CHAPTER 6 FRANCIS Compiler

編譯器實作

6.1 Lexical Analysis

```
PROGRAM MAIN;
VARIABLE INTEGER:U,V,M;
U = 5;
V = 7;
CALL   S1(U ,V , M );
ENP;
SUBPOUTINE  S1(INTEGER:X,Y,M);
M = X + Y + 2.7;
ENS;
```

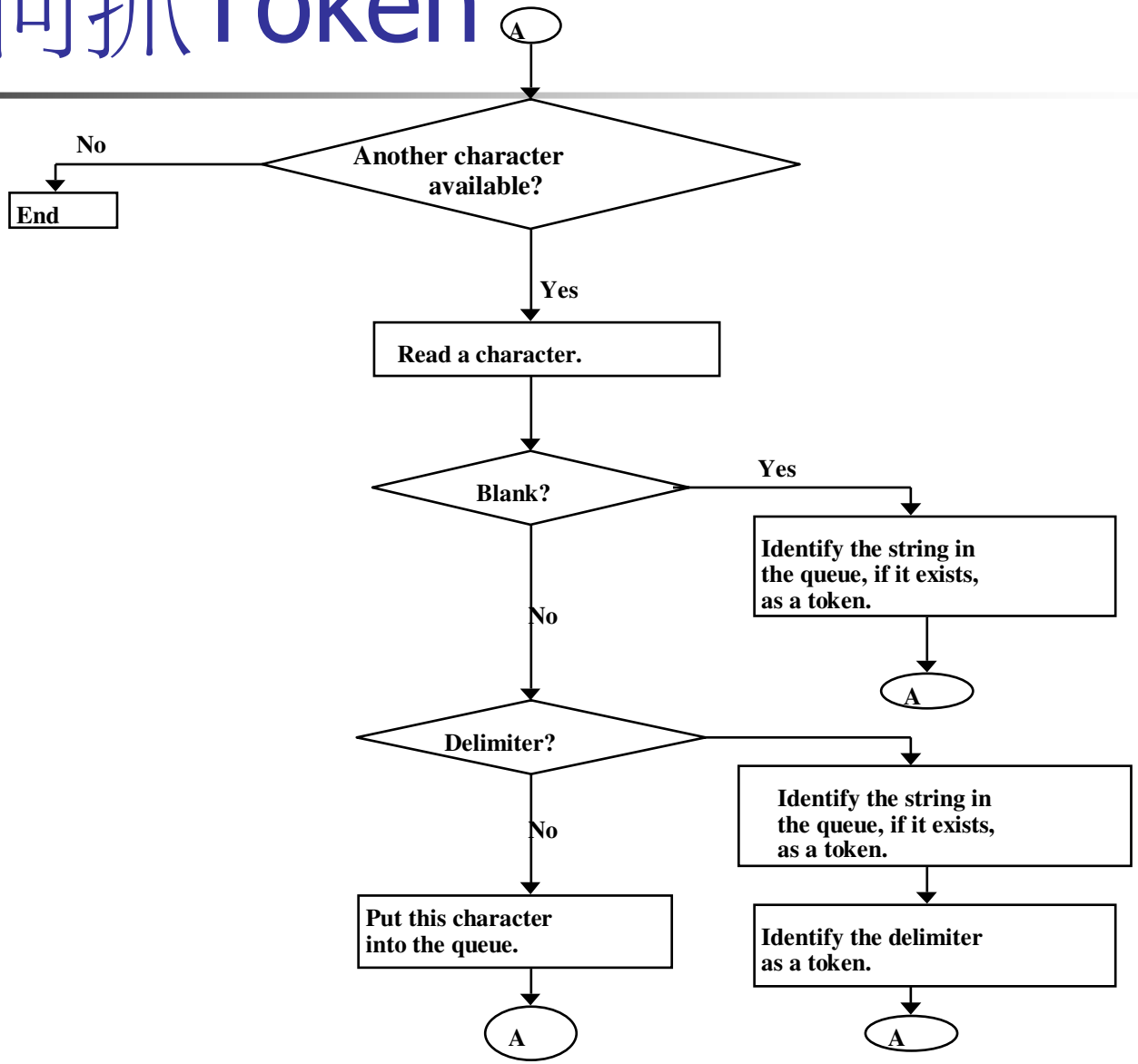
FRANCIS 語言所寫之程式

Lexical Analysis(2)

```
PROGRAM MAIN;
(2,21)          (5,3) (1,1)

VARIABLE INTEGER:       U       ,       V       ,       M       ;
(2,25)          (2,14)      (1,12) (5,1) (1,11) (5,5) (1,11) (5,6) (1,1)

U       =               5       ;
(5,1)   (1,4)       (3,1)   (1,1)

V       =           7       ;
(5,5)   (1,4)   (3,2)   (1,1)

CALL    S1      (       U       ,       V       ,       M       )       ;
(2,3)   (5,10) (1,2) (5,1) (1,11) (5,5) (1,11) (5,6) (1,3) (1,1)

ENP     ;
(2,6)   (1,1)

SUBPOUTINE  S1      (       INTEGER :       X       ,       Y       ,       M       )       ;
(2,23)              (5,10)  (1,2)  (2,14)       (1,12)  (5,8) (1,11) (5,4) (1,11) (5,9) (1,3) (1,1)

M       =   X       +       Y       +       2.7     ;
(5,9) (1,4) (5,8) (1,5) (5,4) (1,5) (4,1) (1,1)

ENS         ;
(2,7)   (1,1)
```

FRANCIS 語言所寫之程式，被轉換成記號的格式

如何抓Token。


A
Another character available?
No
End
Yes
Read a character.
Blank?
Yes
Identify the string in the queue, if it exists, as a token.
A
No
Delimiter?
Identify the string in the queue, if it exists, as a token.
No
Put this character into the queue.
Identify the delimiter as a token.
A
A

Lexical Analysis(Table 1)

1	;
2	(
3	)
4	=
5	+
6	-
7	*
8	/
9	↑
10	‘
11	’
12	:

Table 1 Delimiters

Lexical Analysis(Table 2)

