

次の問8は必須問題です。必ず解答してください。

問8 次のプログラムの説明及びプログラムを読んで、設問1～4に答えよ。

文字列の誤りを検出するために、N種類の文字に0, 1, …, N-1の整数値を重複なく割り当て、検査文字を生成するプログラムと、元となる文字列の末尾に検査文字を追加した検査文字付文字列を検証するプログラムである。ここで扱う30種類の文字、及び文字に割り当てた数値を、表1に示す。空白文字は“ ”と表記する。

表1 文字、及び文字に割り当てた数値

文字	□	.	,	?	a	b	c	d	e	f	g	h	i	j	k
数値	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

文字	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
数値	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

[プログラムの説明]

検査文字の生成と検査文字付文字列の検証の手順を示す。

(1) 検査文字の生成

- ① 文字列の末尾の文字を1番目の文字とし、文字列の先頭に向かって奇数番目の文字に割り当てた数値を2倍してNで割り、商と余りの和を求め、全て足し合わせる。
- ② 偶数番目の文字に割り当てた数値は、そのまま全て足し合わせる。
- ③ ①と②の結果を足し合わせる。
- ④ Nから、③で求めた総和をNで割った余りを引く。さらにその結果を、Nで割り、余りを求める。求めた数値に対応する文字を検査文字とする。

## (2) 検査文字付文字列の検証

- ① 検査文字付文字列の末尾の文字を 1 番目の文字とし、文字列の先頭に向かって偶数番目の文字に割り当てた数値を 2 倍して  $N$  で割り、商と余りの和を求め、全て足し合わせる。
- ② 奇数番目の文字に割り当てた数値は、そのまま全て足し合わせる。
- ③ ①と②の結果を足し合わせる。
- ④ ③で求めた総和が  $N$  で割り切れる場合は、検査文字付文字列に誤りがないと判定する。 $N$  で割り切れない場合は、検査文字付文字列に誤りがあると判定する。

### [検査文字付文字列の生成例]

表 1 及び検査文字の生成の手順を用いることによって、文字列 `ipa_` に対し、生成される検査文字は `f` である。

検査文字付文字列は、文字列の末尾に検査文字を追加し、`ipa_f` となる。

### [プログラムの仕様]

各関数の仕様を (1)～(4) に示す。ここで、配列の添字は 1 から始まるものとする。

- (1) 関数 `calcCheckCharacter` は、文字列及び文字列長を用いて生成した検査文字を返す。関数 `calcCheckCharacter` の引数及び返却値の仕様は、表 2 のとおりである。

表 2 関数 `calcCheckCharacter` の引数及び返却値の仕様

引数/返却値	データ型	入力/出力	説明
<code>input[]</code>	文字型	入力	文字列が格納されている 1 次元配列
<code>len</code>	整数型	入力	文字列の文字列長 (1 以上)
返却値	文字型	出力	生成した検査文字を返す。

関数 `calcCheckCharacter` は、関数 `getValue`、関数 `getChar` を使用する。

- (2) 関数 `validateCheckCharacter` は、検査文字付文字列を検証し、検証結果を返す。  
関数 `validateCheckCharacter` の引数及び返却値の仕様は、表 3 のとおりである。

表 3 関数 `validateCheckCharacter` の引数及び返却値の仕様

引数/返却値	データ型	入力/出力	説明
<code>input[]</code>	文字型	入力	検査文字付文字列が格納されている 1 次元配列
<code>len</code>	整数型	入力	検査文字付文字列の文字列長 (2 以上)
返却値	論理型	出力	検査文字付文字列に誤りがないと判定した場合は <code>true</code> 、誤りがあると判定した場合は <code>false</code> を返す。

関数 `validateCheckCharacter` は、関数 `getValue` を使用する。

- (3) 関数 `getValue` は、表 1 に従い、引数として与えられた文字に割り当てた数値を返す。
- (4) 関数 `getChar` は、表 1 に従い、引数として与えられた数値に対応する文字を返す。

設問1 プログラム中の  に入れる正しい答えを、解答群の中から選べ。ここで、a1 と a2 に入れる答えは、a に関する解答群の中から組合せとして正しいものを選ぶものとする。

[プログラム]

○文字型関数： calcCheckCharacter(文字型: input[], 整数型: len)

○整数型： N, sum, i, value, check\_value

○論理型： is\_even

・ N ← 30

・ sum ← 0

・ is\_even ← false

■ i: len, i > 0, -1

・ value ← getValue(input[i])

▲ is\_even =  a1

・ sum ← sum + value

・ sum ← sum + (value × 2) ÷ N + (value × 2) % N

▼

・ is\_even ← not is\_even

■

・ check\_value ←  b

・ return getChar(check\_value)

