2014 年 1 月 31 日システムソフトウエア演習問題

学科　　学籍番号＿＿＿＿＿＿＿＿＿＿＿＿　　氏名＿＿＿＿＿＿＿＿＿＿＿＿

1．　つぎの文章の空白を埋めなさい．（各 1 点：合計 24 点）

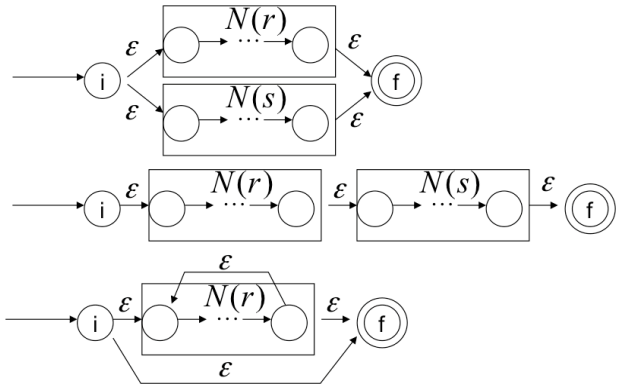
- プログラム言語処理系は，C, C++, Pascal, Basic,などの(A)　　言語で書かれたプログラムテキストを，(1)CPU が解釈実行可能な(B)　　語に変換する(C)　　や，(2)逐次解釈・実行するインタープリタ，などで用いられる．
- プログラムテキストは，(D)　　文法で表現可能な字句の集まりとして表現可能であり，プログラムテキストからこれを抽出する字句解析のためにオートマトンが用いられる．
- 字句解析の後には，構文解析が行われる．通常は(E)　　文法を用いてプログラミング言語を定義することができる．
- 構文解析では，(F)　　記号に対してどのような順序で(G)　　が適用され，入力として与えられた終端記号の列が生成されたのかを分析する．この分析結果は(H)　　木で表すことができる．
- (H)木は, 分岐のない中間節をまとめ, 演算子を分枝節に引き上げることで(I)　　木（演算子木とも呼ばれる）に変換することができる．
- 算術式から得られた(I)木を(J)　　走査することにより(K)　　記法を生成し，これを入力としてスタック上で式の計算を行うことができる．
- 構文解析の方法は大きく分けて(L)　　型の構文解析と，(M)　　型の構文解析法がある．(L)型構文解析では最左導出を前提とした文法の下での構文解析が行われ，(M)型構文解析では最右導出を前提とした構文解析が行われる．
- (L)型構文解析の例としては(N)　　構文解析，(M)型構文解析の例としては，(O)　　構文解析がある．この方法では，アクション表と GOTO 表をもとにして，スタックを用いて shift と reduce の操作を行いながら構文解析が行われる．このうち reduce は，生成規則の(P)　　辺に対応する部分を見つけて(Q)　　辺に置き換えるという(R)　　操作に対応している．
- 文法と文法にマッチした文字列に対する計算手続きを与えて，プログラミング言語の構文解析器を作るソフトウエアのことを(S)　　と呼ぶ. この例としては, Unix や Linux 上の(T)　　やこれを更に拡張した Bison などがある．これらの(S)で用いられているのは(U)　　という講文解析アルゴリズムである．
- (T)や Bison は単体で用いられることはほとんどなく，字句解析用オートマトンを生成する(V)　　や flex と併用されて，(C)やインタープリタが作成される．
- 言語は文法によって規定された(W)　　記号列の(X)　　集合である．プログラミング言語処理系もある文法に従って作成されており，それを使用するプログラマも文法に従ったプログラムを書いている以上，文法上のエラーは起こさない．その意味で，プログラマが使用する言語の文法を知っておくことは必須である．

2. 以下の正規表現（A～D）に対応する記号の集合（すなわち言語）L(A)～L(D)があるとき，L(A)～L(D)の関係はどのようになっているかを BEN 図で表しなさい．但し，集合の交わりの部分がある場合には，その要素も列挙すること．また，a,b,c,d は終端記号を表わす．（６点）

A: a*b*
B: aa*b*
C: a*bb*
D: aa*bb*

３． 正規表現 r を受理する非決定性有限状態オートマトンを $N(r)$ で表す．このとき，正規表現に対応するオートマトンは右図のようになる．この図を参考にして，正規表現 $(r|s)^*$ に対応するオートマトンを図的に表現しなさい．（１０点）

