \rightarrow 2 \rightarrow 4 \rightarrow 5$
- エ. $0 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 5$

(2) の解答群

- ア. 6
- イ. 7
- ウ. 8
- エ. 10

<設問 2> 次のプログラムの説明および関数の説明を読み、プログラム中の に入るべき適切な字句を解答群から選べ。

[プログラムの説明]

Dijkstra関数は、出発地点と目的地点を受け取り、ダイクストラ法により最短経路を求めて、出発地点からの経路を出力するものである。

このプログラムは、図3のように二次元配列MAPを使って、地点間の距離とつながりを表している。図3は図1の経路を表現したものである。直接結ばれていない地点および自分自身の距離は0になっている。この情報は、Dijkstra関数が呼び出される前に二次元配列MAPに読み込まれる。また、地点数は変数Nに読み込まれる。なお、各地点間の距離は9999未満とし、地点数は100以下とする。

N	5
MAP	
	0 1 2 3 4
0	0 2 1 3 0
1	2 0 3 0 3
2	1 3 0 1 4
3	3 0 1 0 1
4	0 3 4 1 0

図3 二次元配列MAPの構造と変数N

また、各地点の情報は、構造体nodeを使って管理する。

構造体node		
開始地点からの距離	確定フラグ	前の地点

図4 構造体nodeの形式

なお、出発地点から目的地点までは、いずれかの経路によって必ず結ばれているものとし、各地点は1つ以上の地点と必ず結ばれているものとする。

[関数の説明]

Dijkstra 関数

引 数：出発地点，目的地点

機 能：ダイクストラ法により最短経路を求める処理の本体

戻り値：なし

CalcDistance 関数

引 数：確定地点

機 能：確定地点からつながる各地点の距離を計算する

戻り値：なし

Min 関数

引 数：なし

機 能：未確定要素地点の中から最小の距離を持つ要素を探す

戻り値：未確定地点の要素位置

PrintRoute 関数

引 数：出発地点，経由地点

機 能：出発地点から目的地点までの経路をたどり，その地点を出力する

戻り値：なし

[プログラム]

```
#include <stdio.h>
#include <limits.h>
#define MAX 100          /* 地点の最大数は100までとする */

void Dijkstra(int, int);
void CalcDistance(int);
int Min(void);
void PrintRoute(int, int);

typedef struct {
    int Distance;        /* 距離を保持する */
    int Decision_sw;    /* 確定要素となったとき1にする */
    int From;           /* 前の地点を保持する */
} node;

int MAP[MAX][MAX];     /* 地点間の距離とつながりが格納される */
int N;                 /* 地点数が格納される */
node NODE[MAX];        /* 各地点の情報を保持する */

void Dijkstra(int StartPosi, int EndPosi) {
    int i, loop_sw, posi;
    for (i=0; i<N; i++) {          /* 初期化 */
        NODE[i].Distance = INT_MAX; /* 距離は最大値としておく */
        NODE[i].Decision_sw = 0;   /* 確定スイッチは0にする */
        NODE[i].From = -1;         /* 前の場所は-1にする */
    }
    NODE[StartPosi].Distance = 0;  /* 出発位置を確定要素とする */
    NODE[StartPosi].Decision_sw = 1;
    loop_sw = 0;
    posi = StartPosi;             /* 出発地点を最初の位置とする */
```

```

while ( (3) ) { /* 未確定要素が無くなるまで繰り返す */
    CalcDistance(posi); /* 現在の確定位置からつながる距離の計算*/
    posi = Min(); /* 未確定要素中の距離の最小値を求める */
    if (posi == INT_MAX) /* 未確定要素がなければ終了 */
        loop_sw = 1;
    else
        NODE[posi].Decision_sw = 1; /* 確定要素とする */
}
PrintRoute(StartPosi, EndPosi); /* 最短経路の出力 */
}

```

/* 確定位置からつながる地点の距離を計算 */

```

void CalcDistance(int posi) {
    int i, k;
    for (i=0; i<N; i++) {
        if (MAP[posi][i]>0 && NODE[i].Decision_sw == 0 && i != posi) {
            k = (4); /* 距離の計算 */
            if (k < NODE[i].Distance) { /* 置き換えるかを判断 */
                NODE[i].Distance = (5);
                NODE[i].From = (6);
            }
        }
    }
}

```

/* 未確定要素の最小値を求める */

```

int Min(void) {
    int i, m, min_posi;
    m = HIGHVALUE;
    min_posi = INT_MAX;
    for (i=0; i<N; i++) {
        if (NODE[i].Distance < m && NODE[i].Decision_sw == 0) {
            min_posi = i;
            (7);
        }
    }
    return min_posi;
}

