

平成22年度	後期理解度テスト2問題			
科目名	コンバイラ			
担当教員	石原真紀夫			
実施日付	1月21日（金） 1限目（1組・2組合同） A11			
持ち込み	許可・禁止			
情報工学科	年	組	学籍番号	
氏名				
	_____点／104点			

《表記》BNFの記号として ::= | ■* ■+ を用います。

問1 文法1の下降型構文解析LL（1）に関する以下の設問に答えなさい。
（文法1）
非終端記号 A B P E
終端記号 n + − =
生成規則
A ::= n P n B ①
B ::= P n B ②
B ::= ε ③ ※ε（イプシロン）は長さ0の語です
P ::= + E ④
P ::= − E ⑤
E ::= = ⑥ 出発記号 A

設問1 文法1における次の First（ ）と Follow（ ）を答えなさい。ここで、Follow（A）＝ { \$ } です。【各2計14】
〔1〕First（A） 〔2〕First（B） 〔3〕First（P）
〔4〕First（E） 〔5〕Follow（B） 〔6〕Follow（P）
〔7〕Follow（E）

設問2 文法1の LL 解析表を以下に示します。入力記号列をn＋＝nとして構文解析を行いなさい。解析過程は1ステップずつ示し、複数のステップを一度に行わないこととします。【10】
（LL 解析表）

＼	n	+	−	=	\$
A	A→n P n B				
B		B→P n B	B→P n B		B→ε
P		P→+E	P→−E		
E				E→=	

問2 上昇型構文解析SLR（1）に関する以下の設問に答えなさい。
※LR（0）項の括弧 [] 、集合の括弧 { } 、closure（ ）や goto（ ）の引数の括弧（ ）を明確に区別して解答してください。

設問1 文法2のLR 解析表を作りたい。
（文法2）
非終端記号 A B
終端記号 a b
生成規則
A ::= a B ①
B ::= b B ②
B ::= a ③ 出発記号 A
文法2にA'→Aを加えた文法2'の正規集合を下記の空欄を適切に埋めて完成させなさい。【各2計12】
（正規集合）

l₀ = closure({ [〔1〕] })
{[A'→・A], [A→・a B]}
goto(l₀,A) = closure({[A'→A・]})
= { [〔2〕] } = l₁
goto(l₀,a) = closure({[A→a・B]})
= { [〔3〕] } = l₂
goto(l₂,B) = {[A→a B・]} = l₃
goto(l₂,b) = {[B→b・B], [B→・b B], [B→・a]} = l₄
goto(l₂,a) = {[B→a・]} = l₅
goto(l₄, [〔4〕]) = {[B→b B・]} = l₆
goto(l₄, [〔5〕]) = l₄と等しい
goto(l₄, [〔6〕]) = l₅と等しい
よって、正規集合CはC={ l₀, l₁, l₂, l₃, l₄, l₅, l₆ }です。

設問2 以下に示す文法2のLR 解析表の空欄を埋めなさい。【各3計12】

LR 解析	a	b	\$	A	B
0	〔1〕			〔2〕	
1			a c c		
2	s 5	s 4			3
3			〔3〕		
4	s 5	s 4			6
5			〔4〕		
6			r 2		

設問3 文法1のLR 解析表を以下に示す。入力記号列をn＋＝nとして構文解析を行いなさい。解析過程は1ステップずつ示し、複数のステップを一度に行わないこととします。【10】
（LR 解析表）

＼	n	+	−	=	\$	A	B	P	E
0	s 2					1			
1					a c c				
2		s 4	s 5					3	
3	s 6								
4				s 8					7
5				s 8					9
6		s 4	s 5		r 3		1 0	1 1	
7	r 4								
8	r 6								
9	r 5								
10					r 1				
11	s 1 2								
12					r 3		1 3	1 1	
13					r 2				

問3 次に示す式の後置記法表現に関する設問に答えなさい。
〔1〕a a b c * d + + = 〔2〕a a b + c d + * =
〔3〕a a b c d + * + = 〔4〕a a b + c * d + =