○論理型関数： validateCheckCharacter(文字型: input[], 整数型: len)

○整数型： N, sum, i, value

○論理型： is\_odd, ret\_value

・ N ← 30

・ sum ← 0

・ is\_odd ← true

・ ret\_value ← true

■ i: len, i > 0, -1

・ value ← getValue(input[i])

▲ is\_odd =  a2

・ sum ← sum + value

・ sum ← sum + (value × 2) ÷ N + (value × 2) % N

▼

・ is\_odd ← not is\_odd

■

▲  c

・ ret\_value ← false

▼

・ return ret\_value

aに関する解答群

	a1	a2
ア	false	false
イ	false	true
ウ	true	false
エ	true	true

bに関する解答群

ア  $N - \text{sum} \% N$

イ  $\text{sum} \% N$

ウ  $(N - \text{sum} \% N) \% N$

エ  $(\text{sum} - N) \% N$

cに関する解答群

ア  $\text{sum} \div N = 0$

イ  $\text{sum} \div N \neq 0$

ウ  $\text{sum} \% N = 0$

エ  $\text{sum} \% N \neq 0$

設問2 次の記述中の  に入れる正しい答えを、解答群の中から選べ。

本プログラムでは、検査文字付文字列の誤りが1文字であれば、誤りを検出できる。しかし、複数の文字に誤りがある場合には、誤りがないと判定されることがある。例えば、関数 `validateCheckCharacter` で表4に示す検査文字付文字列を検証した場合、誤りがないと判定されるケースは 。ここで、文字列 `ipa` に対し生成される検査文字は `f` である。

表4 検査文字付文字列

ケース	検査文字付文字列
1	<code>ipb</code>
2	<code>api</code>
3	<code>pia</code>
4	<code>api</code>

dに関する解答群

- ア 1と2と3と4である      イ 2である      ウ 2と3である  
エ 2と3と4である      オ 2と4である      カ ない

設問3 次の記述中の  に入れる正しい答えを、解答群の中から選べ。

本プログラムを文字列長が同じである複数の文字列に対して適用することを考える（図1参照）。

		列					
		1	2	3	4	5	6
行	1	i	p	a	□	□	
	2	t	e	s	t	s	
	3	m	a	k	e	□	
	4	i	t	.	□	□	
	5						

図1 作成中の検査文字付表

[考え方]

文字列長が  $n$  である  $m$  個の文字列について考える。文字列に対して、 $(m + 1)$  行  $(n + 1)$  列の表を用意する。以後、この表を検査文字付表という。

(1) 検査文字の生成

例えば、文字列長が5である4個の文字列  $ipa□□$ 、 $tests$ 、 $make□$ 、 $it.□□$  を、図1の太枠内のように、各文字列の先頭の位置を最左列に揃え、各文字列を上から順に格納して、表を作成する。この表の太枠内の各行各列をそれぞれ文字列とみなして検査文字を生成し、最右列と最下行に格納する。

この手順で作成した検査文字付表を図2に示す。作成した検査文字付表の5行5列目（網掛け部分）の検査文字は   $e$   である。

		列					
		1	2	3	4	5	6
行	1	i	p	a	□	□	f
	2	t	e	s	t	s	i
	3	m	a	k	e	□	n
	4	i	t	.	□	□	h
	5	r	a	v	b		

図2 完成した検査文字付表

(2) 検査文字付表の検証

(1)で作成した検査文字付表の、最下行を除く各行と最右列を除く各列を文字列とみなし、それぞれ関数 `validateCheckCharacter` で検証した結果、全て誤りがないと判定された場合には、検査文字付表に誤りがないと判定する。一つでも誤りがあると判定された場合は、検査文字付表に誤りがあると判定する。

eに関する解答群

ア j                  イ k                  ウ l                  エ m

設問4 次の記述中の  に入れる正しい答えを、解答群の中から選べ。

図2の1行目の検査文字付文字列を取り除いた、図3の検査文字付表について考える。表4のケース1～4の検査文字付文字列を順に、図3の1行目に格納して検証した場合、検査文字付表に誤りがないと判定されるケースは  f  。

		列					
		1	2	3	4	5	6
行	1						
	2	t	e	s	t	s	i
	3	m	a	k	e	□	n
	4	i	t	.	□	□	h
	5	r	a	v	b		

注記 5行5列目（網掛け部分）には、設問3の正しい答えが入っているものとする。

図3 1行目を取り除いた検査文字付表

fに関する解答群

- ア 1と2と3と4である      イ 2である      ウ 2と3である  
 エ 2と3と4である      オ 2と4である      カ ない