1. AND
2. BOOLEAN
3. CALL
4. DIMENSION
5. ELSE
6. ENP
7. ENS
8. EQ
9. GE
10. GT
11. GTO
12. IF
13. INPUT
14. INTEGER
15. LABEL
16. LE
17. LT
18. NE
19. OR
20. OUTPUT
21. PROGRAM
22. REAL
23. SUBROUTINE
24. THEN
25. VARIABLE

Table 2 (Reserved Word Table)

Lexical Analysis(Table 3,4)

1	5
2	7

Table 3 (Integer Table)

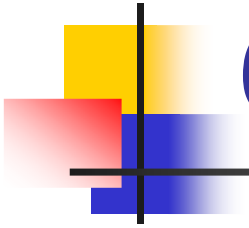
1	2.7

Table 4 (Real Number Table)

Lexical Analysis(Table 5)

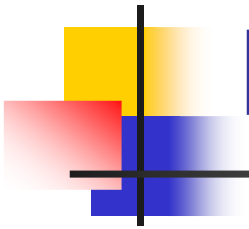
	Identifier	Subroutine	Type	Pointer
1	U	3		
2				
3	MAIN			
4	Y	10		
5	V	3		
6	M	3		
7				
8	X	10		
9	M	10		
10	S1			

Table 5 (Identifier Table)



6.2 Syntax Analysis

1. 將Tokens分辨爲Statement
2. 檢查文法
   - 正確
   - 不正確：Report error message
3. 將一些資訊匯入表格內

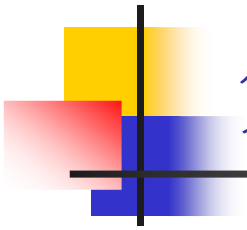


Example 4.1

```
      PROGRAM A1;
      VARIABLE INTEGER:X,Y,I;
      DIMENSION INTEGER:A(12);
      LABEL L91, L92;
      I=1;
      X=5;
      Y=11;
L91   IF  X GT Y  THEN  GTO L92  ELSE X=X+2;
      A(I)=X;
      I=I+1;
      GTO L91;
L92   ENP;
```

Quadruple form

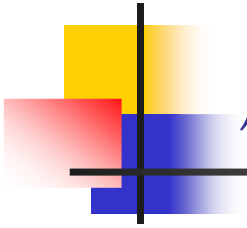
1	((5, 8),	,	)	X
2	((5,11),	,	)	Y
3	((5, 2),	,	)	I
4	((5, 7),	,	)	A
5	((5,14),	,	)	L91
6	((5,15),	,	)	L92
7	((1, 4),(3,2),	,(5,2))	I=1	
8	((1, 4),(3,3),	,(5,8))	X=5	
9	((1, 4),(3,4),	,(5,11))	Y=11	
10	((2,10),(5,8),(5,11),(0,1))		T1=X GT Y	
11	((2,12),(0,1),(6,12),(6,13))		IF T1 GO TO 12, ELSE GO TO 13	
12	((2,11),	,(6,18))	GTO L92	
13	((1, 5),(5,8),(3,5),(0,2))		T2=X+2	
14	((1, 4),(0,2),	,(5,8))	X=T2	
15	((1, 4),(5,8),(5,7),(5,2))		A(I)=X	
16	((1, 5),(5,2),(3,2),(5,2))		I=I+1	
17	((2,11),	,(6,10))	GTO L91	
18	((2,6),	,	)	L92 ENP



分辨Statement

- 以；當作Statement之結束
- Statement 計有

PROGRAM	GTO
SUBROUTINE	assignment
VARIABLE	CALL
LABEL	INPUT
DIMENSION	OUTPUT
IF	



處理PROGRAM(1)

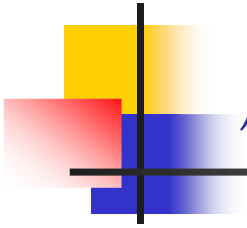
- 文法　　PROGRAM identifier;

  e.g.　　PROGRAM main;
- 代表程式的開頭
- 此程式的中間碼在哪裡

處理 PROGRAM(2)

- Pointer指向中間碼所在位置

	Identifier	Subroutine	Type	Pointer
1	U	3		
2				
3	MAIN			1
4	Y	10		
5	V	3		
6	M	3		
7				
8	X	10		
9	M	10		
10	S1			



處理變數宣告(1)

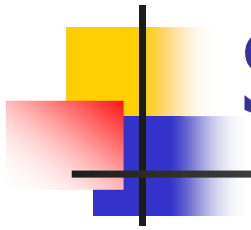
- VARIABLE INTEGER : U, V, M
- Data Type :

ARRAY	1
BOOLEAN	2
CHARACTER	3
INTEGER	4
LABEL	5
REAL	6

處理變數宣告(2)

- Subroutine 是哪個 routine 宣告的
- Type 是哪一種 Data Type

	Identifier	Subroutine	Type	Pointer
1	U	3	4	
2				
3	MAIN			
4	Y	10		
5	V	3		
6	M	3		
7				
8	X	10		
9	M	10		
10	S1			



Source

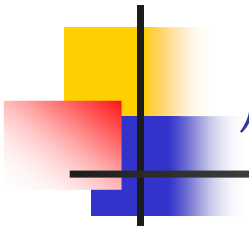
```
PROGRAM MAIN;
VARIABLE INTEGER:U,V,M;
U = 5;
V = 7;
CALL   S1(U ,V , M );
ENP;
SUBPOUTINE  S1(INTEGER:X,Y,M);
M = X + Y + 2.7;
ENS;
```

FRANCIS 語言所寫之程式

處理變數宣告(3)

- 產生中間碼Table 6

1 ((5,1), , ,)U

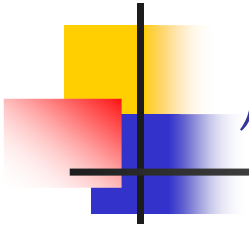


處理變數宣告範例(1)

```
VARIABLE INTEGER: U,V,M;

1  ((5,1),     ,     ,     )U
2  ((5,5),     ,     ,     )V
3  ((5,6),     ,     ,     )M
```

處理變數宣告範例(2)

	Identifier	Subroutine	Type	Pointer
1	U	3	4	
2				
3	MAIN			
4	Y	10		
5	V	3	4	
6	M	3	4	
7				
8	X	10		
9	M	10		
10	S1			



處理陣列宣告(1)

