平成１４年度　大阪大学基礎工学部編入学試験

[計算機科学・ソフトウェア科学コース専門科目] 試験問題

受験番号	志望学科・コース
	学科 コース

問題　１

下に示す手続き *insert* および *delete* は、ヒープと呼ばれるデータ構造への操作を行うものである。以下の問 (1) ～ (4) に答えよ。ただし、手続き中の以下の名前は、次のように宣言されているものとする。

- N : 大域的な整数型定数。ヒープに格納できるデータ数の上限を表す。
- num : 大域的な整数型変数。ヒープに格納されているデータ数を表す。
- A : 大域的なサイズ N の配列型変数。添字は 1 から N までとし、配列 A の各要素のデータ型は整数型とする。
- $swap(i, j)$: $A[i]$ のデータと $A[j]$ のデータを交換する手続き。

また、手続き中の i div 2 は i を 2 で割ったときの商の整数部分を表すものとする（例えば、7 div 2 = 3）。

(1) $N = 10$ とし、$num = 0$、各 i $(1 \leqq i \leqq N)$ について $A[i] = 0$ の状態を考える。この状態から、手続き *insert* を次の順に実行し終えたときの配列 A の内容を示せ。

$insert(10)$, $insert(5)$, $insert(8)$, $insert(7)$, $insert(2)$

(2) 上の問 (1) の解の状態から、手続き *delete* を 1 回実行し終えたときの配列 A の内容を示せ。

(3) $num = 0$、各 i $(1 \leqq i \leqq N)$ について $A[i] = 0$ の状態を考える。この状態から、異なる k 個（ただし、$k \leqq N$）のデータ $x_1, x_2, \ldots, x_k$ に対し、k 回の *insert* を次の順に行う。

$insert(x_1)$, $insert(x_2)$, $\ldots$, $insert(x_k)$

さらにその後、手続き *delete* を k 回実行する。この k 回の *delete* の実行を終えたとき、配列 A の内容がどうなっているか説明せよ。

(4) $num = 0$、各 i $(1 \leqq i \leqq N)$ について $A[i] = 0$ の状態を考える。この状態から、何回か *insert* と *delete* を行ったところ、$num = k$（ただし、$k \leqq N - 1$ とする）となった。このとき、さらに手続き $insert(x)$ を実行したとする。この *insert* で実行される手続き *swap* の実行回数が最も多いのは、配列 A と x のそれぞれの値がどのような関係の場合か述べよ。

```
procedure insert(x:integer);
  var
    i:integer;
  begin
    if num ≧ N then
      writeln('Heap is full')
    else begin
      num := num + 1;
      A[num] := x;
      i := num;
      while i > 1 do begin
        if A[i] < A[i div 2] then
          swap(i, i div 2);
        i := i div 2
      end
    end
  end;
```

```
procedure delete;
  var
    i, j:integer;
  begin
    if num = 0 then
      writeln('Heap is empty')
    else begin
      swap(1, num);
      num := num - 1;
      i := 1;
      while 2 * i ≦ num do begin
        if 2 * i = num then
          j := 2 * i
        else if A[2 * i] < A[2 * i + 1]
          then j := 2 * i
          else j := 2 * i + 1;
        if A[i] > A[j] then
          swap(i, j);
        i := j
      end
    end
  end;
```

平成１４年度　大阪大学基礎工学部編入学試験

[計算機科学・ソフトウェア科学コース専門科目] 試験問題

受験番号	志望学科・コース
	学科 コース

問題　２

(1) 組合せ回路 R は、組合せ回路 P と組合せ回路 Q を図 1 に示すように結線したものである。ここで、P への入力は x_1, x_2, x_3、Q への入力は x_1, x_2, x_3, y である。また、P と R が実現する論理関数 p と r のそれぞれの真理値表は表 1 および表 2 である。$*$ は対応する入力の値の組合せが、ドントケアであることを示している (つまり、これらの入力の値の組合せに対しては、出力が 0 でも 1 でも良い)。以下では組合せ回路 Q が実現するべき関数について考える。

(a) P が接続されているため、Q への入力 (x_1, x_2, x_3, y) には出現しない値の組合せが存在する。このような値の組をすべて列挙せよ。

(b) (a) で見たように、いくつかの値の組合せは Q の入力には出現しないため、これらの値の組合せは Q にとってドントケアであるとみなすことができる。組合せ回路 R が表 2 の関数を実現するように、組合せ回路 Q が実現するべき論理関数 q を定めたい。q の真理値表を示せ。

