

〔Ⅲ〕 次の説明を読み、下の問に答えなさい。

管理用コンピュータ1台と、n台の端末コンピュータを使ってじゃんけんをするプログラムを考える。なお、ここでは管理用コンピュータのプログラムのみを考えることとし、端末コンピュータのプログラムは考えないこととする。また、プログラムは次の形式で表現されるものとする。なお、問題中のプログラムで、 $\langle \dots \rangle$ の部分には具体的な命令文が書かれていないが、適切な命令文が入っているものとする。

- 変数とは、数値や文字のデータを記憶するための仕組みで、次の表に示す2種類のものがあるとする。

変数	意味
a	小文字で表す変数には、一つのデータのみが代入される。
A	大文字で表す変数には、複数のデータが代入される。 代入された複数のデータは、 A_1, A_2, \dots というように、下付き数字によって区別される。下付き数字の部分は変数でもよい。

- 変数にデータを記憶することを「代入する」と呼ぶ。変数にデータを代入するには、「変数名←値」とする。
- 加減乗除算にはそれぞれ、 $+$ 、 $-$ 、 \times 、 \div を使う。
- プログラムは命令文の集まりとして記述され、使用できる命令文は次の表に示すもののみにする。

命令	意味
$a \leftarrow \text{受信}(pc)$	変数 pc が表す端末コンピュータからデータの一つを受信する。受信したデータは、変数 a に代入される。
送信($pc, \text{"abc"}$) 送信(pc, a)	変数 pc が表す端末コンピュータに "abc" という文字または変数 a の値を送信する。

条件分岐命令	意味
If 条件 Then 処理 1 Else 処理 2 EndIf	「条件」を満足したときは「処理 1」の部分を実行し、満足しなかったときは「処理 2」の部分を実行する。「処理 1」、「処理 2」の部分は複数行であってもよい。 「If 条件 Then」、「Else」、「EndIf」も1つの命令文と呼ぶ。

繰り返し命令	意味
For j ← 1 to n 処理 EndFor	変数 j = 1 の場合, 2 の場合, ..., n の場合と, j の値を変えて, 「処理」の部分を n 回実行する。 「処理」の部分は複数行であってもよい。 「For j ← 1 to n」, 「EndFor」も 1 つの命令文と呼ぶ。
Repeat 処理 While 条件	「処理」の部分を実行した後, 「条件」が判定される。 「処理」は, 「条件」を満足している限り, 繰り返し実行される。 「処理」の部分は複数行であってもよい。 「Repeat」, 「While 条件」も 1 つの命令文と呼ぶ。

- 「条件」は次のように表すものとする。なお, 数値計算の「×」が「+」に優先して先に計算されるように, 「And」が「Or」よりも先に計算されるものとする。

条件	意味
$a = b$, $a \neq b$	「=」は2つのものが等しいことを, 「≠」は等しくないことを表す。a, b は変数でもよい。
$a > b$, $a < b$	2つの数値の大小関係を表す。a, b は変数でもよい。
x And y	「かつ」を意味し, 条件 x, y を両方満足することを表す。たとえば, 「 $a = 0$ And $b = 0$ 」は, 「 $a = 0$ かつ $b = 0$ 」を意味する。
x Or y	「または」を意味し, 条件 x, y の少なくともどちらか一方を満足することを表す。たとえば, 「 $a = 0$ Or $b = 0$ 」は, 「 $a = 0$ または $b = 0$ 」を意味する。

- [例 1] 次の管理用コンピュータのプログラムでは、端末コンピュータ PC_1 から受信したデータを変数 x に代入し、 x の値が1のときは "1を受信しました" という文字を、1以外のときは "1以外を受信しました" という文字を PC_1 に送信する。

```

x ← 受信( $PC_1$ )
If x = 1 Then
    送信( $PC_1$ , "1を受信しました")
Else
    送信( $PC_1$ , "1以外を受信しました")
EndIf
    
```

- [例 2] 次のプログラムでは、3台の端末コンピュータからデータを1つずつ受信し、 PC_1 のデータを変数 A_1 に、 PC_2 のデータを A_2 に、 PC_3 のデータを A_3 に代入する。

```

For j ← 1 to 3
     $A_j$  ← 受信( $PC_j$ )
EndFor
    
```