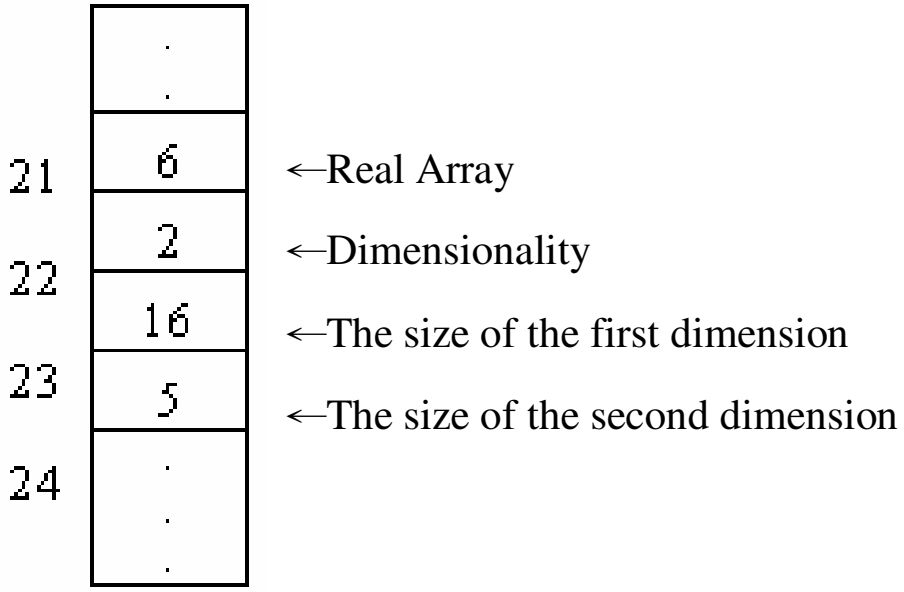
DIMENSION REAL A(16,5);

- 必須紀錄此陣列的大小
- 必須產生中間碼

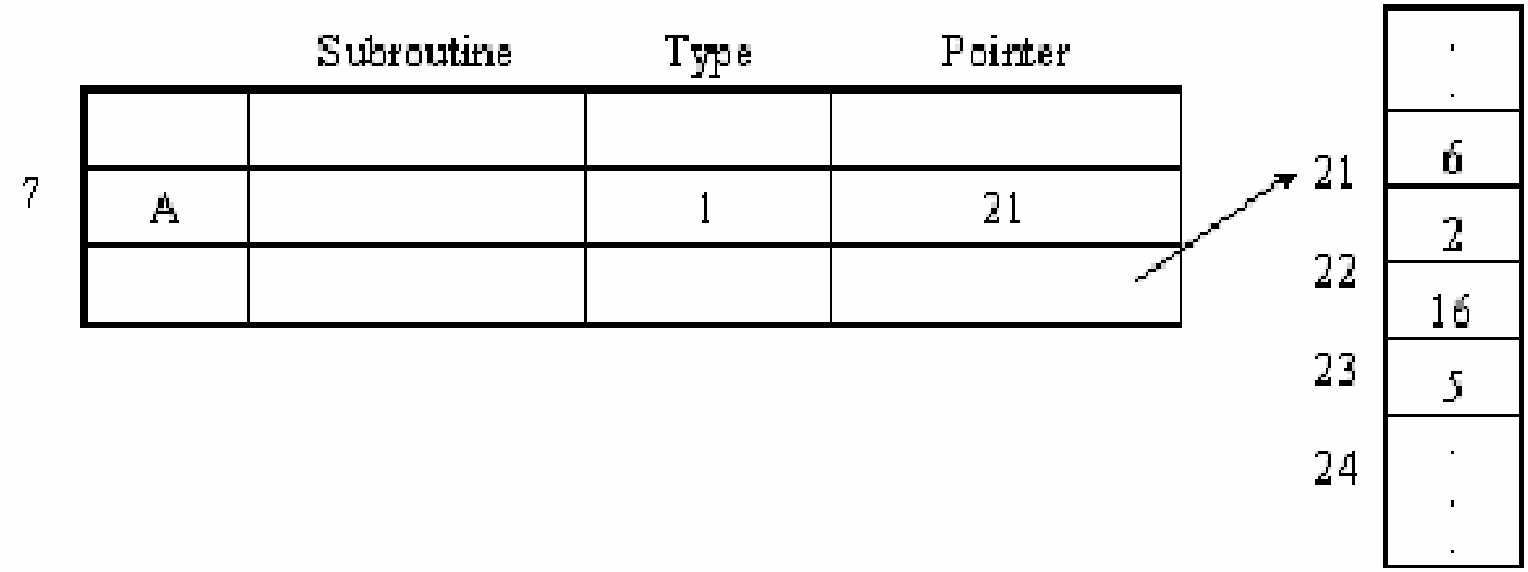
處理陣列宣告(2)

- 記錄陣列大小

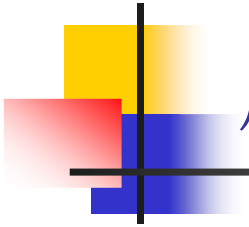
  使用Information Table 7



處理陣列宣告(2)



- TYPE 1 爲 ARRAY
- Pointer 21 指向information Table



處理陣列宣告(3)

- 產生中間碼

  ((5,7), , ,) A

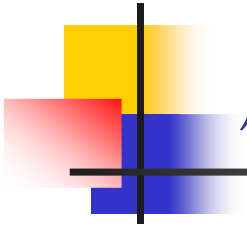
處理LABEL定義(1)

- 文法　LABEL identifier ;

  e.g　LABEL L91,L92;

		Subroutine	Type	Pointer
14	L91	5	5	
15	L92	5	5	

- Type 5 表示爲 LABEL



處理LABEL定義(2)

- 產生中間碼

((5,14), , ,) L91

((5,15), , ,) L92

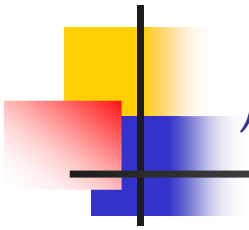
處理LABEL(1)