(c) (b) で求めた論理関数 q の最簡積和形表現を示せ。

表 1: 論理関数 p

x_1	x_2	x_3	p
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

表 2: 論理関数 r

x_1	x_2	x_3	r
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	*

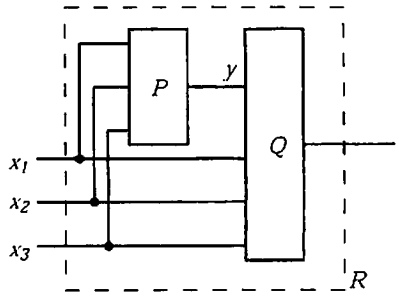

図 1: 組合せ回路 R

(2) 以下のような順序機械 M の論理設計を考える。M は入力系列中に 1101 または 1100 を検出すると、その時点で 1 を出力する機械である。M は検出するべき系列が重なり合って含まれている場合でも、系列を認識した時点で 1 を出力するものとし、たとえば、11011001101101 に対する出力は 00010010001001 となる。

(a) この順序機械 M の状態遷移図を示せ。

(b)D フリップフロップを 2 つ使って、順序機械 M を同期式順序回路として設計することを考える。2 つの D フリップフロップの出力には、それぞれ変数 y_0 と y_1 を対応させるものとする。入力を x、出力を z として、状態遷移関数と出力関数を論理式の形で示せ。どのような状態割当てを行ったかを明示せよ。

平成１４年度　大阪大学基礎工学部編入学試験

[計算機科学・ソフトウェア科学コース専門科目] 試験問題

受験番号	志望学科・コース
	学科 コース

問題　３

汎用レジスタ R_0, R_1, R_2, R_3 を持つCPUのアセンブリ言語の命令の表記と意味を以下に与える。

命令の表記	命令の意味（ここで $(0 \leqq i, j \leqq 3)$）
ADD R_i, R_j	$R_i + R_j$ の結果を R_i に入れる。
SUB R_i, R_j	$R_i - R_j$ の結果を R_i に入れる。
LOAD R_i, n	n の値を R_i に入れる。ここで n はレジスタ R_j もしくは数値 (10進記法) である。
JPZ R_i, label	R_i の値が0ならばlabelで指定された行から命令を実行する。それ以外は、直後の命令に進む。
JPNZ R_i, label	R_i の値が0でなければlabelで指定された行から命令を実行する。それ以外は、直後の命令に進む。
JP label	labelで指定された行から命令を実行する。
STOP	それ以降の実行を行わない。

以下は上記アセンブリ言語によるプログラムの記述例 (プログラムA) と意味である。

```
        LOAD   R0, 10
        LOAD   R1, 5
        LOAD   R2, 1
loop :  ADD    R0, R1
        SUB    R1, R2
        JPNZ   R1, loop
        STOP
```

プログラムA

i. レジスタ R_0 に10を入れる。
ii. レジスタ R_1 に5を入れる。
iii. レジスタ R_2 に1を入れる。
iv. レジスタ R_0 に R_1 を加算した結果を R_0 に入れる。
v. レジスタ R_1 から R_2 を減算した結果を R_1 に入れる。
vi. レジスタ R_1 の値が0でなければloopのラベルのある4行目 (すなわちiv) に進む。それ以外は、次 (vii) に進む。
vii. 停止する。

(1) プログラムAの実行直後のレジスタ R_0 の値を述べよ。

(2) プログラムBの空白部を埋めて、乗算プログラムを構成せよ。必ずしもすべての空白に何かの文字列が入るわけではない。何も入らない空白については「空白」と解答せよ。

なお、プログラムB及び(3)で用いるプログラムCにおいて次を仮定する。

- プログラム実行前、レジスタ R_0, R_1 に乗算の引数値が正整数として入っている $(R_0 \geqq R_1)$。
- プログラム実行後、乗算結果はレジスタ R_2 に入り、それ以外のレジスタ値についての制限はない。
- オーバーフロー、アンダーフロー等が生じない範囲で用いる。

```
(a)   LOAD   R2, 0
(b)   LOAD   R3, 1
(c)   ADD    (d)
(e)   (f)    (g)
(h)   (i)    (j), loop
      STOP
```

プログラムB

```
(a)   LOAD   R2, 0
(b)   (c)    R1, skip
(d)   ADD    R2, R0
(e)   SLA    (f)
      (g)    (h)
      JPNZ   R1, loop
      STOP
```

プログラムC

上記のCPUに下記の命令群を追加したCPU2を考える。

命令の表記	命令の意味（左を上位bitとする）
SLA R_i	R_i を左に算術シフトする。
SRA R_i	R_i を右に算術シフトする。
JPE R_i, label	R_i の値が偶数ならばlabelで指定された行から命令を実行する。それ以外は、直後の命令に進む。
JPO R_i, label	R_i の値が奇数ならばlabelで指定された行から命令を実行する。それ以外は、直後の命令に進む。

(3) CPU2に対して、算術シフトを用いた効率の良い乗算プログラムを先ほどと同様にプログラムCの空白部を埋めて完成させよ。