平成 31 年度(2018/8/4 実施)　入試　解答例

〇大問 1

(1)
答：a[j+1] < a[j]
※バブルソートであるため隣接要素を入れ替える。

(2)
もともとの配列 win,grade 間の同添字の対応を保ったままで配列 win に対する昇順にそれぞれの配列がソートされる。
なお時間計算量は 8 行目の比較処理が$\sum_{1}^{n-1}(n-k)=\frac{1}{2}n(n-1)$回行われるため、$O(n^2)$である。

(3)
配列 win に対して変数 lot の値を探索する二分探索を行っている。
戻り値としては配列内で値が見つかった場合はその配列のインデックス（添え字）を返し、見つからなかった場合には整数値-1 を返す。

(4)
(4-1)答：(ⅲ)
※クイックソートであるため(ⅰ)、(ⅳ)は考えられない。
(ⅱ)に関しては i>j(i=j+1)となった場合に{t～j-1},j,i,{i+1～w}のように再帰が起こるが j 番目 i 番目が含まれないこととなるため不可。
対して(ⅲ)は i>j(i=j+1)となった場合に{t～j-1,j},{i,i+1～w}のように再帰が起こるためこれが正しい。

(4-2)
$O(n\log n)$
※理由は書くべきか不明であるが余裕があれば書いたほうがいいと思われる。

(5)
(ア)：functionA(grade,win,0,n-1)
※修正前は win に対して昇順ソートを行っていたものを grade に対してソートする。これによって先頭が一等の win,grade のペアとなる。

(イ)：if((k=functionB(win,lot,1))!= -1)
※functionB の第三変数の配列の要素数を表す部分に 1 を入れると関数内では 1 番目の要素のみから探索を行うためO(1)となる。
なお、40 行目で grade(k)の出力を行っていることから変数 k への代入は必要である。

○大問２

(1)「数の表現」

(1-1)

符号絶対値表現：10001111

1 の補数表現：11110000

(1-2)

正の最大値：01111111(127)

負の最小値：10000000(-128)

(1-3)

(1-3-1)

43=　0010 1011

-5=　1111 1011

38= 10010 0110

この際最上位ビット（右から 9 ビット目の 1 は捨てられる）

(1-3-2)

減算を加算器のみの機構で行えることからコスト面で効率が良い。

(1-4)

(1-4-1)

$c = 1$のときオーバーフローとする。

(1-4-2)

$ab\bar{s} \vee \bar{a}\bar{b}s = 1$のときオーバーフローとする。

(2)「セマフォ、プロセス管理」

(2-1)

デッドロック状態：

複数のプロセスが互いの必要としている資源を持ったまま待機状態に移行してしまうこと

飢餓状態：

あるプロセスの必要としている資源をそれ以外のプロセスが使用し続けてしまい、半永久的にプロセスが待機し続けてしまうこと。

(2-2)
(ア)(ウ)(エ)(イ)　もしくは　(ア)(ウ)(イ)(エ)

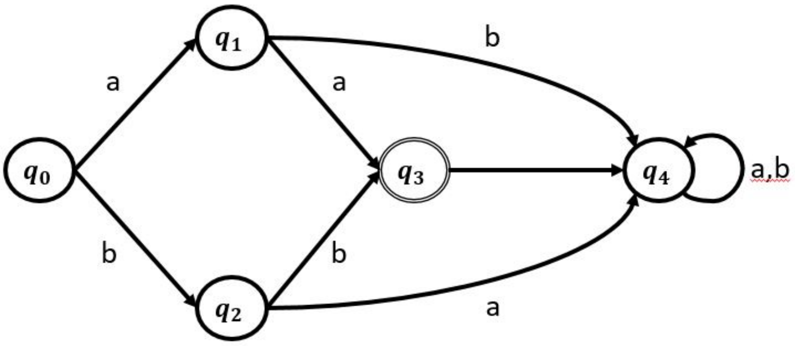
(2-3)
(2-3-1) a の初期値：a=1
・プロセス 1
```
P(a);
top = top – 1;
stack(top) = push_item;
V(a);
```
・プロセス 2
```
P(a);
pop_item = stack(top);
top = top + 1;
V(a);
```

(2-3-2)　初期値：b=0,c=1
・プロセス 3
```
while(true){
  P(b);
  m = m + 1;
  V(c);
}
```
・プロセス 4
```
while(true){
  P(c);
  print(m);
  V(b);
}
```
※セマフォ b,c はそれぞれインクリメント操作、表示操作に対応しており問題文の条件から先に表示操作を行う必要があるため初期値は c=1,b=0 とする。

