大阪大学大学院情報科学研究科

2009，H21年度　博士前期課程
入学試験「情報工学」解答例

【文責】藤田　敦 (a-fujita@ist.osaka-u.ac.jp)
(2010年度・B4)
【作成日】2010年8月23日

$\boxed{1}$ **（必須問題：アルゴリズムとプログラミング）**

解答

(1)

- 28 の格納　：　添字 **8** のセルに格納される
- 35 の格納　：　添字 **6** のセルに格納される

(2-1)

配列 `table[0]` から `table[19]` の値は以下の表のようになっている（index：添字，value：値）．

index	0	1	2	3	4	5	6	7	8	9
value	-1	-1	39	43	-1	45	-1	27	25	-1
index	10	11	12	13	14	15	16	17	18	19
value	-1	31	-1	-1	65	95	76	-1	-1	59

(2-2)

（ア） hash(d)　　**（イ）** h　　**（ウ）** next(h)　　**（エ）** -1

(2-3)

42 行目の for による繰り返し文で，関数 `store` の呼び出しにより配列 `data` の値をハッシュテーブルに格納している．繰り返しの制御変数が `i=8` であるときの関数 `store` の呼び出し終了時，つまり，`data[8]` までの値のハッシュテーブルへの格納を終えているとき，配列 `table` の値は以下の表のようになっている．

index	0	1	2	3	4	5	6	7	8	9
value	-1	-1	-1	39	-1	45	-1	27	-1	25
index	10	11	12	13	14	15	16	17	18	19
value	-1	31	-1	65	-1	95	76	-1	-1	59

ここで，繰り返しの制御変数が `i=9` であるときの関数 `store(table,data[9])` の呼び出しにおいて，**19 行目のハッシュ値との衝突確認のループが無限ループとなる**.

なぜならば， `hash(data[19]) = 3` であるが，ハッシュテーブルの $(3+4k) \bmod 4$ 番目のセルには既に値が格納されているので衝突が連続して繰り返されるからである．

従って，19 行目が無限ループとなるため，45 行目以降は実行されない．

(2-4)

　MAX と SKIP が**互いに素**であるように値を決めると (2-3) と同等の原因による無限ループは発生しない.

解説

ハッシュテーブルへの値の格納，探索の方法を問う基本的な問題である．教科書「アルゴリズム論」該当ページと講義「データ構造とアルゴリズム」の内容が理解できていれば十分に完解できるだろう．

教科書で紹介されているハッシュ法と異なるのは，データの衝突が発生した場合の処理である．教科書では衝突した場合にハッシュテーブルの次のセルが空きかどうかを調べているが，本問では，`k(SKIP)` 先のセルが空きかどうかを調べている．

(1)

ハッシュ法の定義より明らか．35 の格納の場合は，衝突が発生する．

(2-1)

ハッシュ法の定義，および，プログラムコードで実行をトレースしていけばわかる．

(2-2)

(ア) ハッシュテーブル検索のために，整数値 `d` のハッシュ値が必要になるので，関数 `hash` を呼び出す．

(イ) 27 行目の if 文の条件が,「`table[h] ==d`」であり，ハッシュテーブルの探索が成功した場合である．問題文の指示通りにハッシュ値をリターンする．

(ウ) 27 行目の if 文が実行されない場合なので,「`table[h] !=d`」であり，かつ，セルが `EMPTY` でない場合である．次のセルの探索のために，関数 `next` を呼び出す．

(エ) 探索の結果，見つからなかった場合である．問題文の指示通りに，-1 をリターンする．

(2-3)，(2-4)

理由，説明は解答記載の通り．

通常のハッシュ法では，ハッシュテーブルのサイズを素数にとることが多いので，`MAX` を素数にとれば，(2-3) のような無限ループは発生しない．しかし，ハッシュテーブルがすべて埋まっている場合はその例外である．

$\boxed{2}$ **（必須問題：論理回路）**

解答

(1-1)

・出力 S を表すカルノー図

S \ a_1a_0 ＼ b_1b_0	00	01	11	10
00	1	0	0	0
01	1	1	0	0
11	1	1	1	1
10	1	1	0	1

カルノー図内，丸囲みの主項はすべて必須項であるので，最小積和形は以下になる．

$$S = \boxed{a_1a_0} + \boxed{a_1\bar{b}_1} + \boxed{a_1\bar{b}_0} + \boxed{a_0\bar{b}_1} + \boxed{\bar{b}_1\bar{b}_0}$$

(1-2)

・出力 S を表すカルノー図（d は don’t care）

S \ S_1S_0 ＼ E_1E_0	00	01	11	10
00	0	d	d	d
01	0	0	d	d
11	1	1	1	1
10	1	d	d	0

・出力 E を表すカルノー図（d は don’t care）

E \ S_1S_0 ＼ E_1E_0	00	01	11	10
00	0	d	d	d
01	0	0	d	d
11	0	0	1	0
10	0	d	d	0

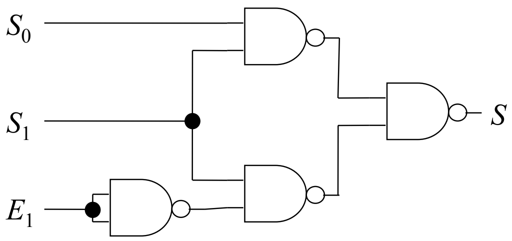
カルノー図内，S の丸囲みの主項はすべて必須項でないので，最小積和形は 2 通りある．一方，E の丸囲みの主項はすべて必須項あるので，最小積和形は一意に定まる．それぞれ，以下のようになる．

$$S = \boxed{S_1S_0} + \boxed{S_1\bar{E}_1} = \boxed{S_1\bar{E}_1} + \boxed{S_0E_1}$$

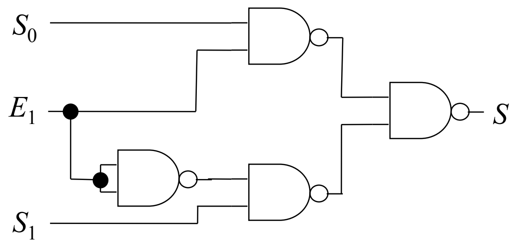
$$E = \boxed{E_1E_0}$$

(1-3)

・(1-2) で $S = S_1 S_0 + S_1 \bar{E}_1$ の場合

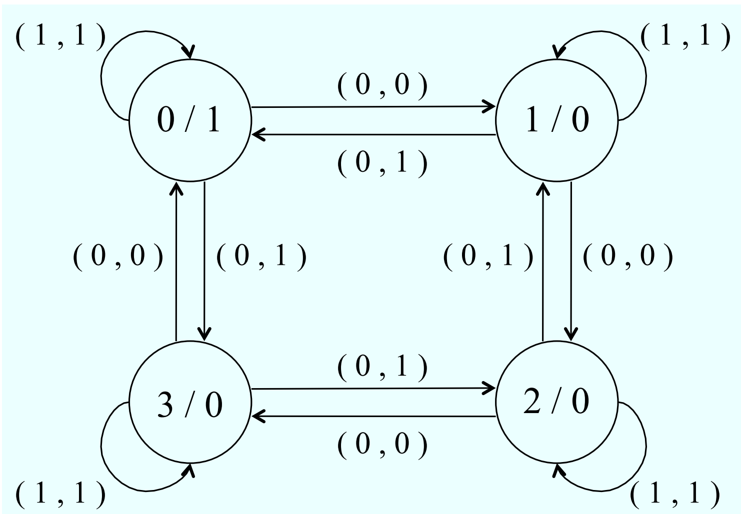


・(1-2) で $S = S_1 \bar{E}_1 + S_0 E_1$ の場合



(2-1)