		Subroutine	Type	Pointer
14	L91	5	5	10
15	L92	5	5	18

- Pointer 指向要跳到的中間碼位置

處理LABEL(2)

```
                .
                .
              LABEL L91, L92;
                .
                .
L91           X =Y+2;
                .
                .
                .
L92           ENP;
```



處理LABEL(3)

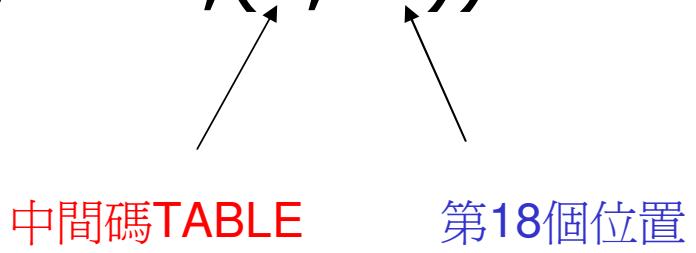
```
5   ((5,14),     ,     ,     )          L91
6   ((5,15),     ,     ,     )          L92
                  .
                  .
10  ((1,5),(5,3),(5,10),(5,13))   L91 X=Y+Z
                  .
                  .
                  .
18  ((2,6),       ,          ,       )   L92 ENP
```

處理GTO

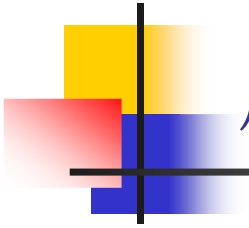
GTO L92

- 至Table 5內找到L92指向之中間碼位置
- 產生中間碼

((2,11), , ,(6,18)) GTO L92



6.3 Function Definition and Function Call



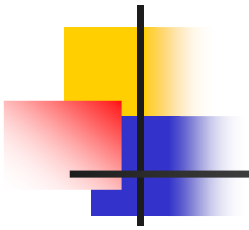
處理SUBROUTINE(1)

SUBROUTINE S1(INTEGER:X,Y,M);

- 可看做是PROGRAM及VARIABLE兩個部分
- 要在TABLE 5之POINTER指向中間碼位置(TABLE 6)
- 要在TABLE 5之SUBROUTINE加上X,Y,M的SCOPE

處理SUBROUTINE(2)

4	Y	10		
8	x	10		
9	M	10		
10	s1			20

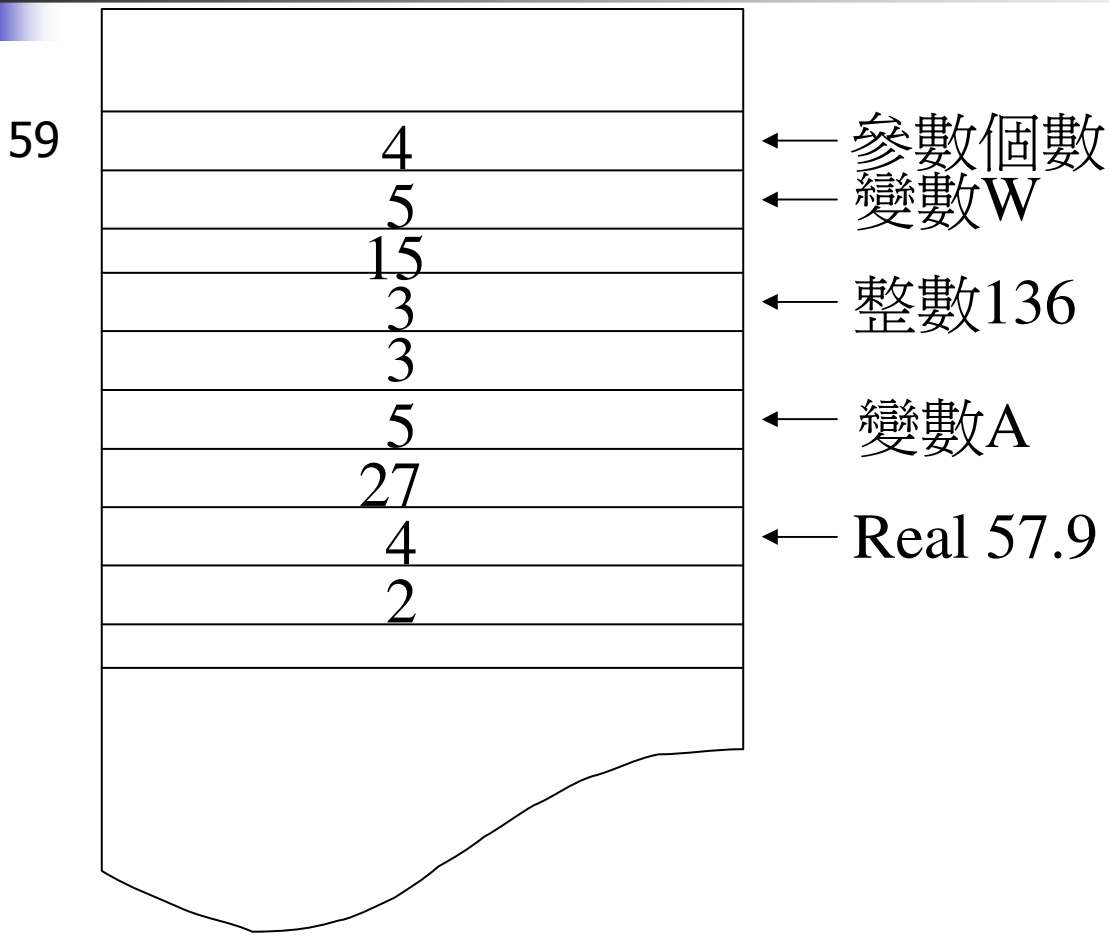


處理CALL(1)

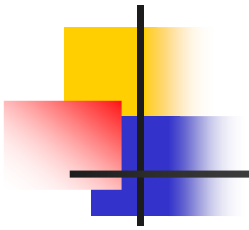
CALL S1(W, 136, A, 57.9);

- 將傳遞的參數建立在Information table
- 產生中間碼

處理CALL(2)



Information table 7



處理CALL(3)

((2 , 3) , (5 , 10),　　,(7 , 59))

CALL　　S1

處理CALL(4)