- $r|s$
- rs
- r^*



4. 下記文法のもとで，「3-2-1」の構文解析を行った結果を，導出木，構文木の形で表わしなさい。（導出木７＋構分木３＝10 点）

<算術式> →<算術式><加減演算子><項>|<項>

<項>→ <項><乗除演算子><因子>|<因子>

<因子>→ <変数>|<数値>|(<算術式>)

<数値>→1|2|... |9

<変数>→A|B|C|...|Z

<加減演算子>→+|－

<乗除演算子>→×|÷

5. 右の文法の元で，−1＋1 を LL 構文解析で構文解析するとどうなるか？構文解析表を求め，導出木を示しなさい．(20 点＝構文解析表 10 点＋導出木 10 点)

S→F
S→－F+S
F→ 1

6. 下記の文法とアクション表、GOTO 表が与えられている条件下で「１＋０＊１」の構文解析を行いなさい。結果は導出木で示しなさい。(20 点)

(1) <E> →<E>*<B>
(2) <E> →<E>+<B>
(3) <E> →<B>
(4) <B> → 0
(5) <B> → 1

	アクション					GOTO	
状態	*	+	0	1	$	E	B
0			s1	s2		3	4
1	r4	r4	r4	r4	r4		
2	r5	r5	r5	r5	r5		
3	s5	s6			acc		
4	r3	r3	r3	r3	r3		
5			s1	s2			7
6			s1	s2			8
7	r1	r1	r1	r1	r1		
8	r2	r2	r2	r2	r2		

7. 下記の文法から右下に示すアイテム集合を得た．このとき，アクション表と GOTO 表を示し，衝突が起きている部分を示せ(20 点)

(1) <E> →<E>+<B>

(2) <E> →<B>

(3) <B> →<B>*<T>

(4) <B> →<T>

(5) <T> → 0

(6) <T> → 1

2014 年 1 月 31 日システムソフトウエア演習問題解答例

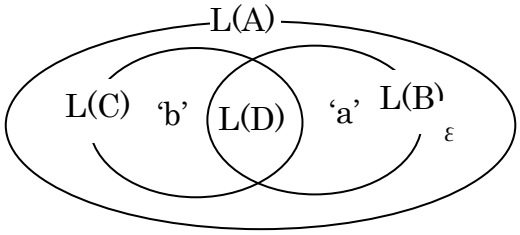
学科　　学籍番号＿＿＿＿＿＿＿＿＿＿　　氏名＿＿＿＿＿＿＿＿＿＿

1．　つぎの文章の空白を埋めなさい．（各 1 点：合計 24 点）

- プログラム言語処理系は，C, C++, Pascal, Basic,などの(A) 高級言語で書かれたプログラムテキストを，(1)CPU が解釈実行可能な(B) 機械語に変換する(C) コンパイラや，(2)逐次解釈・実行するインタープリタ，などで用いられる．
- プログラムテキストは，(D) 正規文法で表現可能な字句の集まりとして表現可能であり，プログラムテキストからこれを抽出する字句解析のためにオートマトンが用いられる．
- 字句解析の後には，構文解析が行われる．通常は(E) 文脈自由文法を用いてプログラミング言語を定義することができる．
- 構文解析では，(F) 出発記号に対してどのような順序で(G) 生成規則が適用され，入力として与えられた終端記号の列が生成されたのかを分析する．この分析結果は(H) 導出木で表すことができる．
- (H)木は，分岐のない中間節をまとめ，演算子を分枝節に引き上げることで(I) 構文木（演算子木とも呼ばれる）に変換することができる．
- 算術式から得られた(I)木を(J) 後順走査することにより(K) 逆ポーランド記法を生成し，これを入力としてスタック上で式の計算を行うことができる．
- 構文解析の方法は大きく分けて(L) 下降型の構文解析と，(M) 上昇型の構文解析法がある．(L)型構文解析では最左導出を前提とした文法の下での構文解析が行われ，(M)型構文解析では最右導出を前提とした構文解析が行われる．
- (L)型構文解析の例としては(N) LL 構文解析，(M)型構文解析の例としては，(O) LR 構文解析がある．この方法では，アクション表と GOTO 表をもとにして，スタックを用いて shift と reduce の操作を行いながら構文解析が行われる．このうち reduce は，生成規則の(P)右　辺に対応する部分を見つけて(Q)左　辺に置き換えるという(R)還元　操作に対応している．
- 文法と文法にマッチした文字列に対する計算手続きを与えて，プログラミング言語の構文解析器を作るソフトウエアのことを(S)　コンパイラ･コンパイラと呼ぶ. この例としては，Unix や Linux 上の(T)　Yacc やこれを更に拡張した Bison などがある．これらの(S)で用いられているのは(U)　LALR という講文解析アルゴリズムである．
- (T)や Bison は単体で用いられることはほとんどなく，字句解析用オートマトンを生成する(V) lex や flex と併用されて，(C)やインタープリタが作成される．
- 言語は文法によって規定された(W) 終端 記号列の(X) 無限集合である．プログラミング言語処理系もある文法に従って作成されており，それを使用するプログラマも文法に従ったプログラムを書いている以上，文法上のエラーは起こさない．その意味で，プログラマが使用する言語の文法を知っておくことは必須である．