設問1 各後置記法表現が表す演算と同等の演算を行う式を答えなさい。【各3計12】
〔a〕a = (a + b) * (c + d) 〔b〕a = (a + b) * c + d
〔c〕a = a + b * (c + d) 〔d〕a = a + b * c + d

設問2 各後置記法表現が表す演算と同等の演算を行う4つ組表現を答えなさい。ここで、AとBは演算結果保持用の一時変数とします。【各3計12】
〔a〕(+ , c , d , A) 〔b〕(* , b , c , A)
(+ , a , b , B) (+ , A , d , B)
(* , B , A , a) (+ , a , B , a)
〔c〕(+ , c , d , A) 〔d〕(+ , a , b , A)
(* , b , A , B) (* , A , c , B)
(+ , a , B , a) (+ , B , d , a)

問4 次に示すスタック機械用の各コードを実行した後のスタックの状態を解答欄の記入例にしたがい答えなさい。ただし、最初スタックは空の状態とします。下の表にスタック機械の命令コードとその機能、意味を示します。【各2計8】

〔1〕PUSH 1 〔4〕PUSH 7
PUSH 2 PUSH 8
ADD LTOP
〔2〕PUSH 3 PUSH 9
PUSH 4 PUSH 1
SUB GTOP
〔3〕PUSH 5 ANDOP
PUSH 6
EQOP

問5 ハッシュ法を用いた記号表のデータ構造に関する設問に答えなさい。以下に識別子とそのハッシュ値、2通りのハッシュ表の状態を示します。

〔ハッシュ値〕		〔ハッシュ表1〕		〔ハッシュ表2〕	
iphone	0	0	house	0	house
blueray	1	1		1	iphone
pine	1	2		2	3dtv
3dtv	2	3	car	3	car
dog	2	4		4	apple
apple	2	5		5	pine
android	4	6		6	dog
cat	4	7		7	blueray
		8		8	cat
		9		9	android
		10		10	

設問1 次の順に識別子をハッシュ表1に格納した後のハッシュ表の状態をそれぞれ答えなさい。ただし、衝突した場合は線形走査法を用いることとします。【各2計4】
〔1〕dog → cat → apple → pine
〔2〕3dtv → blueray → iphone → android

設問2 ハッシュ表2において次の各識別子を探索した場合の探索回数を答えなさい。例えば、dog を探索すると5回目の探索で見つかりますので探索回数は5です。【各2計6】
〔1〕cat 〔2〕apple 〔3〕pine