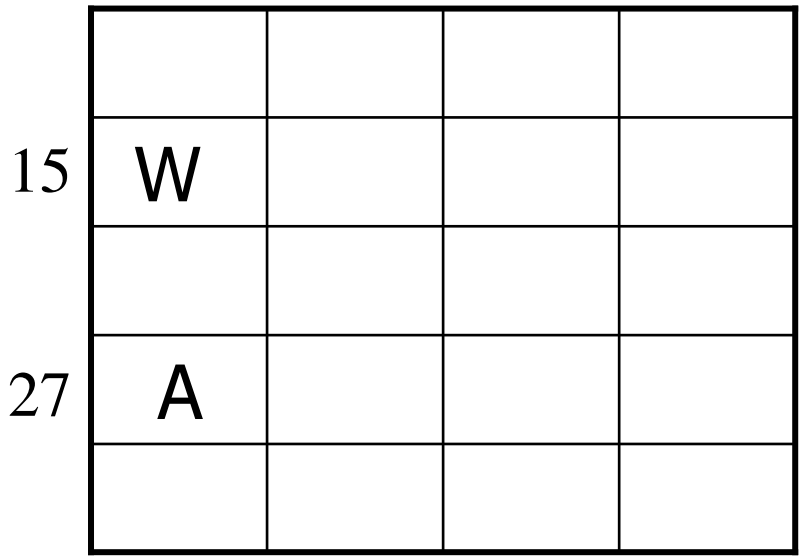


Table 5

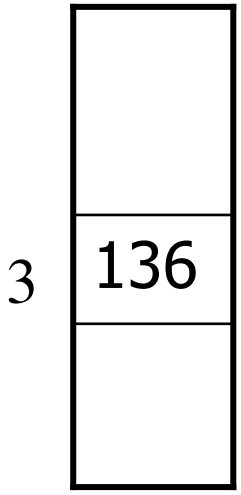


Table 3

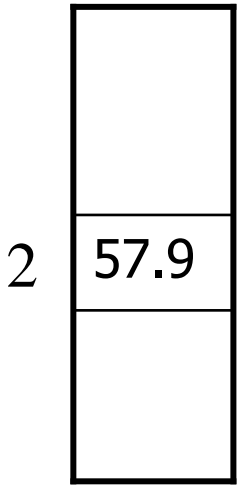
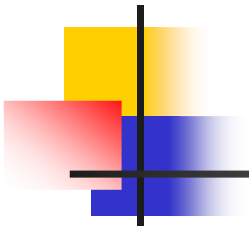


Table 4



6.4 Assignment
處理Assignment(1)

- X=Y+Z ;

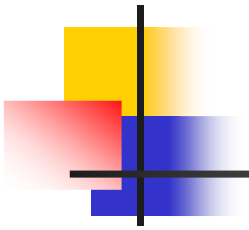
  (+ , Y , Z , X)

  ((1,5) , (5,3) (5,10) , (5,13))

- X=Y+U*V ;

  (* , U , V , T1)

  (+ , Y , T1 , X)



處理Assignment(2)

- 使用Reverse Polish Notation

  ↑> ＊,÷ > ＋,－ > (,) >=

- 可以透過stack來進行處理

Reverse Polish Notation(1)

Input X=Y+U*V

Operand stack

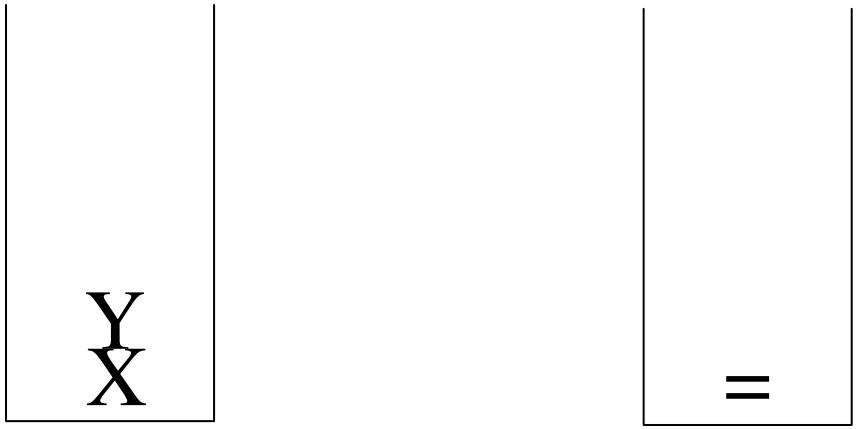
Operator stack

Reverse Polish Notation(2)

Input : Y+U*V

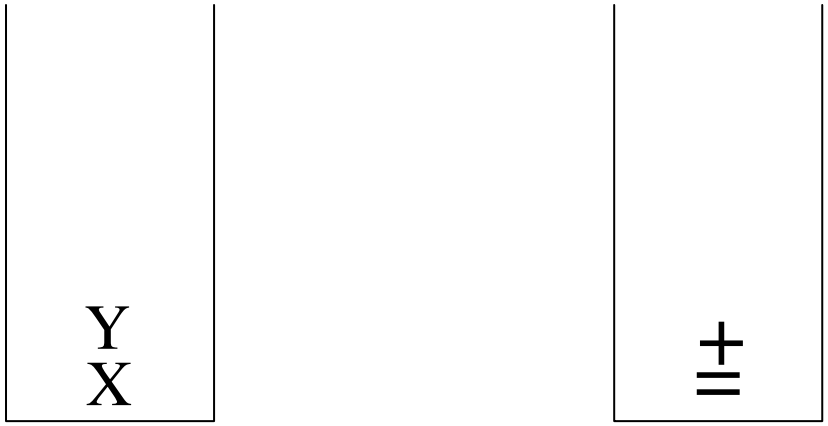
Reverse Polish Notation(3)

Input : +U*V



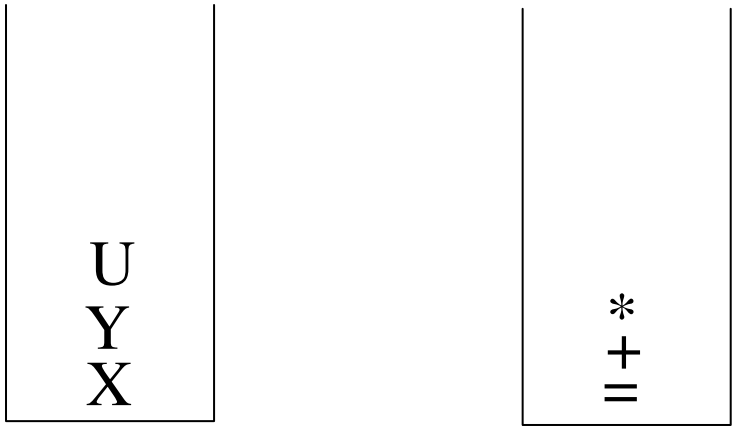
Reverse Polish Notation(4)