・カウンタの状態遷移図



(2-2)

・カウンタの状態遷移表，および，出力表

（状態遷移図の状態値（0，1，2，3）の 2 進数を状態変数 $Q_1 Q_0$ に割り当てている）

現状態 $(Q_1 Q_0)$	次状態 $(Q_1^+ Q_0^+)$ (x_1, x_0) (0,0)	(0,1)	(1,1)	出力 (f)
00	01	11	00	1
01	10	00	01	0
10	11	01	10	0
11	00	10	11	0

(2-3)

D フリップフロップを用いるので，状態変数 $Q_1^+Q_0^+$ の値をそのまま D フリップフロップの入力 D_1D_0 にできる．

・出力 D_1 を表すカルノー図
(d は don't care)

D_1		x_1x_0 00	01	11	10
Q_1Q_0	00	0	1	0	d
	01	1	0	0	d
	11	0	1	1	d
	10	1	0	1	d

・出力 D_0 を表すカルノー図
(d は don't care)

D_0		x_1x_0 00	01	11	10
Q_1Q_0	00	1	1	0	d
	01	0	0	1	d
	11	0	0	1	d
	10	1	1	0	d

カルノー図内，丸囲みの主項はすべて必須項であるので，最小積和形はそれぞれ以下になる．

$D_1 = \boxed{Q_1x_1} + \boxed{Q_1Q_0x_0} + \boxed{\bar{Q_1}Q_0\bar{x_0}} + \boxed{Q_1\bar{Q_0}\bar{x_0}} + \boxed{\bar{Q_1}\bar{Q_0}\bar{x_1}x_0}$

$D_0 = \boxed{\bar{Q_0}\bar{x_1}} + \boxed{Q_0x_1}$

出力信号 f は $(Q_1, Q_0) = (0, 0)$ のときのみに 1 を出力すればよいので，

$f = \bar{Q_1}\bar{Q_0}$

解説

論理回路の問題は，例年，組み合わせ回路，順序回路の問題がそれぞれ小問として 1 題ずつ出題されている．講義「情報数学基礎」の東野先生担当分の資料内容や院試過去問で練習をしておけば対応できる．ただ，2010 年度，2011 年度は若干難化し，傾向が変化している．作業量が多いこととカルノー図の書き間違いをしやすいので，注意して解きたい．

(1-1)

問題のリード文にあるように，$A(a_1a_0) \geq B(b_1b_0)$ であれば，$S = 1$，$A(a_1a_0) < B(b_1b_0)$ であれば，$S = 0$ である．この関係をカルノー図のセルに書き込んで，最小積和形を求めればよい．

(1-2)

(1-1) と比べると少しややこしい．時間はかかるが，$A(a_3a_2a_1a_0), B(b_3b_2b_1b_0)$ の値を具体的に定めて S_1, S_0, E_1, E_0 を求めて考えると正確にカルノー図を作成できる．問題文にあるように $S_i = 0$ であれば $E_i = 0$ であるため，$S_i = 0$，$E_i = 1$ である場合は don't care である．

(1-3)

最小積和形で与えられる論理関数は NAND ゲートのみで構成できる（詳細は，情報数学基礎　東野先生担当分　講義資料　4.3 節参照のこと）．積和形で考えているため，1 段目は AND 回路で，2 段目は OR 回路で実現できるが，この両方を NAND 回路で置き換えることができる．

本問では，入力信号の否定は利用できないので，NAND ゲートの両方の入力に同一信号（E_1）を入力することで否定（$\bar{E_1}$）を実現する．

(2-1)

問題のリード文にあるように入力 x_1x_0 の値によってカウンタで保持する値が変化する．カウンタ値を状態名とする状態を設けその遷移を示せばよい．Moore 型順序回路で実現するので，状態に対して出力が決まることに注意．

(2-2)

状態遷移図の状態値（0，1，2，3）の 2 進数を状態変数 Q_1Q_0 に割り当てている．状態変数 Q_1Q_0 の状態遷移を (2-1) の状態遷移図を元にして表にする．尚，状態変数の割り当ての方法を変えると，解答の表とは異なり，(2-3) の最小積和形も変化する．

(2-3)

(2-2) の状態遷移表の次状態 $(Q_1^+Q_0^+)$ の値をもとに，D フリップフロップの入力を表すカルノー

図を作成する．この作業は必須であるので十分練習しておきたい．尚,（作業量の関係か）2011 年度までの試験では D フリップフロップしか出題されていないため，次の遷移の状態変数をそのままフリップフロップの入力にできるが，SR・JK フリップフロップはこのようにはできない．励起表を利用してフリップフロップの入力を決定する必要がある．一応，練習しておいたほうがよいだろう．

3 （必須問題：計算機システムとシステムプログラミング）

解答

(1-1)

(a)	（ウ）	時間的局所性	**(b)**	（ア）	空間的局所性
(c)	（カ）	完全連想マッピング	**(d)**	（オ）	直接マッピング
(e)	（コ）	低	**(f)**	（カ）	完全連想マッピング
(g)	（イ）	セット連想（群連想）マッピング			

(1-2)

- 実装の容易さ

  FIFO はキャッシュに空きブロック枠がなく該当ブロックが存在しないときに，最も古くにキャッシュされたブロックとの置き換えを行う方式であり，キューとして実現することができるため，実装は容易である．

  一方，LRU は最も長い間参照されなかったブロックとの置き換えを行う方式であり，どのブロックがいつ参照されたかの情報をもとに置き換えるブロックを決定する必要があるため，実装は FIFO に比べて容易でない．

- ブロック枠の数とヒット率の関係

  FIFO はその方式の特徴から，ブロック枠の数によらず，参照の局所性により参照頻度の高いブロックが置き換えられる可能性が高くなる．従って，ブロック枠の数を変化させてもヒット率の変化はそれほどない．

