

平成23年度	後期理解度テスト2問題解答
科目名	コンバイラ
担当教員	石原真紀夫
実施日付	1月19日(木)
	2回目(1組・2組合同) A13
持ち込み	許可・禁止
情報工学科	年 組 学籍番号
氏名	
	点/102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

終端記号 n O + -

生成規則

A ::= n B A ①

A ::= O ②

B ::= + ③

B ::= - ④

出発記号 A

設問1 文法1での次のFirst()とFollow()を答えなさい。【各3計12】
〔1〕First(A) 〔2〕First(B) 〔3〕Follow(A) 〔4〕Follow(B)

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-O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。【6】

《文法1のLL解析表》

	n	O	+	-	\$
A	A→nBA	A→O			
B			B→+	B→-	

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(O)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。
(文法2)

非終端記号 A B C

終端記号 a 「 n 」

生成規則

A ::= a B ①

B ::= C B ②

B ::= C ③

C ::= 「 n 」 ④

出発記号 A

設問1 文法2での次のclosure()とgoto()を答えなさい。【各3計18】
〔1〕closure({ [B→・C] })
〔2〕closure({ [A→a・B] })
〔3〕closure({ [B→C・] })

I₀ = { [C→「n・」] } I₁ = { [B→・CB] } I₂ = { [B→・C] }

〔4〕goto(I₀, n) 〔5〕goto(I₁, C) 〔6〕goto(I₂, C)

設問2 文法2にA'→Aを加えた文法2'の正規集合は以下の通りです。LR解析表の各空欄を適切に埋めなさい。【各2計14】
(正規集合)

I₀ = { [A'→・A]、[A→・aB] }

goto(I₀, A) = { [A'→A・] }

= I₁

goto(I₀, a) = { [A→a・B]、[B→・CB]、
[B→・C]、[C→・「n」] }

= I₂

goto(I₂, B) = { [A→aB・] }

= I₃

goto(I₂, C) = { [B→C・B]、[B→C・]、[B→・CB]、
[B→・C]、[C→・「n」] }

= I₄

goto(I₂, 「」) = { [C→「n・」] }

= I₅

goto(I₄, B) = { [B→CB・] }

= I₆

goto(I₄, C) = I₄

goto(I₄, 「」) = I₅

goto(I₅, n) = { [C→「n・」] }

= I₇

goto(I₇, 「」) = { [C→「n」・] }

= I₈

よって、正規集合Cは { I₀, I₁, I₂, I₃, I₄, I₅, I₆, I₇, I₈ } となります。

《文法2のLR解析表》

	a	「	」	n	\$	A	B	C
0	s2					1		
1					〔1〕			
2		s5					〔2〕	4
3					〔3〕			
4		〔4〕			r3		6	〔5〕
5				s7				
6					r2			
7			〔6〕					
8		〔7〕			r4			

設問3 文法1のLR解析表を以下に示します。入力記号列をn+n-O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。【6】

《文法1のLR解析表》

	n	O	+	-	\$	A	B
0	s2	s3				1	
1					a c c		
2			s5	s6			4
3					r2		
4	s2	s3				7	
5	r3	r3					
6	r4	r4					
7					r1		

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〕abc÷d-=-

〔2〕abcd÷-=-

〔3〕abc-d÷=-

〔4〕abcd-d÷=-

設問1 各後置記法表現が表す構文木を描きなさい。【各2計8】

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つ組表現をそれぞれ答えなさい。ここで、Aは演算結果保持用の一時変数とします。【各2計8】

〔a〕(-, c, d, A) 〔b〕(÷, c, d, A)

(÷, b, A, a)

(-, b, A, a)

〔c〕(-, b, c, A) 〔d〕(÷, b, c, A)

(÷, A, d, a) (-, A, d, a)

問4 次の各処理を実行するスタック機械プログラムを以下のように作成しました。プログラム中のそれぞれの空欄を適切な命令で埋めなさい。スタック機械の各命令の機能と意味は下の表に示します。【各2計16】

〔1〕if(a<b) a=2;

〔2〕if(a==b) a=2;

〔3〕if(a==b) a=2; else b=2;

〔4〕while(a!=b) a+=2;

〔5〕while(a<b) a+=2;

〔6〕do a+=2; while(a!=b);