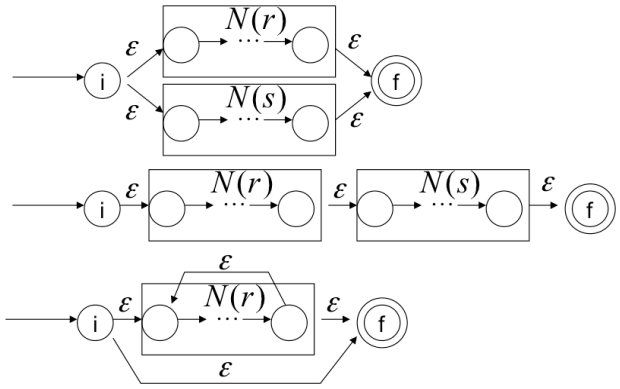
2.　以下の正規表現（A～D）に対応する記号の集合（すなわち言語）L(A)～L(D)があるとき，L(A)～L(D)の関係はどのようになっているかを BEN 図で表しなさい．但し，集合の交わりの部分がある場合には，その要素も列挙すること．また，a,b,c,d は終端記号を表わす．（６点）

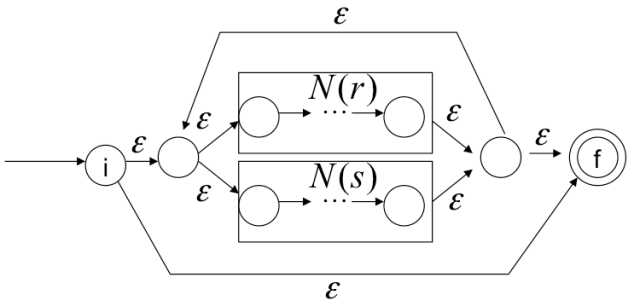
A:	a*b*
B:	aa*b*
C:	a*bb*
D:	aa*bb*



３．　正規表現 r を受理する非決定性有限状態オートマトンを $N(r)$ で表す．このとき，正規表現に対応するオートマトンは右図のようになる．この図を参考にして，正規表現 $(r|s)^*$ に対応するオートマトンを図的に表現しなさい．（１０点）

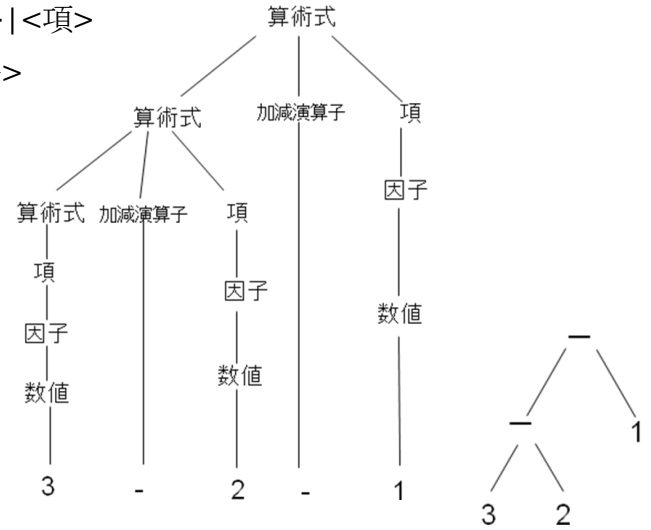
- $r|s$
- rs
- r^*





4.　下記文法のもとで，「3-2-1」の構文解析を行った結果を，導出木，構文木の形で表わしなさい。（導出木７＋構分木３＝10 点）

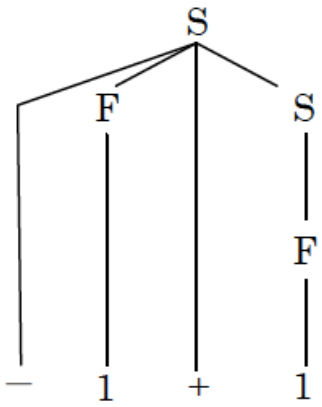
<算術式> →<算術式><加減演算子><項>|<項>
<項>→ <項><乗除演算子><因子>|<因子>
<因子>→ <変数>|<数値>|(<算術式>)
<数値>→1|2|... |9
<変数>→A|B|C|...|Z
<加減演算子>→+|－
<乗除演算子>→×|÷