  一方，LRU はその方式の特徴から，ブロック枠の数が大きくなくとも参照頻度の高いブロックはキャッシュされ続けているためヒット率は高い．

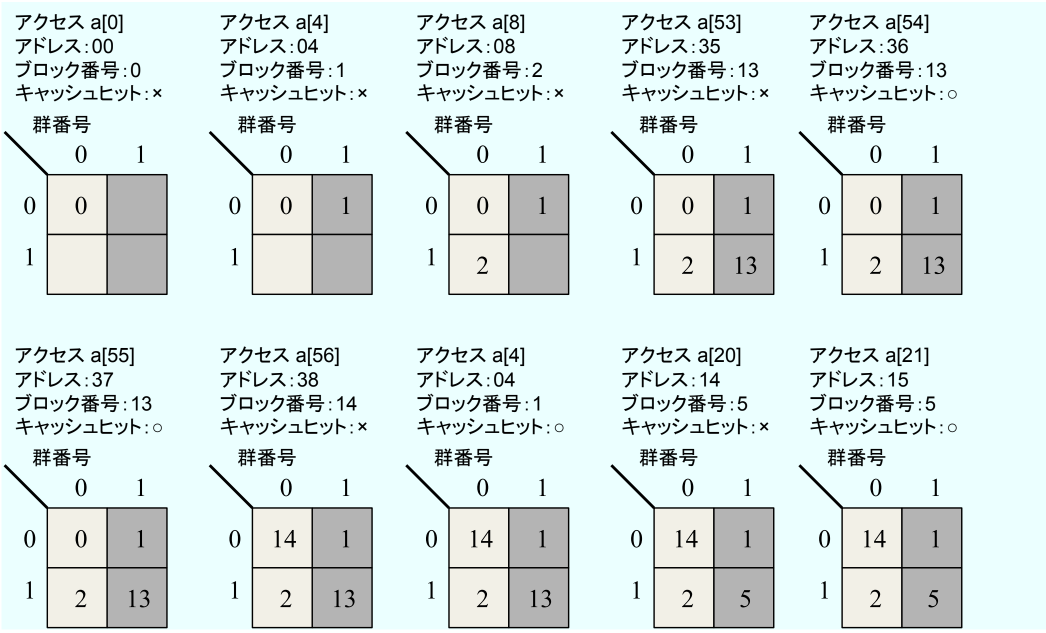
(1-3-1)

5 ビット

(1-3-2)

各アクセスによるキャッシュの中身の変化を以下に示す．

10 回のアクセスでキャッシュヒットは 4 回であるので，ヒット率は **40 %**である．



(2-1)

(a) （カ） 実行可能状態

(b) （キ） ディスパッチ

(c) （シ） 実行状態

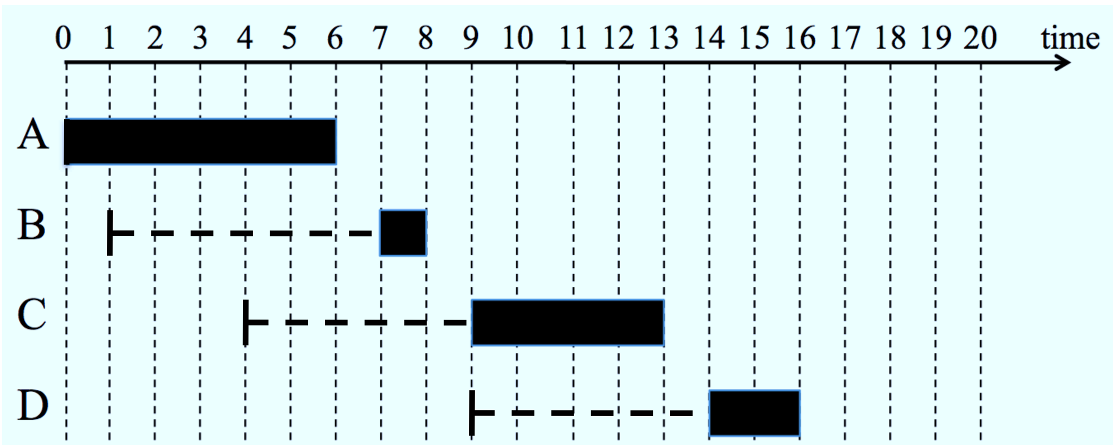
(d) （ウ） 待ち状態

(e) （ソ） プリエンプション

(f) （イ） コンテクストスイッチ

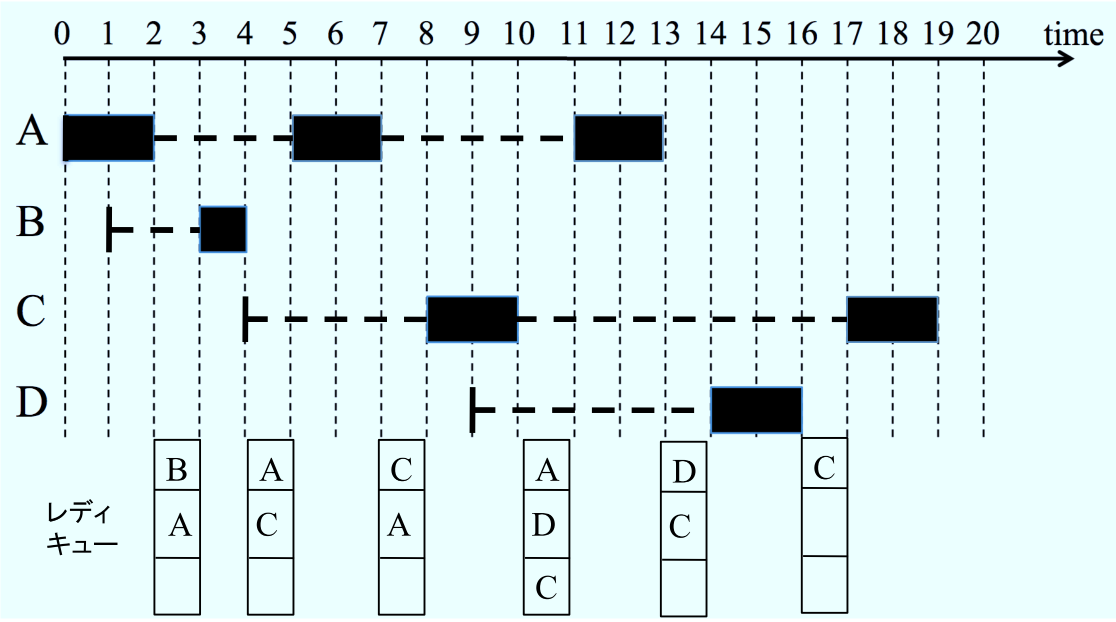
(2-2)

- FIFO 方式



  – 平均ターンアラウンドタイム = (6+7+9+7)/4 = **7.25**
  – 平均応答時間 = (0+6+5+5)/4 = **4.0**

- RR 方式



  - 平均ターンアラウンドタイム = (13+3+15+7)/4 = **9.5**
  - 平均応答時間 = (0+2+4+5)/4 = **2.75**

解説

計算機システムの問題は，講義「計算機アーキテクチャ」,「オペレーションシステム」からの出題である．出題される話題は多岐に分かっているが講義内容からではあるので，講義資料や教科書を 1 通り読んでおけばよい．例年，作業量が多いので効率よく解きたい．

(1-1)

