

受験番号	志望学科・コース
	学科
	コース

[計・ソ専門 - 1]

問題 1

配列要素の並べ替えを行う次の手続き q を考える。

```

1: procedure q;
2:   procedure sort(left, right: integer);
3:     var p, i : integer;
4:     pivot : real;
5:   begin
6:     if left < right then
7:       begin
8:         pivot := a[left];
9:         p := left;
10:        for i := left + 1 to right do
11:          if a[i] < pivot then
12:            begin p := p + 1; swap(p, i); end;
13:          a[left] := a[p];
14:          a[p] := pivot;
15:          sort(left, p - 1); sort(p + 1, right);
16:        end;
17:      end;
18:   begin
19:     sort(1, m);
20:   end;

```

手続き q を呼び出す際には、整数型の定数 m が大域的に定義され、添字が 1 から m までの配列 a が宣言されているものとする。配列 a の配列要素のデータ型は実数型とする。添字 k ($1 \leq k \leq m$) の配列要素は $a[k]$ でアクセスできる。また手続き $\text{swap}(i, j)$ は $a[i]$ と $a[j]$ の内容を交換する。

手続き q は配列 a の添字 1 から m までの部分に格納されている m 個の実数型の値を昇順に並べ替えて、配列 a の添字 1 から m までの部分に格納しなおす。

手続き q に関する以下の問 (1) ~ (4) に答えよ。

- (1) 14 行目の処理が完了した時点での、 $a[\text{left}]$, $a[\text{left}+1]$, \dots , $a[\text{right}]$ のそれぞれと pivot との大小関係を答えよ。
- (2) n 個のデータを昇順に並べ替える場合を考える (プログラム中の定数 m の値は整数 n である)。手続き q が行う実数型の値の比較の回数は、一般に入力データの内容に依存する。 n 個のある入力データ d が与えられたとき、手続き q が行う実数型の値の比較の回数を $Q_n(d)$ とする。手続き q に入力データ d が与えられたとき、15 行目の手続き呼び出し $\text{sort}(\text{left}, p-1)$ と $\text{sort}(p+1, \text{right})$ で行われる実数型の値の比較の回数をそれぞれ R_L , R_R とする。
- (2-1) $Q_1(d)$ の値は入力データ d の内容に依存しない。 $Q_1(d)$ を求めよ。
- (2-2) $f(n)$ を n の関数とすると、 $n \geq 2$ のとき $Q_n(d) = f(n) + R_L + R_R$ の関係が成り立つ。 $f(n)$ を n の多項式の形で示せ。
- (3) 任意の d に対する $Q_n(d)$ の最大値を \bar{Q}_n とする。 $\bar{Q}_0 = 0$ と定義すると、問 (2-2) より $n \geq 2$ のとき $\bar{Q}_n \leq f(n) + \max_{0 \leq x < n} \{\bar{Q}_x + \bar{Q}_{n-1-x}\}$ を示すことができる。このとき

$$\forall x \in \{0, \dots, n-1\} \left(f(n) + \bar{Q}_x + \bar{Q}_{n-1-x} \leq \frac{n(n-1)}{2} \right)$$

を n に関する帰納法を用いて示すことにより、 $\bar{Q}_n \leq \frac{n(n-1)}{2}$ を示せ。

- (4) 手続き q が行う比較の回数が $\frac{n(n-1)}{2}$ となる入力データが存在する。問 (3) の不等式で等号が成立するのはどんな場合か説明せよ。その結果をふまえて、手続き q が行う比較の回数が最大となるのは入力データがどんな場合か説明せよ。

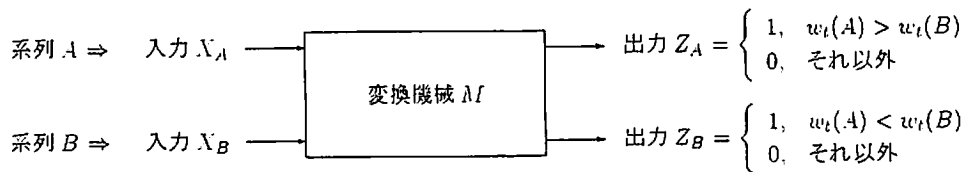
受験番号	志望学科・コース
	学科
	コース

問題 2

(1) 任意の論理関数は、2変数 AND 関数と (1変数) 否定関数から合成できることが知られている。このことを利用して、任意の論理関数は2変数 NOR 関数から合成できることを示せ。

(2) 2元系列 $S = s_1s_2\cdots$ (ここで、 $s_1, s_2, \dots \in \{0, 1\}$)、及び正整数 t に対して、 S の最初の t 個の成分 s_1, s_2, \dots, s_t に現れる記号 1 の個数を $w_t(S)$ と定義する。また、 $w_0(S) = 0$ と定義する。