設問3 ある原始プログラム内で出現した識別子とその割合が次のようであった場合、ハッシュ表2における識別子の平均探索回数はいくらか答えなさい。【4】

出現識別子	出現割合	出現識別子	出現割合
3dtv	20%	lphone	10%
blueray	30%	android	40%

（スタック機械命令コード）			
命令(オペコード)	パラメータ(オペランド)	機能	意味
PUSH	para	プッシュ	paraが変数であればその中の値をスタックへ積み、paraが数値であればその値をスタックへ積み。
POP		ポップ	スタックのトップの値を取り出す。
ASSIGN	var	アサイン	スタックのトップをポップし、その値を変数varへ書き込む。
JUMP	label	分岐	ラベルlabelへ飛ぶ。
FJUMP	label	分岐	スタックのトップをポップし、0であればラベルlabelへ飛ぶ。
TJUMP	label	分岐	スタックのトップをポップし、0でなければラベルlabelへ飛ぶ。
INV		符号反転	スタックのトップをポップし、その値の符号を反転する。結果をスタックのトップへプッシュする。
ADD		加算	スタックのトップと2番目をポップし、それらを加算する。結果をスタックのトップへプッシュする。
SUB		減算	スタックのトップと2番目をポップし、2番目からトップを減ずる。結果をスタックのトップへプッシュする。
MULT		乗算	スタックのトップと2番目をポップし、それらを乗算する。結果をスタックのトップへプッシュする。
DIV		除算	スタックのトップと2番目をポップし、2番目をトップで割る。結果をスタックのトップへプッシュする。
MOD		剰余	スタックのトップと2番目をポップし、2番目をトップで割ったときの余りを計算する。結果をスタックのトップへプッシュする。
GTOP		>	スタックのトップと2番目をポップし、2番目がトップより大きければ1、そうでなければ0をスタックのトップへプッシュする。
GEOP		>=	スタックのトップと2番目をポップし、2番目がトップ以上であれば1、そうでなければ0をスタックのトップへプッシュする。
LTOP		<	スタックのトップと2番目をポップし、2番目がトップより小さければ1、そうでなければ0をスタックのトップへプッシュする。
LEOP		<=	スタックのトップと2番目をポップし、2番目がトップ以下であれば1、そうでなければ0をスタックのトップへプッシュする。
EQOP		==	スタックのトップと2番目をポップし、2番目とトップが等しいならば1、そうでなければ0をスタックのトップへプッシュする。
NEOP		!=	スタックのトップと2番目をポップし、2番目とトップが等しくないならば1、そうでなければ0をスタックのトップへプッシュする。
ANDOP		&&	スタックのトップと2番目をポップし、2番目とトップで一方または両方が0ならば0、そうでなければ1をスタックのトップへプッシュする。
OROP			スタックのトップと2番目をポップし、2番目とトップで両方が0ならば0、そうでなければ1をスタックのトップへプッシュする。

【解答欄】			
【問1 設問1】			
〔1〕First（A）＝	{ n }		
〔2〕First（B）＝	{ + , − , ε }		
〔3〕First（P）＝	{ + , − }		
〔4〕First（E）＝	{ = }		
〔5〕Follow（B）＝	Follow（A）＝ { \$ }		
〔6〕Follow（P）＝	{ n }		
〔7〕Follow（E）＝	Follow（P）＝ { n }		

【問1 設問2】			
＼	入力記号列	スタック	動作
1	n + = n \$	A \$	A→n P n B
2	n + = n \$	n P n B \$	nを削除
3	+ = n \$	P n B \$	P→+E
4	+ = n \$	+ E n B \$	+を削除
5	= n \$	E n B \$	E→=
6	= n \$	= n B \$	=を削除
7	n \$	n B \$	nを削除
8	\$	B \$	B→ε
9	\$	\$	（終わり）
10			
11			
12			

【問2 設問1】			
〔1〕	[A'→・A]		
〔2〕	[A'→A・]		
〔3〕	[A→a・B]、[B→・b B]、[B→・a]		
〔4〕	B		
〔5〕	b		
〔6〕	a		

【問2 設問2】			
〔1〕	s 2	〔2〕	1
〔3〕	r 1	〔4〕	r 3

【問2 設問3】			
＼	入力記号列	スタック	動作
1	n + = n \$	0	s 2
2	+ = n \$	2 0	s 4
3	= n \$	4 2 0	s 8
4	n \$	8 4 2 0	r 6
5	n \$	7 4 2 0	r 4
6	n \$	3 2 0	s 6
7	\$	6 3 2 0	r 3
8	\$	1 0 6 3 2 0	r 1
9	\$	1 0	a c c
10			
11			
12			
13			

【問3 設問1】			
〔1〕	d	〔2〕	a
〔3〕	c	〔4〕	b

【問3 設問2】			
〔1〕	b	〔2〕	a
〔3〕	c	〔4〕	d

【問4】
〔記入例〕スタック機械用のコードを実行した後に、スタック中に数値1が底に、数値2がその上に残る場合は次のように示して下さい。

2	
1	

〔1〕

3	

〔3〕

0	

【問5 設問1】

〔1〕	
0	house
1	pine
2	dog
3	car
4	cat
5	apple
6	

【問5 設問2】

〔1〕	5	〔2〕	3
〔3〕	5		

【問5 設問3】	
平均探索回数	4. 9

お疲れ様でした