教科書「計算機アーキテクチャ」3.3.1 節　キャッシュメモリの構成，および，講義資料ノート 4 を参照のこと．

(1-2)

置き換えアルゴリズムとしての FIFO，LRU の特徴が理解できていれば解答できる．教科書「計算機アーキテクチャ」3.3.1 節　キャッシュメモリの構成，教科書「オペレーションシステム」3.9 節　ページ置き換えアルゴリズムの該当範囲，および，講義資料を参照のこと．

(1-3-1)

群連想マッピング方式を用いた場合の主記憶アドレスの構成は以下のようになる．

郡内ブロック識別番号 = タグ	群番号	ブロック内 アドレス

問題のリード文により，主記憶アドレスは 8 ビット，1 ブロックの大きさは $4(=2^2)$ 語であるのでブロック内アドレスは 2 ビット，群数は $2(=2^1)$ であるので群番号は 1 ビット．よって，郡内ブロック識別番号は 5 ビットである．

(1-3-2)

群連想マッピングの方式通りにキャッシュメモリにマッピングを行う．群が 2 群あるので，番号が偶数のブロックは群番号 0，番号が奇数のブロックは群番号 1 のブロックにキャッシュされる．ブロックの置き換えが生じる場合には，LRU アルゴリズムを利用しているので最も長い間参照されていないブロックを置き換えることに注意する．

(2-1)

プロセスの状態に関する出題である．教科書「オペレーションシステム」2.1 節　プロセスとは　の該当範囲，および，講義資料を参照のこと．

(2-2)

スケジューリングアルゴリズムに関して，プロセスがプロセッサで処理される様子の図示を行う問題であり，図示と，平均ターンアラウンドタイムと平均応答時間を計算する．

図示に関しては，講義「オペレーションシステム」の演習問題で扱われているはずなので難しくはないはずである．しかし，本問では問題文に留意する点として挙げられているように，**次のプロセスのディスパッチ後，処理を開始するまで処理時間が1ある点**に注意する（過去問の同問題ではこの処理時間を0としている試験年もある）．

本問の結果から，平均ターンアラウンドタイムの観点ではプロセス切り替えのオーバーヘッドが少ないFIFO方式が有利で，平均応答時間では最大でもタイムスライス時間ごとに，プリエンプションにより次々に処理するプロセスを切り替えるRR方式が有利であると考察できる．

$\boxed{8}$ **（選択問題：情報論理学）**

(1-1)

(a)

$C : a = 1 \qquad\qquad P : \{p(0) = 0,\, p(1) = 0,\, p(2) = 0\}$

(b)

$F : \{f(0) = 1,\, f(1) = 1,\, f(2) = 1\} \qquad\qquad P : \{p(0) = 1,\, p(1) = 1,\, p(2) = 1\}$

(c)

$C : \{a = 1,\, b = 1\} \qquad\qquad P : \{p(0) = 0,\, p(1) = 0,\, p(2) = 0\}$

(1-2)

(a)

$P : \{p(x) = 1,\, q(x) = 1\}$

(b)

変形すると，$\exists x(\neg\, p(x) \land \neg\, q(x))$.

$P : \{p(x) = 0,\, q(x) = 0\}$

(c)

変形すると，$\forall x(p(x) \land q(x))$.

$P : p(x) = 1,\, q(x) = 1$

(1-3)

$$
P : \{\ \begin{array}{lll} p(x) & = 1 & (x : \text{奇数}) \\ & = 0 & (x : \text{偶数}) \end{array} \qquad \begin{array}{ll} q(x) = 0 & (x : \text{奇数}) \\ \ 1 & (x : \text{偶数})\} \end{array}
$$

という解釈を考えると，常に $p(x)$ か $q(x)$ のどちらかが 1 となるので $\forall x(p(x) \land q(x))$ は 1．だが、$\forall x\, p(x)$ は 0 であり，$\forall x\, q(x)$ も 0 であるので, $\forall x\, p(x) \land \forall x\, q(x)$ は 0 となる．

(2-1)

$$
\begin{aligned}
\neg A &= \neg\,(\forall y \forall x\ \ p(x,y) \wedge \exists y \forall x(\neg\, p(x,y) \rightarrow q(x))) \rightarrow \forall z\ \ q(z)) \\
&= (\forall y \forall x\ \ p(x,y) \wedge \exists y \forall x(\neg\, p(x,y) \vee q(x))) \wedge \neg\, \forall z\ \ q(z) \\
&= \forall y \forall x\ \ p(x,y) \wedge \exists y \forall x(\neg\, p(x,y) \vee q(x)) \wedge \exists z\ \ \neg\, q(z) \\
&= \exists w \exists z \forall y \forall x(p(x,y) \wedge (\neg\, p(x,z) \vee q(x)) \wedge \neg\, q(w))
\end{aligned}
$$

(2-2)

スコーレム関数 a, b を用いると，　$p(x,y) \wedge (\neg\, p(x,b) \vee q(x)) \wedge \neg\, q(a)$.

(2-3)

1. $p(x,y)$	2. $\neg\, p(x,b) \vee q(x)$
3. $q(a)$	4. $p(a,b) \vee q(a)$ $\cdots$ 2.で x に a を代入
5. $p(a,b)$ $\cdots$ 3.と 4.の導出節	6. $p(a,b)$ $\cdots$ 1.で x に a, y に b を代入
7. 0 $\cdots$ 4.と 6.の導出節	

導出節として空節 0 が導かれたので A' は充足不能.

$\boxed{9}$　**（選択問題：計算理論）**

解答

(1-1)

10，010，0010，1110

(1-2)

・M_2 の各状態に対する ϵ-閉包

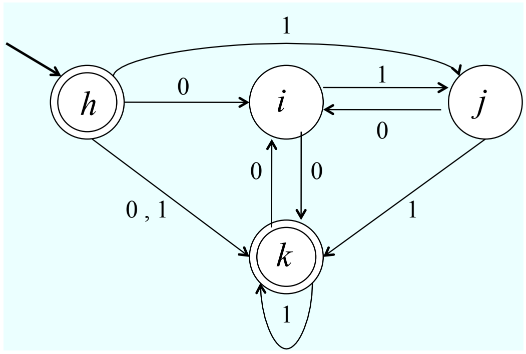
$\epsilon(h) = \{h, i, j, k\}$

$\epsilon(i) = \{i\}$

$\epsilon(j) = \{j, k\}$

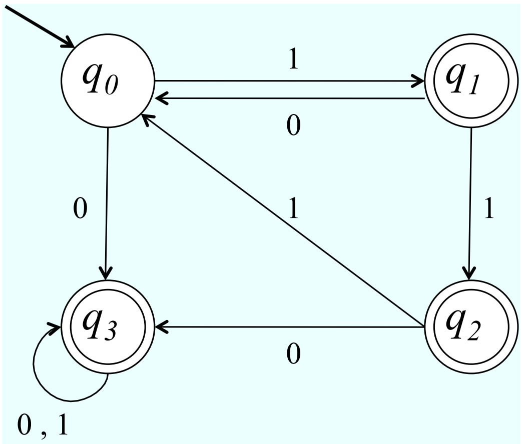
$\epsilon(k) = \{k\}$

・M_2' の状態遷移図



(1-3)