下図に示す2入力2出力の変換機械 (各時点において、その時点までの入りに依存して出力が決まる機械) M を考える。入力記号集合、出力記号集合は共に $\{0, 1\}$ とする。



M の入力 X_A に与えられる2元系列を A 、入力 X_B に与えられる2元系列を B とする。入力はクロックに同期して与えられるとし、時刻 $t (t = 1, 2, \dots)$ には、系列 A, B それぞれの t 番目の記号が入力される。初期状態 (時刻 $t = 0$)、及び各時刻 $t (t = 1, 2, \dots)$ における出力は $w_t(A)$ と $w_t(B)$ の大小関係により下のように定義される。

大小関係	Z_A	Z_B
$w_t(A) > w_t(B)$	1	0
$w_t(A) < w_t(B)$	0	1
$w_t(A) = w_t(B)$	0	0

入力系列に制限をおかなければ、変換機械 M を状態数有限の順序機械として実現することはできない。そこで、以下の小問 (2-1)、(2-2) に述べる2種類の制約を考える。各問に答えよ。

(2-1) $|w_t(A) - w_t(B)| \leq 2$ を満たしている間は、 M と同じ動きをする同期式順序機械を考える。この条件が満たされなくなった場合、その時刻を含めそれ以降は、入力にかかわらず、 Z_A, Z_B とも 1 を出力し続けるものとする。この同期式順序機械 M' について、以下の (2-1-1)、(2-1-2) に答えよ。

(2-1-1) 同期式順序機械 M' の状態を以下のように定めるものとして、その状態遷移図を書け。

状態名	時刻 t に入力を読んで、その状態に遷移するための条件
S_{100}	$w_t(A) - w_t(B) = 1$ 、かつ任意の $t' (0 \leq t' < t)$ について、 $ w_{t'}(A) - w_{t'}(B) \leq 2$
S_{110}	$w_t(A) - w_t(B) = 2$ 、かつ任意の $t' (0 \leq t' < t)$ について、 $ w_{t'}(A) - w_{t'}(B) \leq 2$
S_{000}	$w_t(A) - w_t(B) = 0$ 、かつ任意の $t' (0 \leq t' < t)$ について、 $ w_{t'}(A) - w_{t'}(B) \leq 2$
S_{011}	$w_t(A) - w_t(B) = -2$ 、かつ任意の $t' (0 \leq t' < t)$ について、 $ w_{t'}(A) - w_{t'}(B) \leq 2$
S_{001}	$w_t(A) - w_t(B) = -1$ 、かつ任意の $t' (0 \leq t' < t)$ について、 $ w_{t'}(A) - w_{t'}(B) \leq 2$
S_{111}	ある $t' (0 \leq t' \leq t)$ について、 $ w_{t'}(A) - w_{t'}(B) > 2$

(2-1-2) Dフリップフロップを3個用いて、同期式順序機械 M' の回路を設計する。各フリップフロップの出力を変数 y_1, y_2, y_3 で表す。出力が $y_1 = p_1, y_2 = p_2, y_3 = p_3$ ($p_1, p_2, p_3 \in \{0, 1\}$) のとき、状態が $S_{p_1p_2p_3}$ となる状態割当のもとで、状態遷移関数と出力関数を表す積和形の論理式を示せ。最簡でなくてよい。

(2-2) n を2以上の正整数とする。系列 A, B の長さが共に n 以下であるとして、変換機械 M を実現する同期式順序機械を考える。このような順序機械のうちで状態数最小の機械における状態数はいくつか。個数を示し、その理由も簡単に述べよ。

受験番号	志望学科・コース
	学科
	コース

問題 3

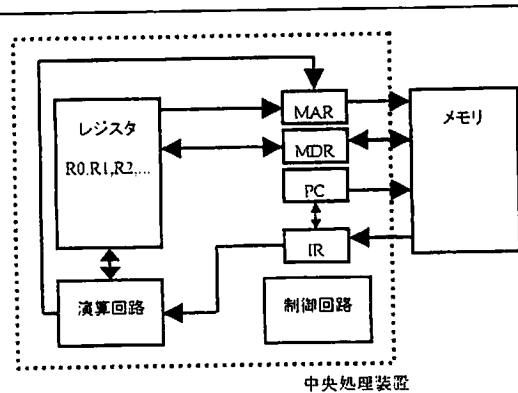
以下の説明文(四角の枠内)を読み、問(1)、(2)に答えよ。

右の図に示したような中央処理装置とメモリからなるシステムを考える。

中央処理装置は、複数のレジスタ(レジスタをそれぞれ R0, R1, R2, ... と書く)、次に実行すべき命令のおかれているメモリ上のアドレスを格納するプログラムカウンタ(PC)、実行中の命令を格納する命令レジスタ(IR)、メモリ上のアドレスを指定するメモリアドレスレジスタ(MAR)、メモリからのデータを格納するメモリデータレジスタ(MDR)、各種演算を行う演算回路、および各構成要素を制御するための制御回路を含む。矢印はデータのの流れの概略を表している。