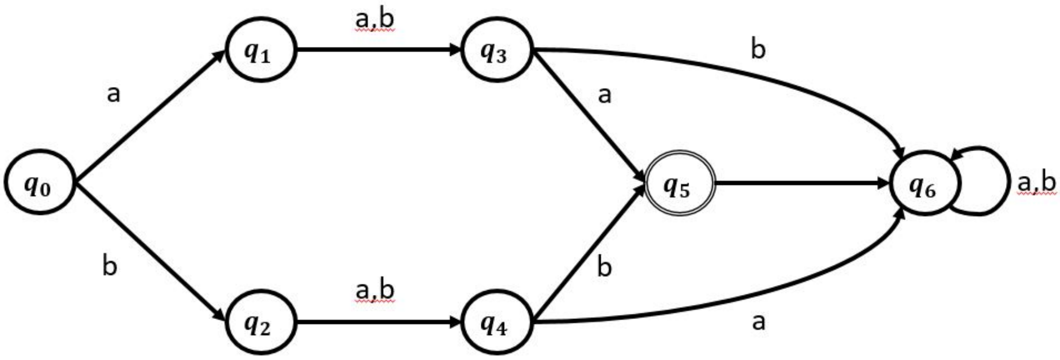
○大問 4

(1)「オートマトン、正則言語の反復補題」

(1-1)



(1-2)



(1-3)

題意の言語をL_0とする。これが正則言語だと仮定すると、反復補題が成り立つ。

反復補題の正定数 n に対して、w = $a^n b^{2n} a^n$とする。

このとき、$|xy| \leq n$ より $y = a^t (1 \leq t \leq n)$となるが、

$xy^0 z = a^{n-t} b^{2n} a^n \notin L_0$ となるため反復補題に反するため矛盾。

よってL_0は正則言語ではない。

(1-4)

(ε,1)/1 , (ε,0)/0 , (ε,Z)/Z　※文字長が偶数のものに相当

(a,1)/1 , (a,0)/0 , (b,1)/1 , (b,0)/0　※文字長が奇数のものに相当

(2)「文脈自由文法」

(2-1)

(ア)

$k > 1$の場合、$A \rightarrow aAbA \overset{*}{\rightarrow} w$、もしくは$A \rightarrow bAaA \overset{*}{\rightarrow} w$のように規則の適用が行われる。
この一回目の規則適用後の非終端記号 A をそれぞれA_1, A_2とおくと、それぞれ$k-1$回以下の適用で$A_1 \overset{*}{\rightarrow} v_1, A_2 \overset{*}{\rightarrow} v_2$のように適用が行われ終点記号列$v_1, v_2$が生成されると考えられる。
また仮定より$|v_1|_a = |v_1|_b, |v_2|_a = |v_2|_b$が成り立つことから、
$|w|_a = |v_1|_a + |v_2|_a + 1 = |v_1|_b + |v_2|_b + 1 = |w|_b$となり適用回数が k の時も成り立つといえる。

(2-2)

(イ)

$|w| = 0$のとき$|w|_a = |w|_b = 0$でありこれを満たすのは$w = \varepsilon$のみである。$A \rightarrow \varepsilon$のように規則を適用すると生成できるため$w \in L_1$となる。

(ウ)

aAbA

(エ)

bAaA

○大問 6

(1) 「MOSFET」

(1-1)

pMOS(丸つき)は g(入力)=L の時 ON となり電流を通し、逆に nMOS(丸なし)は g=H の時 ON となる。

よって左半分の回路は x,y のいずれかが L となれば H(V_{DD})を出力し、x,y どちらも H のときのみ L(GND)を出力する。

右半分の回路は左半分の回路からの入力が H の時 L を出力し L のとき H を出力する。

よって全体の回路としては x,y のいずれかが L となれば L を出力し、x,y どちらも H のときのみ H を出力する。すなわち AND 回路である。

※左半分が NAND 回路、右半分が NOT 回路と知っていればすぐに AND 回路とわかる。

(1-2)