・M_3' の状態遷移図



(2-1)

・最左導出過程

A　→　if e then A else A

　→　if e then s else A

　→　if e then s else if e then A

　→　if e then s else if e then s

・最右導出過程

A　→　if e then A else A

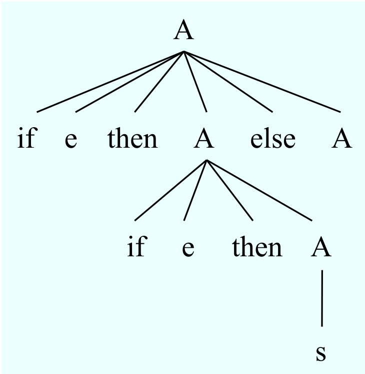
　→　if e then A else if e then A

　→　if e then A else if e then s

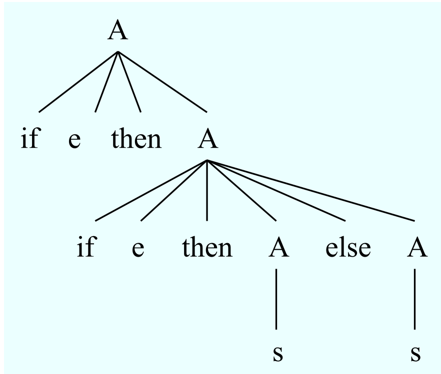
　→　if e then s else if e then s

(2-2)

・導出木 (A)



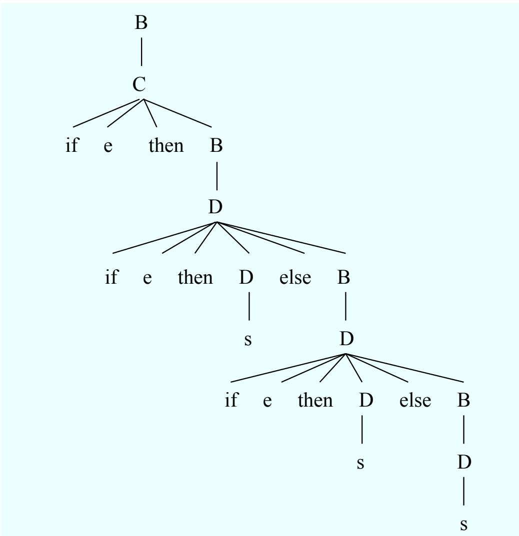
・導出木 (B)



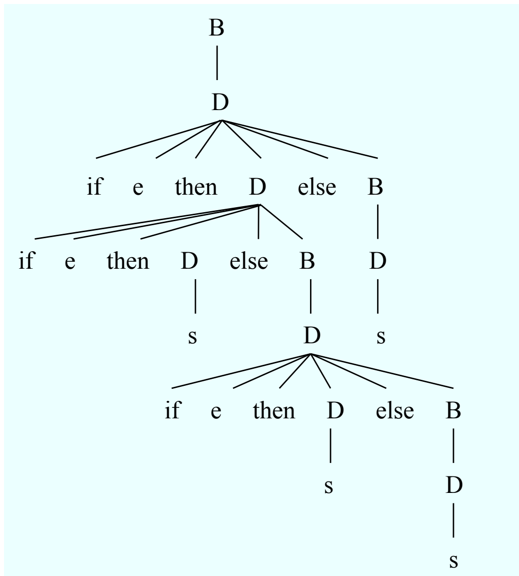
(2-3)

文「if e then if e then s else if e then s else s」に対する導出木が以下の (A)，(B) の 2 通りに書ける．よって，複数の導出過程が存在するので文法 G_2 は曖昧である．

・導出木 (A)



・導出木 (B)



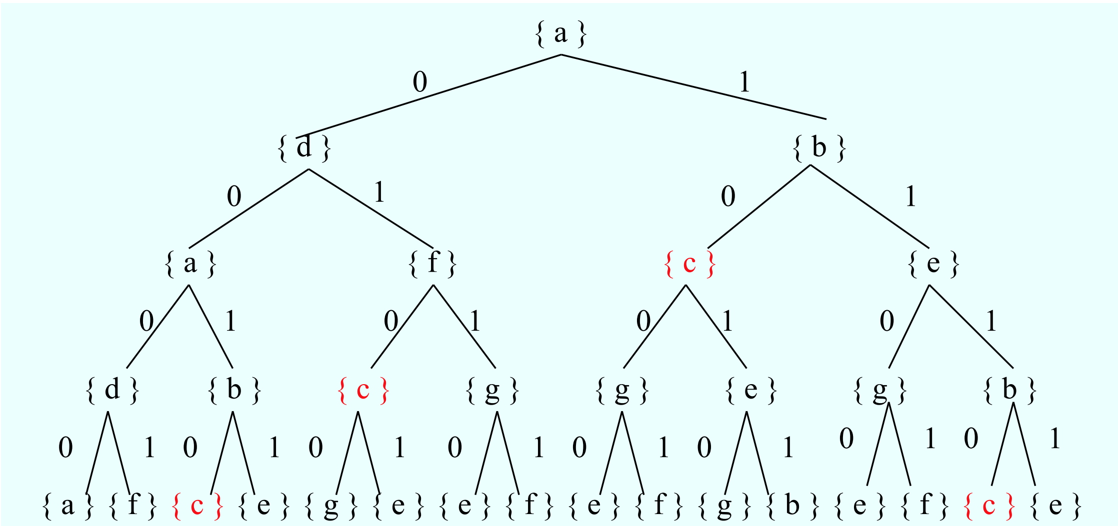
解説

計算理論の問題は，例年，出題傾向，難易度，作業量ともにそれほど変化がないので，一番対策がしやすいものと思われる（ややこしい証明などの出題は希である）．基本手を動かすような問題が中心である．院試過去問，講義の中間試験，期末試験程度の問題で練習しておけば十分対応できる．

(1-1)

結構，遷移がややこしいので，漏れのないように調べ上げること．ちなみに，試験年によって「長さ N の語」か「長さ N **以下** の語」引っかけてくるので注意.

以下のような，状態木（ある状態から次の入力記号を読んだときにどの状態に遷移するかを表現した木）を利用できれば時間はかかるが漏れなく調べ上げることができるかもしれない.



(1-2)

問題文中にあるように，ある状態 x に対する ϵ 閉包は，x そのものと ϵ 動作のみで到達できる状態集合である．M_2 の状態遷移図から ϵ 閉包を決定する.

ϵ 動作のない非決定性有限オートマトン M_2' は，状態 x に対する ϵ 閉包の集合を元にして，集合要素の状態から 0,1 を読んで遷移できる状態への状態 x からの遷移を新たな遷移として追加していけばよい.