このシステムでは、1つの命令の実行がはじまると、終了するまで次の命令の実行は開始されないとする。各命令では、命令の種類のほか、最大3つまでレジスタを指定することができる(このレジスタを dst, src1, src2 と書く)。また、命令によっては、命令中に書き込まれた値を直接参照する場合もある(このときの値を即値という)。

制御回路は、各構成要素へ適切な制御信号を供給することによって、1単位時間(以下1サイクルと呼ぶ)の間に、次の基本動作のうち1つのみを実行することができる。



※制御回路と各構成要素間のデータの流れは省略している

図 中央処理装置とメモリからなるシステム

- (ア) PC で指定されたアドレスから、1語(1命令分)取り出して IR へ書き込む(2語にわたる命令はないものとする)。
- (イ) IR 上の命令で指定されたレジスタ src1, src2 の内容に対して、指定された演算(加算、減算など)を演算回路によって行い、その結果をレジスタ dst に書き込む。
- (ウ) IR 上の命令で指定されたレジスタ src1 の内容を MAR へ転送する。
- (エ) MAR で指定されたアドレスに格納されているデータを、MDR へ書き込む。
- (オ) MDR の内容を IR 上の命令で指定されたレジスタ dst に書き込む。
- (カ) MAR で指定されたアドレスに、MDR の内容を書き込む。
- (キ) IR 上の命令で指定されたレジスタ dst の内容を MDR へ書き込む。
- (ク) IR 上の命令で指定されたレジスタ dst の内容が 0 でないならば、その命令の即値を PC へ書き込む。0 なら何もしない。
- (ケ) IR 上の命令で指定されたレジスタ src1 の内容と即値を加えた値を MAR へ書き込む。
- (コ) PC の内容を 1 増加させる(1命令分だけ増加させる)。

(1) 上記のシステムで種々の命令を実行することを考える。たとえば LOAD R0, (R1) は R1 の内容をアドレスとみて、そのアドレスからデータを読み出して R0 に書き込む命令であり(ここで、R0 と R1 がそれぞれ dst と src1 に対応している)、この命令を実行するためには、制御回路は上に示した基本動作を、(ア) → (ウ) → (エ) → (オ) → (コ) の順序で呼び出せばよい。

この例にならって、以下の各命令(a), (b), (c)を実行する際に制御回路が呼び出すべき基本動作の列を示せ。解は1通りとは限らないが、命令ごとに1通りずつ示せばよい。

- (a) 命令 SUB R0, R1, R2 (R1 の内容から R2 の内容を引いて R0 へ書き込む。ここで、R0, R1, R2 がそれぞれ dst, src1, src2 に対応している。)
- (b) 命令 STORE R0, 100(R1) (R1 の内容と 100 を加えた結果をアドレスとみて、そのアドレスへ R0 の内容を書き込む。ここで、R0 と R1 がそれぞれ dst と src1 に対応している。)
- (c) 命令 BNE R1, 100 (R1 の内容が 0 でないならば PC を 100 にする。ここで、R1 が dst に対応しており、即値は 100 である。)

(2) フィボナッチ数列 ($a_0=0, a_1=1, a_i = a_{i-1} + a_{i-2}$ ($i=2,3,4,\dots$))の a_{100} を計算するためのアセンブリプログラムを右に示す。プログラムの左端の数値はメモリ上のアドレスを表している。R4 に最終的な結果が得られる。また、このプログラムの開始アドレスは 200 であるとする。R0, R1, R2, R3, R4 の初期値をそれぞれ、99, 1, 0, 1, 0 とする。

- a) このプログラムの (i) および (ii) を埋めよ。ここで、ADD R0, R1, R2 は R1 と R2 の内容を加えて R0 に書き込む加算命令であり、MOV R0, R1 は R1 の内容を R0 に書き込む転送命令である。また、STOP はプログラムの実行を終了させる命令である。他の命令は問(1)で示したとおりである。
- b) 実行開始から実行終了まで(STOP 命令が終了するまで)の総サイクル数を示せ。ここで、ADD 命令の実行は SUB 命令と同じサイクル数を要し、MOV 命令は 3 サイクル、STOP 命令は 2 サイクルを要するものとする。他の命令については問(1)の解にもとづいた値を用いよ。

200	ADD R4, R2, R3
201	MOV R2, R3
202	(i)
203	(ii)
204	BNE R0, 200
205	STOP