```

```

/* 最短経路出力 */
void PrintRoute(int StartPosi, int posi) {
    if (posi == StartPosi) {
        printf("%d", StartPosi);
        return;
    } else {
        PrintRoute(StartPosi, NODE[posi].From);
    }
    printf(" → %d", posi);
}

```

(3) の解答群

- ア. loop_sw == 0
- イ. loop_sw == 1
- ウ. NODE[posi].Decision_sw == 0
- エ. NODE[posi].Decision_sw == 1

(4) の解答群

- ア. NODE[i].Distance + MAP[i][posi]
- イ. NODE[i].Distance + MAP[posi][i]
- ウ. NODE[posi].Distance + MAP[i][posi]
- エ. NODE[posi].Distance + MAP[posi][i]

(5) , (6) の解答群

- ア. i イ. k ウ. N エ. posi

(7) の解答群

- ア. m = NODE[min_posi].Distance
- イ. m = NODE[N].Distance
- ウ. m = NODE[i].Distance
- エ. m = NODE[m].Distance

選択問題B 表計算の問題

関数、式の仕様は次のとおりである。

COUNTIF 関数

指定された範囲に含まれるセルのうち、検索条件に一致するセルの個数を求める。

書式：COUNTIF(範囲, 検索条件)

AND 関数

全ての条件式が真のとき、真を返す。

書式：AND(条件式1, 条件式2・・・)

RANK 関数

範囲内にある数値の順位を返す。順序が0の場合、降順でつけた順位を返し、0以外の場合、昇順でつけた順位を返す

書式：RANK(数値, 範囲, 順序)

車両販売を行っているJ社では、車の見積もりを迅速にするために付属品に関する見積もり一覧表を作成することにした。

付属品は、セールス期間内(セールス開始日からセールス終了日の期間)であれば、付属品ごとの割引が適用される。さらに、4種類以上の付属品を購入すると金額合計の10%が特別割引になり、6種類以上では同様に20%が特別割引になる。

表1 付属品見積表

	A	B	C	D	E	F	G	H
1	付属品見積表							
2	日付	2009/9/6						
3								
4	商品コード	商品名	単価	セールス開始日	セールス終了日	割引率	数量	金額
5	P00001	カーナビゲーションシステム	220,000	2009/8/1	2010/1/31	20%	1	176,000
6	P00002	CDチューナー	40,000	2010/1/1	2010/3/31	10%		
7	P00003	ETC	16,000	2009/8/1	2009/12/31	10%	1	14,400
8	P00004	バックモニター	25,000	2009/8/1	2009/12/31	20%	1	20,000
9	P00005	地上デジタルチューナー	68,000	2009/11/1	2010/1/31	30%		
10	P00006	ドライブレコーダー	52,500	2009/11/1	2010/3/31	10%		
11	P00007	サイドバイザー	10,000	2009/8/1	2010/3/31	10%	1	9,000
12	P00008	リアスポイラー	25,000	2009/8/1	2010/3/31	10%	1	22,500
13	P00009	フロアマット	24,000	2009/8/1	2010/3/31	20%	3	57,600
14								
15								金額合計
16								299,500
17								
18								特別割引率
19								0.2
20								
21								提供金額
22								239,600

<設問 1 > 次の割引率の計算に関する記述中の に入れるべき適切な式を解答群から選べ。ただし、値の入っていないセルを計算式に使用すると、エラーが発生するものとする。

セル B2 は、常に現在の日付が表示される。

列 G の数量には購入する数量を入力する。

列 H の金額は、各付属品の数量を入力すると、セールス期間内であれば単価×数量に割引が適用された値を表示し、セールス期間外であれば単価×数量をそのまま表示する。したがって、セル H5 に入力する式は

IF(G5="", "", IF((1)))

となり、セル H6 から H13 まで複写する。

セル H16 の金額合計は、セル H5 から H13 の合計なので、SUM(H5:H13)となる。

セル H19 の特別割引率は、購入した付属品が何種類かにより値を求める。したがって、セル H19 に入力する式は

IF((2), 0.2, IF((3), 0.1, ""))