(1-3)

有限オートマトン M_3 は，非決定性のオートマトンであるのでまずは決定性オートマトンに変換する．次に，変換した決定性オートマトンでは状態数最小であることは保証できないので，状態数最小の決定性オートマトンに変換する．前者の変換は「部分集合構成法」により，後者の変換は「k 等価性を利用しての分割（オートマトンの簡約化）」により行う．各手順の詳細は，教

科書「オートマトンと形式言語」，および，講義資料を参照のこと．

以下に手順を簡単に示す．

・STEP1：部分集合構成法（初期状態 $\{a\}$，受理状態 $\{b, c, d, e, f, g\}$）

仮状態名		0	1
a	$\{l\}$	$\{mn\}$	$\{m\}$
b	$\{m\}$	$\{l\}$	$\{n\}$
c	$\{n\}$	$\{mn\}$	$\{l\}$
d	$\{mn\}$	$\{lmn\}$	$\{ln\}$
e	$\{lmn\}$	$\{lmn\}$	$\{lmn\}$
f	$\{ln\}$	$\{mn\}$	$\{lm\}$
g	$\{lm\}$	$\{lmn\}$	$\{mn\}$

	0	1
a	d	b
b	a	c
c	d	a
d	e	f
e	e	e
f	d	g
g	e	d

・STEP2：オートマトンの簡約化（$\Pi_0 \to \Pi_1$）

0	1					
a	b	c	d	e	f	g
11	01	10	11	11	11	11

0	1	2	3			
a	b	c	d	e	f	g
31	02	30	33	33	33	33

(2-1)

最左導出，最右導出の定義通りに導出過程を示していけばよい．

(2-2)

生成規則を適用して 2 通りの導出木を示す．生成規則を選んでいく作業は難しくない．

(2-3)

文法 G_2 が曖昧であることを示したいので，ある文の生成に対して 2 通りの導出木が書けることを示す．少しややこしいが実際に手を動かして探していくしかない．

$\boxed{11}$ **（選択問題：ネットワーク）**

解答

(1)

I)　※下記より 2 個選択

(a) データフレーム番号によるデータフレーム識別

理由：受信側でフレームに伝送誤りを検出した場合に，どのフレームに誤りがあったかを送信側に知らせる必要があるため

(d) 送信側ノードでの送信データフレーム数の管理

理由：送信側は最大 N 個分のデータフレームしか送信できないので，ACK を受け取っていないフレーム数と後何個のフレームを送れるかの管理が必要になるため

(e) 送信側ノードでの最大 N 個分のデータフレームバッファの管理

理由：受信側から NACK フレームが送信された場合にそのフレームの再送が必要であるのでウインドウサイズ N のデータフレームはバッファに保持しなければならないため

(f) 受信側ノードでの最大 N 個分のデータフレームバッファの管理

理由：最大 N 個のフレームを連続して受信する可能性があるため，N 個分のフレーム用バッファによる管理が必要になるため

II)

(b) 受信側ノードでの受信データフレームの並び替え

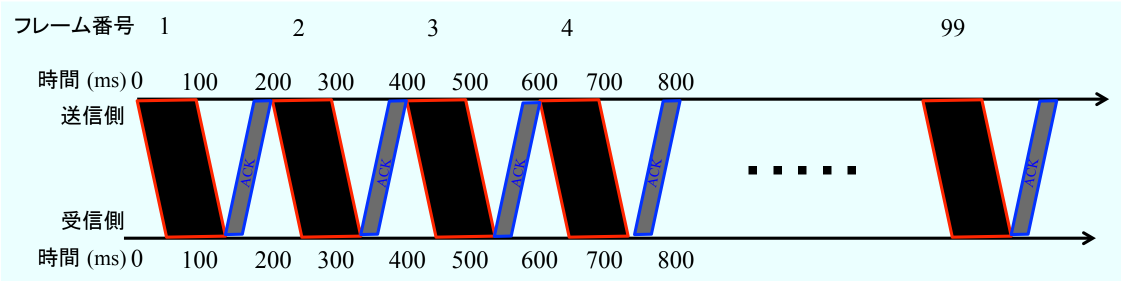
理由：受信側で再送フレームを受信したときに後続番号のフレームは受信済みであるので，順序制御が必要になるため

(c) 受信側ノードでの受信済みデータフレーム番号の管理

理由：順序制御のためにどのフレームを正しく受信し，どのフレームをまだ受信できていないかの管理が必要になるため

(2)

・Stop-and-Wait ARQ 方式



- 平均スループット $= 500$bps
- 計算過程

片方向伝搬遅延時間の 40(ms) を Δ(ms) と表記する．

通信速度 1000bps の通信路において長さ 100bit のデータを送信（受信）するのに要する時間を T_{frame}，ACK フレームを送信（受信）するのに要する時間を T_{ack} とすると以下になる．この値は後続の問でも使用する．

$$T_{\mathrm{frame}} = \frac{\text{送信データフレーム長}}{\text{通信速度 (bps)}} = \frac{100}{1000} = 0.1(\mathrm{s}) = 100(\mathrm{ms})$$

$$T_{\mathrm{ack}} = \frac{\text{送信データフレーム長}}{\text{通信速度 (bps)}} = \frac{50}{1000} = 0.02(\mathrm{s}) = 20(\mathrm{ms})$$

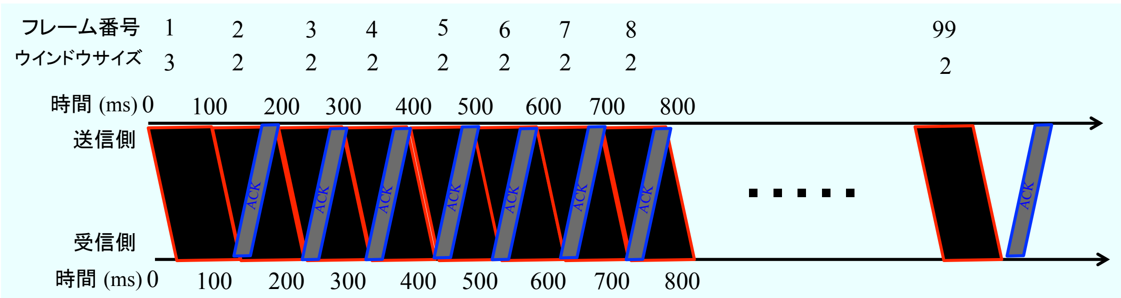
Stop-and-Wait ARQ 方式におけるデータフレームの送信の過程は上図のように図式できる．ある 1 つのデータフレームの送信を開始して ACK の受信完了までの時間は，

$$T_{\mathrm{frame}} + 2\Delta + T_{\mathrm{ack}}$$

である．従って，平均スループットは以下のように計算できる．

$$\frac{\text{送信データ量}}{\text{全データ送信時間}} = \frac{100 \times 99}{(T_{\mathrm{frame}} + 2\Delta + T_{\mathrm{ack}}) \times 99} \times 10^3 = 500(\mathrm{bps})$$