《スタック機械プログラム》※上の番号と対応します

〔1〕	PUSH a PUSH b ①	〔2〕	PUSH a PUSH b ②
	FJUMP L1 PUSH 2 ASSIGN a		TJUMP L1 PUSH 2 ASSIGN a
L1:		L1:	
〔3〕	PUSH a ③ EQOP FJUMP L1 ④ ASSIGN a JUMP L2	〔4〕L1:	PUSH a PUSH b NEOP ⑤ PUSH a PUSH 2 ADD ASSIGN a JUMP L1
L1:	PUSH 2 ASSIGN b	L2:	
L2:		L2:	
〔5〕L1:	PUSH a PUSH b LTOP FJUMP L2 PUSH a PUSH 2 ADD ASSIGN a ⑥	〔6〕L1:	PUSH a PUSH 2 ⑦ ASSIGN a PUSH a PUSH b NEOP ⑧
L2:			

問5 ハッシュ法に関する以下の設問に答えなさい。

設問1 以下に識別子とそのハッシュ値を示します。〔1〕～〔3〕に示す各順で識別子を下のハッシュ表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6】

〔ハッシュ値〕		〔ハッシュ表〕	
識別子	ハッシュ値	添え字	データ
pos_x	1	0	
pos_y	3	1	
distance	4	2	area
length	2	3	
color1	0	4	upper
color2	5	5	lower
tm	6	6	
cnt	7	7	

〔1〕pos_x → length → distance

〔2〕pos_y → length → distance

〔3〕tm → distance → length

設問2 ある2つの識別子AとBのハッシュ値をハッシュ関数の一つである平方探中法を用いて求め、それぞれ10進数で答えなさい。各識別子の文字コード値は表のとおりとし、ハッシュ表のサイズは16とします。【各4計8】

識別子	文字コード値
A	24
B	15

《スタック機械命令一覧》

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む。
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	スタックのトップと2番目をポップし、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する。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, O}
〔2〕First(B) = {+, -}
〔3〕Follow(A) = {\$}
〔4〕Follow(B) = First(A) = {n, O}

	入力記号列	スタック	動作
0	n + n - O \$	A \$	A→nBA
1	n + n - O \$	n B A \$	n消去
2	+ n - O \$	B A \$	B→+
3	+ n - O \$	+ A \$	+消去
4	n - O \$	A \$	A→nBA
5	n - O \$	n B A \$	n消去
6	- O \$	B A \$	B→-
7	- O \$	- A \$	-消去
8	O \$	A \$	A→O
9	O \$	O \$	O消去
10	\$	\$	a c c
11			

【問2 設問1】
〔1〕closure({ [B→・C] }) = = { [B→・C]、[C→・「n」] }

〔2〕closure({ [A→a・B] }) = = { [A→a・B]、[B→・CB]、[B→・C]、[C→・「n」] }
〔3〕closure({ [B→C・] }) = = { [B→C・] }
〔4〕goto(I ₀ , n) = closure({ [C→「n・」] }) = { [C→「n・」] }
〔5〕goto(I ₁ , C) = closure({ [B→C・B] }) = { [B→C・B]、[B→・CB]、[B→・C]、[C→・「n」] }
〔6〕goto(I ₂ , C) = closure({ [B→C・] }) = { [B→C・] }

【問2 設問2】

〔1〕acc	〔2〕3	〔3〕r1	〔4〕s5
〔5〕4	〔6〕s8	〔7〕r4	

【問2 設問3】

	入力記号列	スタック	動作
0	n + n - 0 \$	0	s 2
1	+ n - 0 \$	2, 0	s 5
2	n - 0 \$	5, 2, 0	r 3
3	n - 0 \$	4, 2, 0	s 2
4	- 0 \$	2, 4, 2, 0	s 6
5	0 \$	6, 2, 4, 2, 0	r 4
6	0 \$	4, 2, 4, 2, 0	s 3
7	\$	3, 4, 2, 4, 2, 0	r 2
8	\$	7, 4, 2, 4, 2, 0	r 1
9	\$	7, 4, 2, 0	r 1
10	\$	1, 0	a c c
11			

【問3 設問1】			
〔1〕		〔2〕	
〔3〕		〔4〕	

【問3 設問2】			
〔1〕	d	〔2〕	b
〔3〕	c	〔4〕	a

【問4】

①	LTOP	②	NEOP
③	PUSH b	④	PUSH 2
⑤	FJUMP L2	⑥	JUMP L1
⑦	ADD	⑧	TJUMP L1

【問5 設問1】

	〔1〕		〔2〕		〔3〕
0		0		0	
1	pos_x	1		1	
2	area	2	area	2	area
3	length	3	pos_y	3	length
4	upper	4	upper	4	upper
5	lower	5	lower	5	lower
6	distance	6	length	6	tm
7		7	distance	7	distance

【問5 設問2】

Aのハッシュ値	8	Bのハッシュ値	8
---------	---	---------	---

お疲れ様でした。