となる。

セル H22 の提供金額の計算は、H16*(1-H19)を入力すると、 (4) という問題が発生するので式は (5) となる。

(1) の解答群

- ア. AND(\$B\$2<=E5, \$B\$2>=D5), C5*G5*(1-F5), C5*G5
- イ. AND(\$B2<=E5, \$B2>=D5), C5*G5*(1-F5), C5*G5
- ウ. AND(\$B2<=E5, \$B2>=D5), C5*G5, C5*G5*(1-F5)
- エ. AND(\$B\$2<=E5, \$B\$2>=D5), C5*G5, C5*G5*(1-F5)

(2) , (3) の解答群

- ア. COUNTIF(G5:G13, ">=1")>=6
- イ. COUNTIF(G5:G13, ">=1")>=4
- ウ. COUNTIF(G5:G13, ">=1")<6
- エ. COUNTIF(G5:G13, ">=1")<4

(4) の解答群

- ア. 特別割引が適用されないとエラーが発生する。
- イ. セールス期間割引が適用されるとエラーが発生する。
- ウ. セールス期間割引と特別割引が同時に適用されると計算結果が正しく表示されない。
- エ. 循環参照が生じる。

(5) の解答群

- ア. IF(H19="", H16, H16*(1-H19))
- イ. IF(H16="", H16, H16*(1-H19))
- ウ. IF(COUNTIF(G5:G13, ">=1")>=6, H16, H16*(1-H19))
- エ. H22*(1-H19)

つぎに、顧客に付属品の購買意欲を持たせるために付属品の販売順位表を作成することにした。

<設問 2 > 次の順位付けに関する記述中の に入れるべき字句を解答群から選べ。

付属品販売順位表は、4 種類の車種に対してそれぞれ付属品販売台数の降順に順位を表示する表である。台数はそれぞれの車両購入時に付属品を取り付けた台数を表し、順位は各車種別の台数に対する順位を表すものとする。

表 2 付属品販売順位表 (案 1)

	A	B	C	D	E	F	G	H	I	J	K
1											
2	商品名	JC2D		JC4D		JWST		JWDL		総合	
3		台数	順位	台数	順位	台数	順位	台数	順位	計	総合順位
4	カーナビゲーションシステム	40	3	60	2	15	5	57	1	172	2
5	CDチューナー	15	6	25	5	31	2	5	8	76	6
6	ETC	60	2	85	1	41	1	40	3	226	1
7	バックモニター	20	5	19	6	5	8	54	2	98	4
8	地上デジタルチューナー	25	4	32	4	5	8	35	5	97	5
9	ドライブレコーダー	5	7	5	8	11	6	25	6	46	8
10	サイドバイザー	105	1	37	3	25	3	0	9	167	3
11	リアスポイラー	5	7	2	9	16	4	40	3	63	7
12	フロアマット	3	9	9	7	7	7	20	7	39	9
13	合計	278		274		156		276		984	

順位を表示するためには、セル C4 に (6) を入力し、それをセル C5 から C12 まで複写し、さらに E, G, I および K 列に複写する。

また、表 2 のように付属品購入台数の順位で顧客に判断してもらうのではなく、車種ごとの順位を合計した結果の昇順によって総合順位を求め、判断してもらう考え方もある。

表3 付属品販売順位表（案2）

	A	B	C	D	E	F	G	H	I	J	K
1											
2	商品名	JC2D		JC4D		JWST		JWDL		総合	
3		台数	順位	台数	順位	台数	順位	台数	順位	順位計	総合順位
4	カーナビゲーションシステム	40	3	60	2	15	5	57	1	11	2
5	CDチューナー	15	6	25	5	31	2	5	8	21	4
6	ETC	60	2	85	1	41	1	40	3	7	1
7	バックモニター	20	5	19	6	5	8	54	2	21	4
8	地上デジタルチューナー	25	4	32	4	5	8	35	5	21	4
9	ドライブレコーダー	5	7	5	8	11	6	25	6	27	8
10	サイドバイザー	105	1	37	3	25	3	0	9	16	3
11	リアスポイラー	5	7	2	9	16	4	40	3	23	7
12	フロアマット	3	9	9	7	7	7	20	7	30	9
13	合計	278		274		156		276			

この場合は表3のように列Jの順位計に順位の合計を表示し、その後セルK4に (7) を入力し、それをセルK5からK12に複写する。

この表3のような方法を使用する理由としては、表2の場合では (8) 状態になる可能性があるためである。

(6) の解答群

- ア. RANK(B4, \$B\$4:\$B\$12, 0)
- イ. RANK(B4, \$B4:\$B12, 1)
- ウ. RANK(B4, B\$4:B\$12, 0)
- エ. RANK(B4, \$B\$4:B12, 1)