・Go-Back-N ARQ 方式



- 平均スループット $= 990$(bps)
- 計算過程

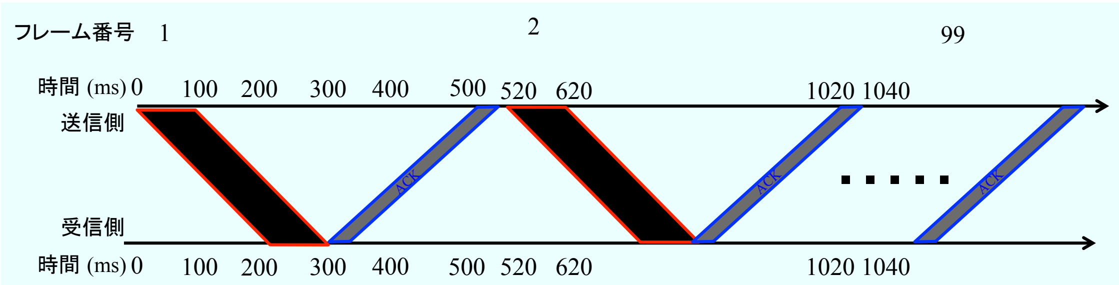
片方向伝搬遅延時間が Δ(ms) である場合は，上図のように，あるデータフレームに対

するACKは2つ後続のデータフレーム送信前に受信完了する．従って，2つ目以降のデータフレームの送信の場合，ウインドウサイズは必ず2であることを保証できるので，99個のデータフレームを連続して送信し続けることが可能である．従って，平均スループットは以下のように計算できる．

$$\frac{\text{送信データ量}}{\text{全データ送信時間}} = \frac{100 \times 99}{T_{\text{frame}} \times 99 + 2\Delta + T_{\text{ack}}} \times 10^3 = 990(\text{bps})$$

(3)

・Stop-and-Wait ARQ 方式

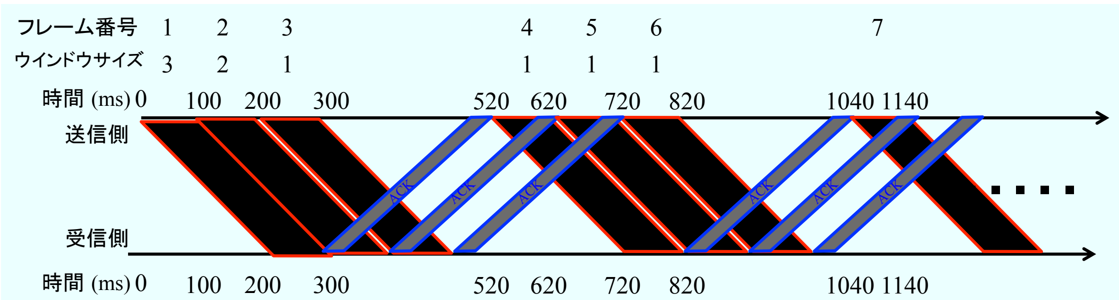


- 平均スループット ＝ 192(bps)
- 計算過程

  (2) の場合と比較して，片方向伝搬遅延時間が5倍になっているが他の条件は同じであるので，平均スループットの計算は (2) の Δ を 5Δ に変更すればよい．従って，

$$\frac{\text{送信データ量}}{\text{全データ送信時間}} = \frac{100 \times 99}{(T_{\text{frame}} + 2 \times 5\Delta + T_{\text{ack}}) \times 99} \times 10^3 = 192(\text{bps})$$

・Go-Back-N ARQ 方式



- 平均スループット ＝ 570(bps)
- 計算過程

  (2) の場合と比較して，片方向伝搬遅延時間が5倍になっていることにより，ウインドウサイズが0になりACK受信完了までデータフレームを送信できない時間が存在する．上図のように3つ連続してデータフレームを送信した後，先頭のデータフレームに対し

ての ACK は最後のデータフレームの送信完了後，220(ms) 後に受信完了するので，次のデータフレームを送信し，その後，連続して後続の 2 つのデータフレームを送信することができる．これを繰り返すと，上図のようにして 3 つのデータフレームを連続して送信した後，220(ms) 待って，次の 3 つの連続したデータフレームを送信する．

従って，3 つのデータフレームを送信し 220(ms) 待つことを繰り返すことになる（繰り返し回数は 33 回）．最後は 98，99 番目のデータフレームに対する ACK フレーム受信のために 200(ms) 余分 に要することを考慮して，平均スループットは以下のように計算できる．

$$\frac{\text{送信データ量}}{\text{全データ送信時間}} = \frac{100 \times 99}{(T_{\text{frame}} \times 3 + 5\Delta + T_{\text{ack}}) \times 33 + 200} \times 10^3 = 570(\text{bps})$$

(4)

・Stop-and-Wait ARQ 方式

- 平均スループット = 450(bps)
- 計算過程

0.1 の確率でフレーム誤りが発生すること以外は，(2) の条件と同じであるので，再送フレームを含めていくつのフレームを送信するかがわかれば，(2) と同様の方法でスループットを計算できる．

この場合，フレーム誤りが発生したフレームを送信側が再送する場合に再送フレームも 0.1 の確率でフレーム誤りが発生することに注意する．

従って，再送フレームも含めたフレームの送信回数（期待値）は，

$$99 \times \sum_{k=0}^{\infty} \left(\frac{1}{10}\right)^k = 99 \times \frac{1}{1 - \frac{1}{10}} = 99 \times \frac{10}{9} = 110 \text{ 回}$$

である．従って，スループットは，

$$\frac{\text{送信データ量}}{\text{全データ送信時間}} = \frac{100 \times 99}{(T_{\text{frame}} + 2\Delta + T_{\text{ack}}) \times 110} \times 10^3 = 450(\text{bps})$$

・Go-Back-N ARQ 方式

- 平均スループット = 818(bps)
- 計算過程

(2) の条件のもとで，0.1 の確率でフレーム誤りが発生する場合，受信側からの NACK フレームを受信したときには，送信側ではフレーム誤りが発生したデータフレームと次のデータフレームを送信済みである．従って，再送の場合は 2 つのデータフレームを再送

しなければならない．ただし，最後の 99 番目のデータフレームに関しては再送フレームは 1 つのみである．

Stop-and-Wait ARQ 方式と同様で，フレーム誤りが発生したフレームを送信側が再送する場合に再送フレームも 0.1 の確率で フレーム誤りが発生することに注意して，再送フレームも含めたフレームの送信回数 (期待値) は，

$$98+98\times\sum_{k=1}^{\infty}\left\{2\times\left(\frac{1}{10}\right)^k\right\}+\sum_{k=0}^{\infty}\left(\frac{1}{10}\right)^k=98+98\times2\times\frac{1}{10}\times\frac{1}{1-\frac{1}{10}}+\frac{1}{1-\frac{1}{10}}$$

$$=98+\frac{98}{5}\times\frac{10}{9}+\frac{10}{9}=98+22=120\ (\text{回})$$

である．従って，スループットは，

$$\frac{\text{送信データ量}}{\text{全データ送信時間}}=\frac{100\times99}{T_{\text{frame}}\times120+2\Delta+T_{\text{ack}}}\times10^3=818(\text{bps})$$