Input : U*V

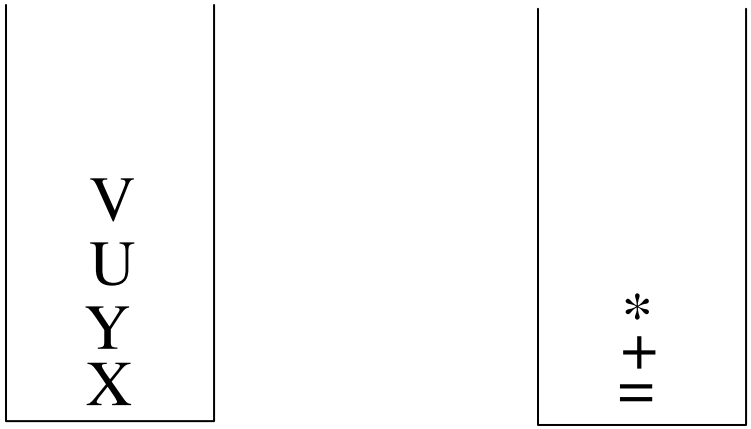


Reverse Polish Notation(5)

Input : V



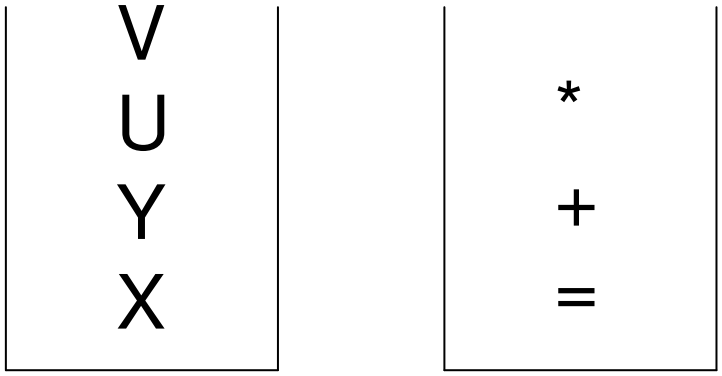
Reverse Polish Notation(6)



Reverse Polish Notation(7)

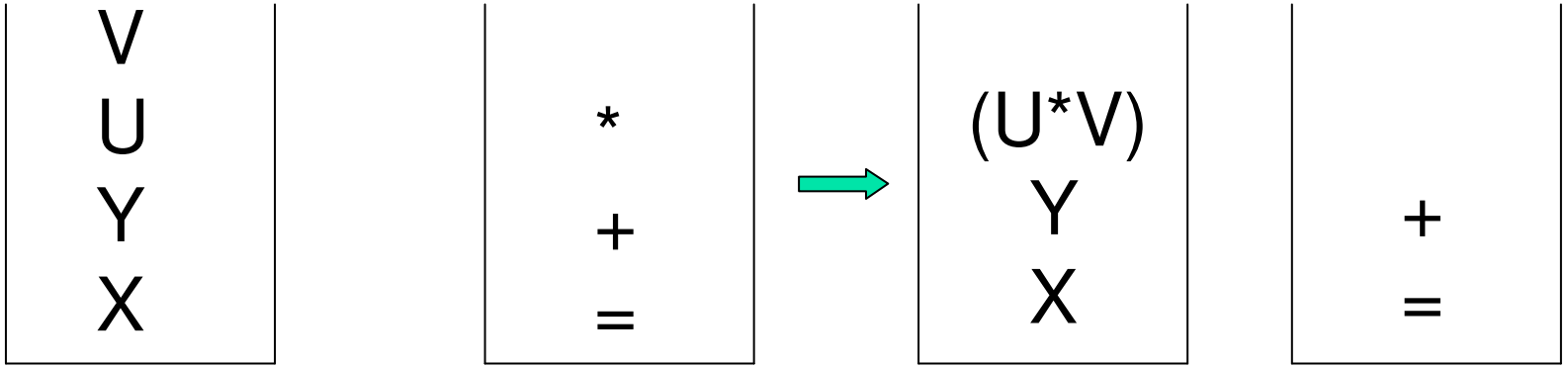
- 得到Reverse Polish Notation

XYUV*+=



Reverse Polish Notation(8)

X=(Y+(U*V))



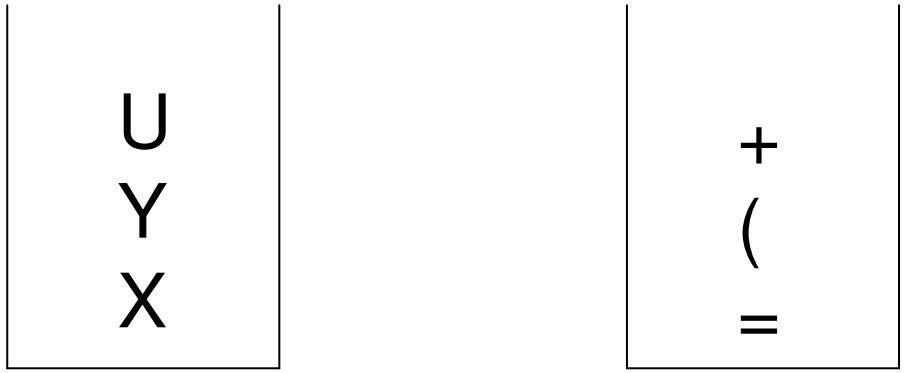
Example(1)

input X=(Y+U)*V

(+,Y,U, T1)

(*,T1,V,T2)

