

情報V

学習指導要領 (3) - 知・技 - イ
 学習指導要領 (3) - 思・判・表 - イ
 学習内容 (3) - イ アルゴリズムとプログラム

ある整数 m と関数 f が与えられている。ただし、 f は、 $1 \leq x \leq m$ であるようなすべての整数 x に対して $f(x)$ が整数で $1 \leq f(x) \leq m$ を満たすとする。

$1 \leq a_1 \leq m$ であるような整数 a_1 に対して、数列 $\{a_1, a_2 = f(a_1), a_3 = f(a_2), \dots\}$ を考える。ある整数 $n \geq 1$ が存在して、 $a_1 = a_{n+1}$ となる時、部分数列 $\{a_1, \dots, a_n\}$ を「循環列」と呼ぶことにする。

例えば、 $m = 3, f(1) = 2, f(2) = 1, f(3) = 3$ であれば、数列 $\{1, 2\}$ 、数列 $\{2, 1\}$ 、数列 $\{3\}$ が循環列となる。しかし数列 $\{1, 2\}$ と数列 $\{2, 1\}$ は実質的に同じものであるから、ひとまとめにして考えたい。そこで、循環列に含まれる数の中で最小の数が先頭にあるものを「代表循環列」と呼ぶことにする。この例では代表循環列は数列 $\{1, 2\}$ と数列 $\{3\}$ である。

(ア) 空欄 ～ に当てはまるものを下の選択肢から選び、その番号を解答欄にマークしなさい。

$1 \leq a \leq m$ であるような1個の整数 a を与えて、 a から始まる（代表とは限らない）循環列があればそれを出力するアルゴリズムを次のように書いた。ただし、各行の左端の数字は、変更箇所を示すための行番号である。

- 1: 関数 f は与えられているものとする。
- 2: 変数 a の値を与えられた数とする。
- 3: 変数 b の値を $f(a)$ とする。
- 4: 変数 c の値を長さ1の数列 $\{a\}$ とする。
- 5: $a \neq b$ である間、処理 A を繰り返し実行する。
- 6: 処理 A の始め
- 7: c の値を、 c の末尾に b を付け加えた数列とする。
- 8: b の値を $f(b)$ とする。
- 9: 処理 A の終わり
- 10: c の値を出力する。

しかし、このアルゴリズムは正しくない。その理由は からである。

この欠点を解消するためには、6行目の次に『もし ならば「 a から始まる循環列はない」と出力してアルゴリズムを終了する。』という命令を追加すればよい。

次に、代表でない循環列を出力しないように変更したい。そのためには、上で追加した行の次にさらに『もし $\square^{(66)}$ ならば「 a から始まる代表循環列はない」と出力してアルゴリズムを終了する。』という命令を追加すればよい。

【 $\square^{(64)}$ の選択肢】

- (1) 循環列でないものが出力される場合がある
- (2) b の値が $1 \leq b \leq m$ の範囲から外れる場合がある
- (3) 処理 A が 1 回も実行されないことがある
- (4) 処理 A を無限に繰り返して、実行が終了しない場合がある

【 $\square^{(65)}$ ~ $\square^{(66)}$ の選択肢】

- | | |
|-------------------------|--------------------------|
| (1) $a < b$ | (2) $a > b$ |
| (3) $a = b$ | (4) $a \neq b$ |
| (5) $1 \leq b \leq m$ | (6) $b < 1$ または $m < b$ |
| (7) b が c の中に含まれている | (8) b が c の中に含まれていない |

(イ) 空欄 $\square^{(67)}$ ~ $\square^{(72)}$ に当てはまるものを下の選択肢から選び、その番号を解答欄にマークしなさい。

すべての代表循環列を出力するアルゴリズムを、上のアルゴリズムの a を 1 から m まで順に増やしていく形で次のように書いた。

- 1: 関数 f と定数 m は与えられているものとする。
- 2: 変数 a の値を最初は 1 とし、1 ずつ増やしながら m になるまで処理 B を繰り返す。
- 3: 処理 B の始め
- 4: 変数 b の値を $f(a)$ とする。
- 5: 変数 c の値を長さ 1 の数列 $\{a\}$ とする。
- 6: $a \neq b$ である間、処理 C を繰り返し実行する。
- 7: 処理 C の始め
- 8: もし (65) ならば (67)。
- 9: もし (66) ならば (68)。
- 10: c の値を、 c の末尾に b を付け加えた数列とする。
- 11: b の値を $f(b)$ とする。
- 12: 処理 C の終わり
- 13: c の値を出力する。
- 14: 処理 B の終わり

しかし、このアルゴリズムは無駄な実行をする場合がある。ある代表循環列が既に見つかって出力されていれば、その代表循環列に含まれる数についてはもはや処理 B を実行する必要はない。無駄な実行を減らすためには、次のように変更すればよい。(注意：処理 C についても無駄な実行を減らすことが可能であるが、ここでは処理 B についてのみ考えることにする)

- 3 行目の後に『もし (69) ならば (70)』という命令を追加する。
- (71) の後に『 c に含まれるすべての数 i について d_i の値を 1 とする。』という命令を追加する。
- (72) の後に『変数 d_1, \dots, d_m の値を 0 とする。』という命令を追加する。

【(67) ~ (70) の選択肢】

- (1) $d_a = 0$
- (2) $d_a = 1$
- (3) 処理 C の残りの部分は実行せず、「処理 C の始め」から次の繰り返しを実行する
- (4) 処理 C の繰り返しを中止し、「処理 C の終わり」の次の命令から実行する

- (5) 処理 B の残りの部分は実行せず、「処理 B の始め」から次の繰り返しを実行する
- (6) 処理 B の繰り返しを中止し、「処理 B の終わり」の次の命令から実行する
- (7) アルゴリズムを終了する

【 (71)、 (72) の選択肢】

- (1) 1 行目 (2) 5 行目 (3) 7 行目 (4) 12 行目