@参考

http://www3.muroran-it.ac.jp/yt-lab/Lecture/KIT/EC/Doc/EC2-09.pdf

http://www.ai-l.jp/Res/LB5.Logic-delay-power.pdf　の（p7,8）

左半分の回路の出力は H→L となるため、放電現象である。

このとき微分方程式はキルヒホッフの法則より、

$$\mathrm{V}_1(t) + 2R_{mos}C_1\frac{\mathrm{dV_1(t)}}{\mathrm{dt}} = 0$$

となる。

この定常解は$V_{1s}(t) = 0$であり、過渡解は$V_{1t}(t) = \mathrm{A}\exp\left(-\frac{t}{2R_{mos}C_1}\right)$ $(A: 定数)$である（微分方程式省略）ため、$V_1(t) = V_{1s}(t) + V_{1t}(t) = \mathrm{Aexp}\left(-\frac{t}{2R_{mos}C_1}\right)$となる。また$V_1(+0) = V_1(-0) = V_{DD}$より$\mathrm{A} = V_{DD}$となるため、$V_1(t) = V_{DD}\mathrm{exp}\left(-\frac{t}{2R_{mos}C_1}\right)$となる。

ここで$V_1(t) = \frac{1}{2}V_{DD}$となるとき右側の回路のゲートソース間の閾値を超え、右側の回路の出力が変化し始めるため、そこまでの時間t_1を求めると

$V_1(t) = V_{DD}\exp\left(-\frac{t}{2R_{mos}C_1}\right) = \frac{1}{2}V_{DD}$より、$t_1 = 2R_{mos}C_1 log2$となる。

次に右側の回路は出力が L→H となるため充電現象である。

このとき微分方程式はキルヒホッフの法則より、

$$V_2(t) + R_{mos}C_2\frac{dV_2(t)}{dt} = V_{DD}$$

となる。

この定常解は$V_{2s}(t) = V_{DD}$であり、過渡解は$V_{2t}(t) = A\exp\left(-\frac{t}{R_{mos}C_2}\right))$である（微分方程式省略）ため、$V_2(t) = V_{2s}(t) + V_{2t}(t) = V_{DD} + A\exp\left(-\frac{t}{R_{mos}C_2}\right)$となる。また$V_2(+0) = V_2(-0) = 0$よりA$= V_{DD}$となるため、$V_2(t) = V_{DD}(1 - \exp\left(-\frac{t}{R_{mos}C_2}\right))$となる。

右半分の回路の入力が閾値を超えてから z からの出力が$V_2(t) = \frac{1}{2}V_{DD}$となるまでの時間$t_2$を求めると、

$V_2(t) = V_{DD}(1 - \exp\left(-\frac{t}{R_{mos}C_2}\right)) = \frac{1}{2}V_{DD}$より、$t_2 = R_{mos}C_2 log2$となる。

よって求める時間 T は、

$$T = t_1 + t_2 = R_{mos}(2C_1 + C_2)log2$$

(2)「論理シフト」

(2-1)

2 ビット左シフトなので(0,1,0,0)

(2-2)

p_3p_2 \ p_1p_0	00	01	11	10
00	0	0	1	1
01	0	0	0	0
11	1	1	1	1
10	1	1	1	1

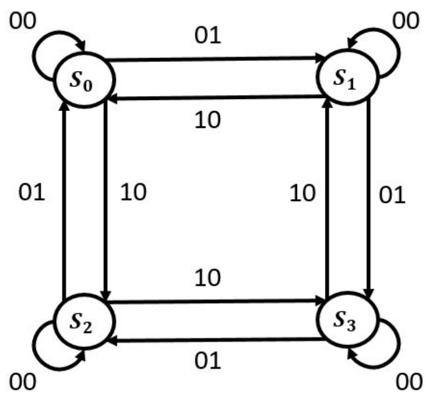
$$s_0 = p_3 \vee \overline{p_2}p_1$$

p_3p_2 \ p_1p_0	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

$$s_1 = p_3 \vee p_2$$

(3)「カウンタ」

(3-1)



(3-2)

現在の状態	(0,0)	(0,1)	(1,0)
S_0	00	01	11
S_1	01	10	00
S_2	10	11	01
S_3	11	00	10

(3-3)

Q_1Q_0 \ x_1x_0	00	01	11	10
00	0	1	X	1
01	1	0	X	0
11	1	0	X	0
10	0	1	X	1

$$D_0 = Q_0\overline{x_1x_0} \vee \overline{Q_0}x_0 \vee \overline{Q_0}x_1$$

Q_1Q_0 \ x_1x_0	00	01	11	10
00	0	0	X	1
01	0	1	X	0
11	1	0	X	1
10	1	1	X	0

$$D_1 = Q_1\overline{Q_0}\overline{x_1} \vee Q_1Q_0\overline{x_0} \vee \overline{Q_1}Q_0x_0 \vee \overline{Q_1Q_0}x_1$$