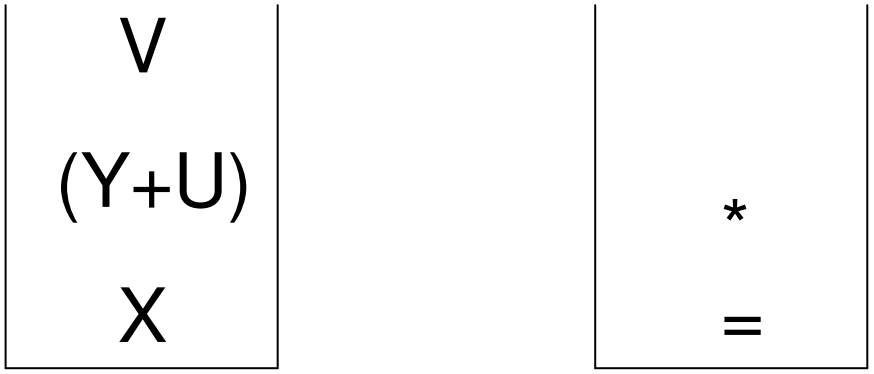
(=,T2, , X)



Example(2)

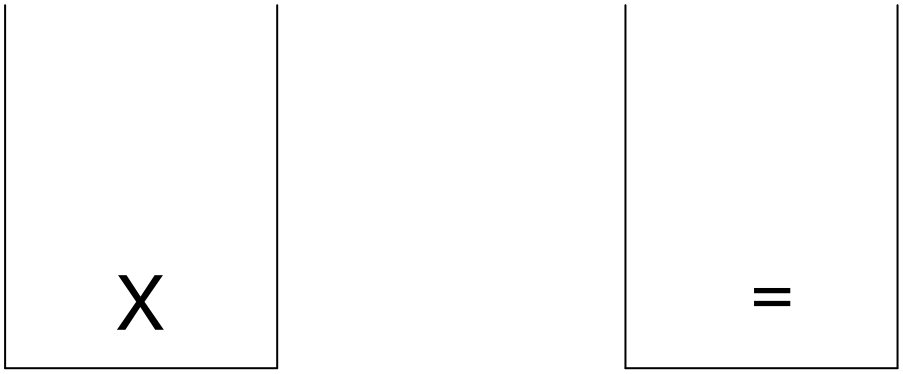
input X=(Y+U)*V

output (Y+U)



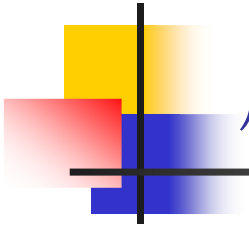
Example(3)

output (Y+U)*V



6.5 IF Statement
處理IF(1)

```
IF X GT Y AND Q THEN X=X+1 ELSE X=X+2;
14          (GT,X,Y,T4)
15          (AND,T4,Q,T1)
16          (IF,T1,(6,17),(6,20))
17          (+,X,1,T2)      <-------  X=X+1
18          (=,T2,      ,X)
19          (GTO,       ,   ,(6,22))
20          (+,X,2,T3)      <-------  X=X+2
21          (=,T3,      ,X)
```



處理IF(2)

IF XGT Y AND Q THEN X=X+1 ELSE X=X+2

- 條件部分用Reverse Polish Notation
- Statement則參照各statement處理

6.6 Array and Elements
處理有陣列元素之statement(1)

- 算出元素所在位置
- 產生中間碼

X=B(I,J)+4;

處理有陣列元素之statement(2)

- B(I,J)元素所在位置
  (J-1)*M+I
- 中間碼

(-,J,1,T1)	T1=J-1
(*,T1,M,T2)	T2=T1*M
(+,T2,I,T3)	T3=T2+I
(=,B,T3,T4)	T4=B(T3)
(+,T4,4,T5)	T5=T4+4
(=,T5, ,X)	X=T5

6.7 Examples

Example 4.1

- Source

```
      PROGRAM A1;
      VARIABLE INTEGER:X,Y,I;
      DIMENSION INTEGER:A(12);
      LABEL L91, L92;
      I=1;
      X=5;
      Y=11;
L91   IF  X GT Y  THEN  GTO L92  ELSE X=X+2;
      A(I)=X;
      I=I+1;
      GTO L91;
L92   ENP;
```

Example 4.1 (Table)

			TYPE	POINTER	
1					
2	I	5	4		
3					
4					
5	A1			1	→ (Quadruple Table)
6					
7	A	5	1	1	→ (Information Table)
8	X	5	4		
9					
10					
11	Y	5	4		
12					
13					
14	L91	5	5	10	→ (Quadruple Table)
15	L92	5	5	18	(Quadruple Table)

Identifier Table (Table5)

1	12
2	1
3	5
4	11
5	2

Integer Table (Table3)

1	4	} Array A
2	1	
3	12	

Information Table (Table 7)

Example 4.1(中間碼)

```
1   (( 5, 8),    ,      ,    )   X
2   (( 5,11),,     ,    )   Y
3   (( 5, 2), ,    ,    )   I
4   (( 5, 7), ,    ,    )   A
5   (( 5,14), ,    ,    )   L91
6   (( 5,15), ,    ,    )   L92
7   (( 1, 4),(3,2),     ,(5,2))   I=1
8   (( 1, 4),(3,3),     ,(5,8))   X=5
9   (( 1, 4),(3,4),     ,(5,11))  Y=11
10  (( 2,10),(5,8),(5,11),(0,1))      T1=X GT Y
11  (( 2,12),(0,1),(6,12),(6,13))     IF T1 GO TO 12, ELSE GO TO 13
12  (( 2,11),,      ,(6,18))  GTO L92
13  (( 1, 5),(5,8),(3,5),(0,2))   T2=X+2
14  (( 1, 4),(0,2),     ,(5,8))   X=T2
15  (( 1, 4),(5,8),(5,7),(5,2))   A(I)=X
16  (( 1, 5),(5,2),(3,2),(5,2))   I=I+1
17  (( 2,11),,      ,(6,10))  GTO L91
18  ((2,6),   ,     ,    )    L92 ENP
    Quadruple Table (Table 6)
```