(5)

グラフの横軸は，**フレーム誤りやフレームロスが発生する確率**を表している．

上： Selective Repeat ARQ 方式

理由：フレーム誤りやフレームロスが発生した場合でも，そのデータフレームだけを再送すればよく正しく送信できたデータフレームは受信側で破棄しないので，スループットは他の方式よりもよいため．

中： Go-Back-N ARQ 方式

理由：Selective Repeat ARQ 方式と比べ，フレーム誤りやフレームロスが発生した場合にそのデータフレームに加え，送信済みの後続データフレームも送信する必要があるため，送信する再送フレーム数が多いため，スループットは小さい．しかし，最大 N 個まで同時にデータフレームを送信することができるため，Stop-and-Wait ARQ 方式よりはスループットが大きい．

下： Stop-and-Wait ARQ 方式

理由：1 つのデータフレームを送信後，受信側から ACK フレームを受信するまで次のデータフレームを送信できないのでデータフレーム送信に最も時間がかかるため，スループットは小さくなるため．

解説

　ネットワークの問題は，出題傾向，難易度，作業量が年によって大きく異なる．2011 年入試までのネットワークの問題を比較して，本年が計算が多く，最も難しいと思われる．このレベルの問題であれば，他の選択問題に逃げた方がよいかもしれない．

　ただ，傾向が大きく異なるとはいえ，講義：情報ネットワークの教科書「マルチメディア情報ネットワーク」の内容を 1 通り理解することができていれば対応はできるはずである．

(1)

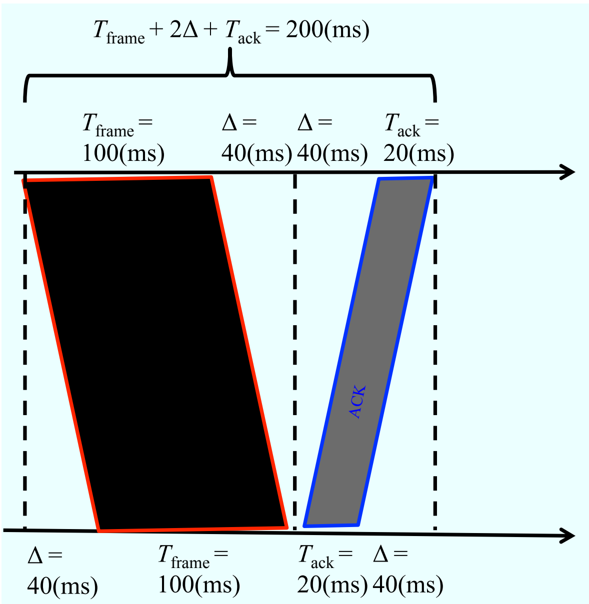
　Stop-and-Wait ARQ 方式，Go-Back-N ARQ 方式，Selective Repeat ARQ 方式のそれぞれの特徴を理解できていれば解答できる．理由に関しては，解答記載の通り．

(2)

　平均スループットの計算は，問題文にあるように送信側ノードにおける 1 つめのデータフレーム送信開始から最後のデータフレームに対する ACK 受信完了までの単位時間あたりの送信データ量である．送信データ量は 100(bit) × 99(個) で一定である．従って，スループットの計算のために全データフレームの送信，ACK フレーム受信完了時間を計算することが必要になる．

・Stop-and-Wait ARQ 方式

　解答の図に示したように 1 つのデータフレームを送信して ACK フレームを受信するまでが繰り返し単位（下図）となっている．



従って，200(ms) × 99(個) が全データフレームの送信，ACK フレーム受信完了時間である．

・Go-Back-N ARQ 方式

片方向伝搬遅延時間が 40(ms) であるため，解答の図，および，計算過程で示したようにウインドウサイズは 2 つ目のデータフレーム以降の送信においてウインドウサイズは常に 2 であるので，送信側ノードはデータフレームを連続して送信することが可能である．従って，最後のデータフレームに対する ACK 受信までの時間：100(ms) を考慮して，100(ms) × 99(個)+100(ms) が全データフレームの送信，ACK フレーム受信完了時間である．

(3)

・Stop-and-Wait ARQ 方式

(2) の場合と比較して，片方向伝搬遅延時間が 5 倍になった以外の条件は等しいので，520(ms) × 99(個) が全データフレームの送信，ACK フレーム受信完了時間である．

・Go-Back-N ARQ 方式

(2) の場合と比較して，片方向伝搬遅延時間が 5 倍に大きくなったので，連続してデータフレームを送信できないことを考慮する必要がある．

解答の図に示したように，3 つのデータフレームを送信完了後（この時点でウインドウサイズは 0），220(ms) 後に最初のデータフレームに対する ACK フレームを受信完了する．後に 100(ms) ごとに後続のデータフレームに対する ACK フレームを受信するので，3 つ連続してデータフレームを送信することができる（この時点でまたウインドウサイズは 0）．

従って，3 つ連続してデータフレームを送信した後，220(ms) 待つ．これを繰り替えし単位として捉え，33(=99/3) 回繰り返すことになる．最後は ACK フレーム受信のために 200(ms) 余分に要することを考慮して，520 × 33+200(ms) が全データフレームの送信，ACK フレーム受信完了時間である．

(4)

フレーム誤りが 0.1 の確率で発生することを除き，条件は (2) のままであるので基本は (2) と同様の計算で行う．問題は**再送フレームを含め，最終的に何個のデータフレームを送信するか**であり，この点をしっかりと計算すればよい．Go-Back-N ARQ 方式の場合は，解答に示したように計算が煩雑．

(5)

どの方式においてもスループットが単調減少であることと，横軸の値が最大であるときにどの方式でもスループットが 0 であることから，横軸はフレーム誤りやフレームロスが発生する確率を表していると判断できる．

上中下の判断に関しては，各方式の特徴を踏まえて判断できるだろう．

雑感

2011 年度入試までをみて，問題形式が現在の形式になっている 2006 年以降では出題分野は同じで，基本的には過去問の内容に沿って勉強を進めればよい感じです．ただ，2010 年度，2011 年度と難化，作業量増大の傾向にある（平均点が 5 割弱?）ので，高得点は取りにくくなっています．しかし，ポイントを押さえて勉強すれば十分に及第点は取れるでしょう．

また，参考までに藤田が面接で聞かれた質問内容，および，提出した志望動機を以下に上げています．稚拙な文章ですがよかったら活用ください．

`\\viola\share\学業\院試過去問\志望動機`

解答は一応精査していますが，間違い等があり修正の必要がありましたら，TEX ソースファイルもサーバに挙げていますので訂正をお願いします．また，計算理論，情報論理学，ネットワーク以外の選択問題を解いていただければありがたいです．