ここでは、次の変数を用いてじゃんけんプログラムを作成することとした。

変数名	意味
ng, nc, np	"グー", "チョキ", "パー" を送信した端末コンピュータの数をそれぞれ代入する。
kekka	勝った手 ("グー", "チョキ" または "パー") を代入する。あいこの場合は "あいこ" を代入する。
$GCP_1 \sim GCP_n$	端末 $PC_1 \sim PC_n$ から送信されたデータを代入する。
$KM_1 \sim KM_n$	KM_1 には、 PC_1 が勝った場合は "勝ち" を、負けた場合は "負け" を代入する。 $KM_2 \sim KM_n$ も同様。

問1～問5 { 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

(問 1) じゃんけんプログラムを次の構成で作成することとした。空欄

～ に入る処理内容を下の①～④の中から1つずつ選び、その番号を解答欄にマークしなさい。

Repeat

If <あいこ> Then

<kekkaに"あいこ"を代入し、PC₁～PC_nに"あいこ"を送信>

Else

<kekkaに勝った手を代入>

EndIf

While kekka = "あいこ"

- ① <PC₁～PC_nに勝ち負けを送信>
- ② <ng, nc, npにそれぞれ該当する端末コンピュータの数を代入>
- ③ <端末コンピュータの送信データをGCP₁～GCP_nに代入する>
- ④ <KM₁～KM_nにPC₁～PC_nの勝ち負けを代入>

(問 2) <あいこ>の部分の条件は、「ng = n Or nc = n Or np = n Or

」のように書ける。このとき、空欄 に入る最も適切な条件を次の①～⑥の中から1つ選び、その番号を解答欄にマークしなさい。

- ① ng = 0 And nc = 0 And np = 0
- ② ng ≠ 0 And nc ≠ 0 And np ≠ 0
- ③ ng < n And nc < n And np < n
- ④ ng = 0 Or nc = 0 Or np = 0
- ⑤ ng ≠ 0 Or nc ≠ 0 Or np ≠ 0
- ⑥ ng < n Or nc < n Or np < n

(問 3) 前問の空欄 に入る条件は, 「And」も「Or」も使わないで記述することができる。その条件を解答欄に記入しなさい。

(問 4) <kekkaに勝った手を代入>の部分, 次のように作成した。空欄 ~ に入る条件の組み合わせとして最も適切なものを, 次の①~⑥の中から1つ選び, その番号を解答欄にマークしなさい。

```

If  Then
    kekka ← "グー"
EndIf
If  Then
    kekka ← "チョキ"
EndIf
If  Then
    kekka ← "パー"
EndIf
    
```

- ① サ : ng = 0 シ : nc = 0 ス : np = 0
- ② サ : nc = 0 シ : np = 0 ス : ng = 0
- ③ サ : np = 0 シ : ng = 0 ス : nc = 0
- ④ サ : ng ≠ 0 シ : nc ≠ 0 ス : np ≠ 0
- ⑤ サ : nc ≠ 0 シ : np ≠ 0 ス : ng ≠ 0
- ⑥ サ : np ≠ 0 シ : ng ≠ 0 ス : nc ≠ 0

- (問 5) 勝ちが1台になるまでじゃんけんを繰り返したい。勝ち残っている端末コンピュータを $W_1 \sim W_m$ に、その数を m という変数に代入することにしたところ、下のようなプログラムになった。なお、負けた端末コンピュータからはデータは送られてこないものとする。このとき、空欄 **セ** ～ **タ** に入る最も適切な命令文を解答群1の①～⑨の中から1つずつ選び、その番号を解答欄にマークしなさい。また、空欄 **チ** に入る最も適切な条件を解答群2の①～⑥の中から1つ選び、その番号を解答欄にマークしなさい。

```

For j ← 1 to n
    Wj ← PCj
EndFor
m ← n
Repeat
    <問1のプログラムと同様に、KM1 ～ KMm に、W1 ～ Wm が表
    す端末コンピュータが勝った場合は "勝ち" を、負けた場合は
    "負け" をそれぞれ代入する。あいこの場合はじゃんけんを繰
    り返す。>
    k ← 1
    For j ← 1 to m
        If KMj = "勝ち" Then
            セ
            ソ
        EndIf
        送信(Wj, KMj)
    EndFor
    タ
While チ
    送信(Wj, "優勝")

```

解答群 1

① $W_j \leftarrow W_k$

② $W_j \leftarrow PC_k$

③ $W_k \leftarrow W_j$

④ $W_k \leftarrow PC_k$

⑤ $k \leftarrow k - 1$

⑥ $k \leftarrow k + 1$

⑦ $m \leftarrow k - 1$

⑧ $m \leftarrow k + 1$

⑨ $m \leftarrow n$

解答群 2

① $m > 0$

② $m > 1$

③ $m = 0$

④ $m = 1$

⑤ $m > k$

⑥ $m = k$