Example 4.2

- Source

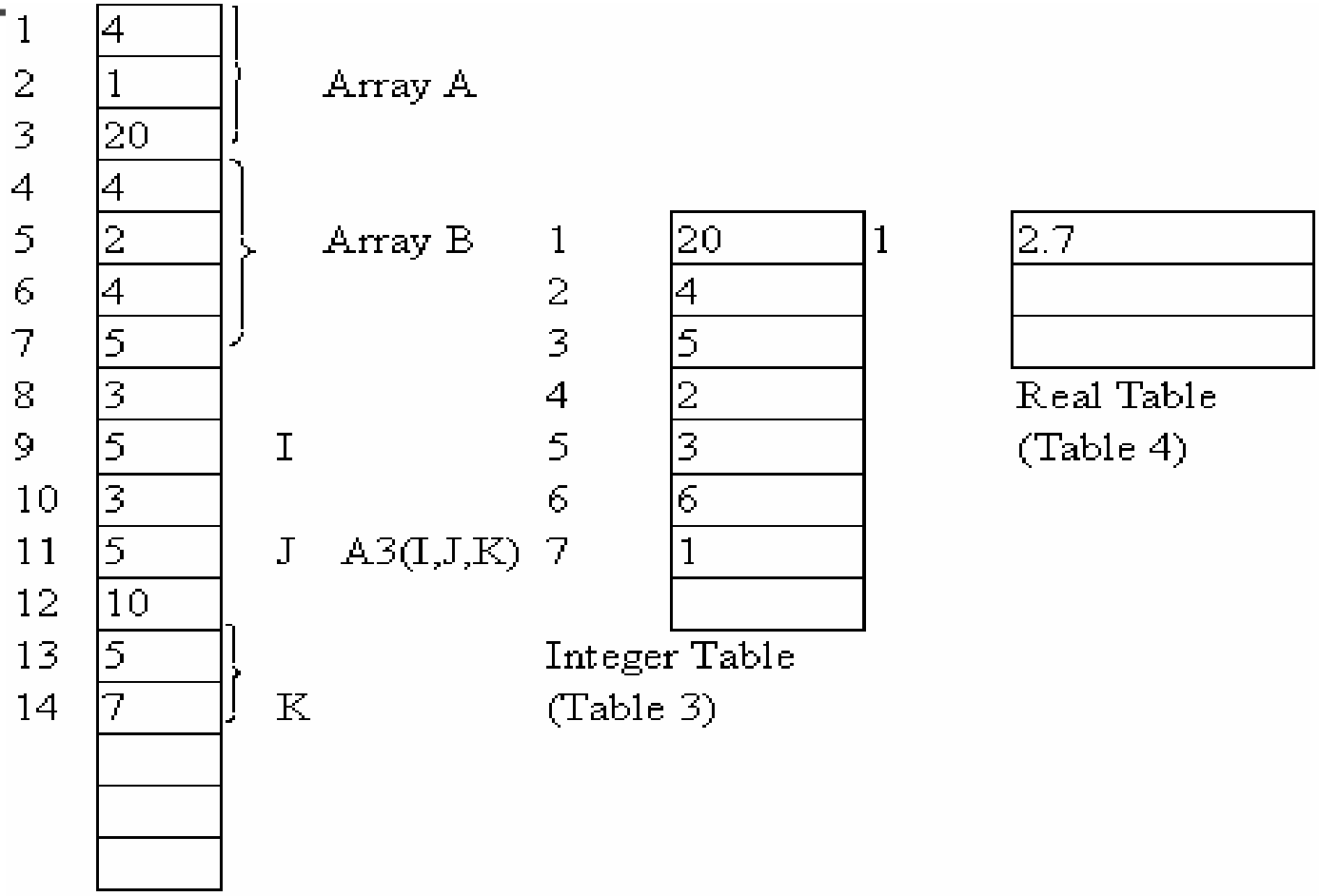
```
PROGRAM A2;
VARIABLE INTEGER: I,J,K;
DIMENSION INTEGER:  A(20),B(4,5);
I=2;
J=3;
CALL A3(I,J,K);
A(K)=B(I,J)+2.7;
ENP;
SUBROUTINE A3(INTEGER:X,Y,K) ;
VARIABLE INTEGER:Z
Z=6;
K=(X-Z) ↑ 2+Y;
ENS;
```

Example 4.2(Table)

			Type	Pointer	
1					
2	X	11	4		
3	I	6	4		
4					
5	A	6	1	1	→ (Information Table)
6	A2			1	→ (Quadruple Table)
7	K	6	4		
8	K	11	4		
9	B	6	1	4	→ (Information Table)
10	J	6	4		
11	A3			16	→ (Quadruple Table)
12					
13	Y	11	4		
14					
15	Z	11	4		
16					

Identifier Table (Table 5)

Example 4.2(Table)



Example 4.2(中間碼)

Quadruples:

```
1   ((5,3),   ,   ,   )   I
2   ((5,10),  ,   ,   )   J
3   ((5,7),   ,   ,   ,   )   K
4   ((5,5),   ,   ,   ,   )   A
5   ((5,9),   ,   ,   ,   )   B
6   ((1,4),(3,4),  ,(5,3))   I=2
7   ((1,4),(3,5),  ,(5,10)) J=3
8   ((2,3),(5,11),,(7,8))   CALL A3(I,J,K)
9   ((1,6),(5,10),(3,7),(0,1))   T1=J-1
10  ((1,7),(0,1),(3,2),(0,2))    T2=T1*4
11  ((1,5),(5,3),(0,2),(0,3))    T3=I+T2
12  ((1,4),(5,9),(0,3),(0,4))    T4=B(T3)
13  ((1,5),(0,4),(4,1),(0,5))    T5=T4+2.7
14  ((1,4),(0,5),(5,5),(5,7))    A(K)=T5
15  ((2,6),   ,   ,   )   ENP
16  ((5,2),   ,   ,   )   X
17  ((5,13),  ,   ,   )   Y
18  ((5,8),       ,   ,   )   K
19  ((5,15),  ,   ,   )   Z
20  ((1,4),(3,6),  ,(5,15)) Z=6
21  ((1,6),(5,2),(5,15),(0,6))   T6=X-Z
22  ((1,9),(0,6),(3,4),(0,7))    T7=T6↑2
23  ((1,5),(0,7),(5,13),(0,8))   T8=T7+Y
24  ((1,4),(0,8),  ,(5,8))   K=T8
25  ((2,7),       ,   ,   )   ENS
```

Program A2

Subroutine A3