(7) の解答群

- ア. RANK(J4, \$J\$4:\$J\$12, 0)
- イ. RANK(J4, \$J4:\$J12, 0)
- ウ. RANK(J4, J\$4:J\$12, 1)
- エ. RANK(J4, J4:J12, 1)

(8) の解答群

- ア. 付属品の購入が多い特定の車種の影響が非常に強くなる。
- イ. 車両販売台数の合計と、ほぼ等しくなる。
- ウ. 販売数の少ない特定の車種の順位と販売数の多い特定の車種の順位が同格に扱われる。
- エ. 付属品の購入数と比例する順位ではなくなる。

選択問題B アセンブラの問題

次のC A S L IIプログラムに関する説明を読み、設問に答えよ。

[プログラムの説明]

数式の文字列を入力し、その計算をしてANS番地に答えを格納するプログラムである。入力すべき式は、次の条件のとおりである。

- ・加算の式を入力する。
- ・数式に使われる文字は'0' ~ '9'の数字と'+', 'および' '='の記号である。
- ・数式は必ず数字で始まる。
- ・数式の最後は必ず数字と'='の並びで終わる。
- ・入力する数式の文字数は4以上80以内である。

なお、計算結果によるあふれは起こらないものとする。

(例) 「12+3+415=」と入力した場合

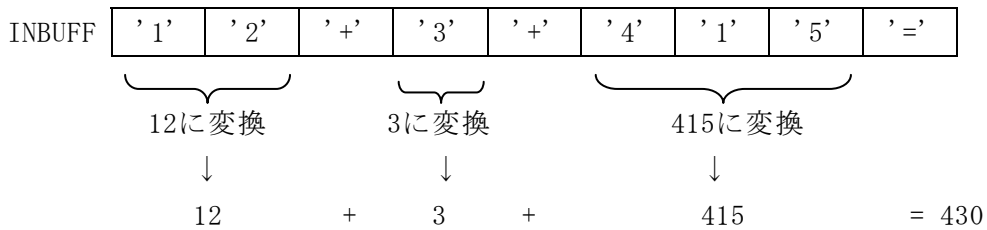


図1 処理の内容

副プログラムDECBINは、文字列で受け取った10進数数値を2進数に変換する。DECBINは、変換する文字列の先頭位置をGR1、文字数をGR2に格納して呼び出され、2進数へ変換した値をGR0に設定して戻る。