5.　右の文法の元で，−1＋1 を LL 構文解析で構文解析するとどうなるか？構文解析表を求め，導出木を示しなさい．(20 点＝構文解析表 10 点＋導出木 10 点)

S→F
S→－F+S
F→1

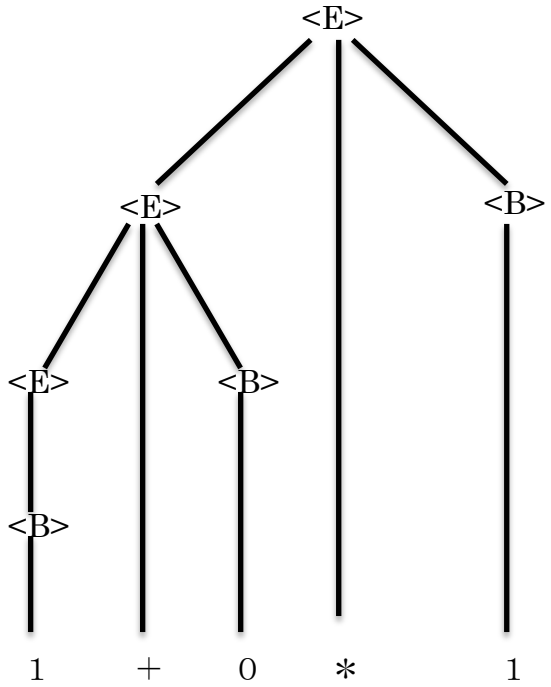
	1	+	−	$
S	1	-	2	-
F	3	-	-	-



6. 下記の文法とアクション表、GOTO 表が与えられている条件下で「１＋０＊１」の構文解析を行いなさい。結果は導出木で示しなさい。(20 点)

(1) <E> →<E>*<B>
(2) <E> →<E>+<B>
(3) <E> →<B>
(4) <B> → 0
(5) <B> → 1

	アクション					GOTO	
状態	*	+	0	1	$	E	B
0			s1	s2		3	4
1	r4	r4	r4	r4	r4		
2	r5	r5	r5	r5	r5		
3	s5	s6			acc		
4	r3	r3	r3	r3	r3		
5			s1	s2			7
6			s1	s2			8
7	r1	r1	r1	r1	r1		
8	r2	r2	r2	r2	r2		



7. 下記の文法から右下に示すアイテム集合を得た．このとき，アクション表と GOTO 表を示し，衝突が起きている部分を示せ(20 点)

(1) <E> →<E>+<B>

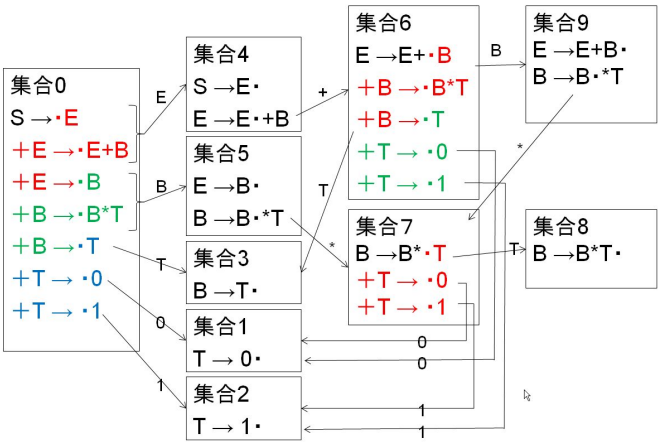
(2) <E> →<B>

(3) <B> →<B>*<T>

(4) <B> →<T>

(5) <T> → 0

(6) <T> → 1



	アクション						GOTO		
状態	*	+	0	1	$		E	B	T
0			s1	s2			4	5	3
1	r5	r5	r5	r5	r5				
2	r6	r6	r6	r6	r6				
3	r4	r4	r4	r4	r4				
4		s6			acc				
5	s7 r2	r2	r2	r2	r2				
6			s1	s2				9	3
7			s1	s2					8
8	r3	r3	r3	r3	r3				
9	s7 r1	r1	r1	r1	r1				

以下余白