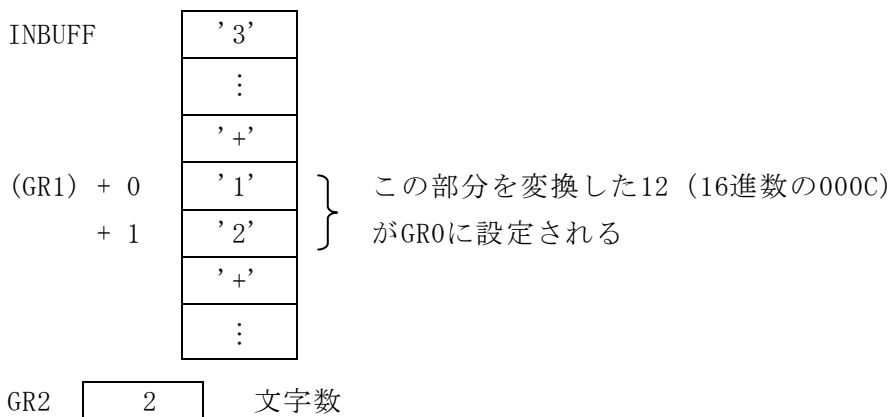


図2 副プログラムDECBINの扱うデータ

<設問 1 > 次のプログラム中の に入れるべき適切な字句を解答群から選べ。

[プログラム]

```
100  PROG      START
110           IN    INBUFF, LEN      ;数式の入力
120           LD    GR1, =0          ;INBUFFの添え字初期化
130           ST    GR1, ANS         ;結果を求める場所を初期化
140  LOOP1     ST    GR1, SVGR1A     ;INBUFFの添え字を一時保存
150           LD    GR2, =0          ;文字数のカウンタを初期化
160  LOOP2     LD    GR0, INBUFF, GR1
170           CPL   GR0, PLUS        ; '+'かを判断
180           JZE   HENKAN
190           CPL   GR0, EQUAL       ; '='かを判断
200           JZE   HENKAN
210           ADDL  GR1, =1           ; 次の文字を調べるための
220           ADDA  GR2, =1           ; カウントアップ
230            (1)
240  HENKAN    ST    GR1, SVGR1B     ; GR1を一時保存
250           LD    GR1, SVGR1A     ; 数値の先頭位置を取り出す
260            (2) ; 絶対アドレスへ変換
270           CALL  DECBIN
280           ADDA  GR0, ANS         ; 加算処理
290           ST    GR0, ANS
300           LD    GR1, SVGR1B     ; GR1を復帰
310           LD    GR0, INBUFF, GR1
320           CPL   GR0, EQUAL       ; '='であったならば終わる
330           JZE   FINISH
340           ADDL  GR1, =1
350            (3)
360  FINISH    RET
370  EQUAL     DC    '='
380  PLUS      DC    '+'
390  ANS       DS    1
400  SVGR1A    DS    1
410  SVGR1B    DS    1
420  INBUFF    DS    80
430  LEN       DS    1
440           END
```

[副プログラム]

```

450 ;-----
460 ; GR1から始まる領域に格納されたGR2の文字数のデータを
470 ; 2進数に変換してGR0へ格納し戻る
480 ;-----
490 DECBIN      START
500             RPUSH
510             LD      GR0, =0           ;結果の初期化
520 LOOP       (4)                       ;
530             ST      GR0, WORK       ; } 以前の値を10倍する
540             SLA     GR0, 2           ; }
550             ADDA    GR0, WORK       ; }
560             LD      GR3, 0, GR1     ;
570             SUBL    GR3, ZERO       ; } 該当する桁を加える
580             ADDA    GR0, GR3        ; }
590             ADDL    GR1, =1
600             SUBA    GR2, =1
610             (5)                       ;全ての文字を変換したか
620             RPOP
630             RET
640 WORK        DS      1
650 ZERO        DC      '0'
660             END

```

(1) , (3) の解答群

ア. JUMP LOOP1	イ. JMI LOOP1
ウ. JZE LOOP1	エ. JUMP LOOP2
オ. JMI LOOP2	カ. JZE LOOP2

(2) の解答群

ア. LAD GR1, 0, GR1	イ. LAD GR1, 1, GR1
ウ. LAD GR1, INBUFF	エ. LAD GR1, INBUFF, GR1

(4) の解答群

ア. ADDA GR0, WORK	イ. ADDA GR0, GR1
ウ. SLA GR0, 1	エ. SLA GR0, 2

(5) の解答群

ア. JZE LOOP	イ. JPL LOOP
ウ. JUMP LOOP	エ. JMI LOOP

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

次の図 3 のデータを入力した場合、240 行の命令は (6) 回実行される。また、最初に 240 行の命令が実行されるとき GR1 の値は (7) である。

'1'	'2'	'5'	'8'	'+'	'2'	'3'	'2'	'1'	'='
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

図 3 入力されるデータ

(6) , (7) の解答群

- | | | |
|------|------|------|
| ア. 0 | イ. 1 | ウ. 2 |
| エ. 3 | オ. 4 | カ. 5 |

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

このプログラムで、規定している文字以外のデータが入力された場合は 'ERROR' と表示するように変更したい。そこで、プログラムの追加および修正をすることにした。なお、COMET II で使用する文字の符号表は次のとおりである。

表 1 文字符号表

列 行	02	03	04	05	06	07
0	間隔	0	@	P	`	p
1	!	1	A	Q	a	q
2	"	2	B	R	b	r
3	#	3	C	S	c	s
4	\$	4	D	T	d	t
5	%	5	E	U	e	u
6	&	6	F	V	f	v
7	'	7	G	W	g	w
8	(8	H	X	h	x
9)	9	I	Y	i	y
10	*	:	J	Z	j	z
11	+	;	K	[k	{
12	,	<	L	¥	l	
13	-	=	M]	m	}
14	.	>	N	^	n	~
15	/	?	O	_	o	

まず、350 行の次と 430 行の次に以下の命令とデータを追加した。

351	ERROR	OUT	EMESS, ELEN
------------	--------------	------------	--------------------

431	CZERO	DC	'0'
432	CNINE	DC	'9'
433	EMESS	DC	'ERROR'
434	ELEN	DC	5

次に、

(8)

の次に以下の命令を追加した。

	CPL	GR0, CZERO
	JMI	ERROR
	CPL	GR0, CNINE
	JPL	ERROR

(8) の解答群

ア. 160 行

イ. 200 行

ウ. 230 行

エ. 330 行

<メモ欄>

<メモ欄>

