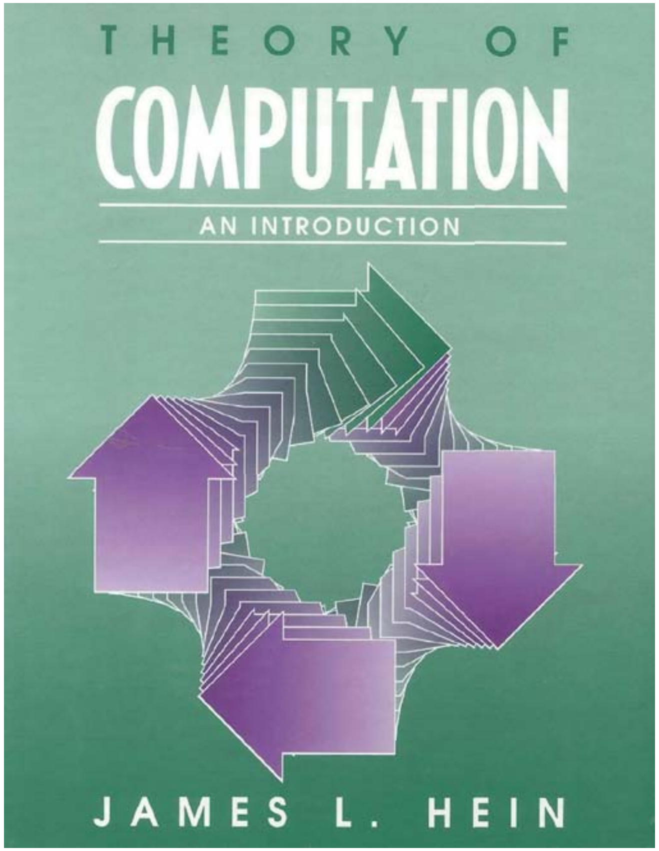
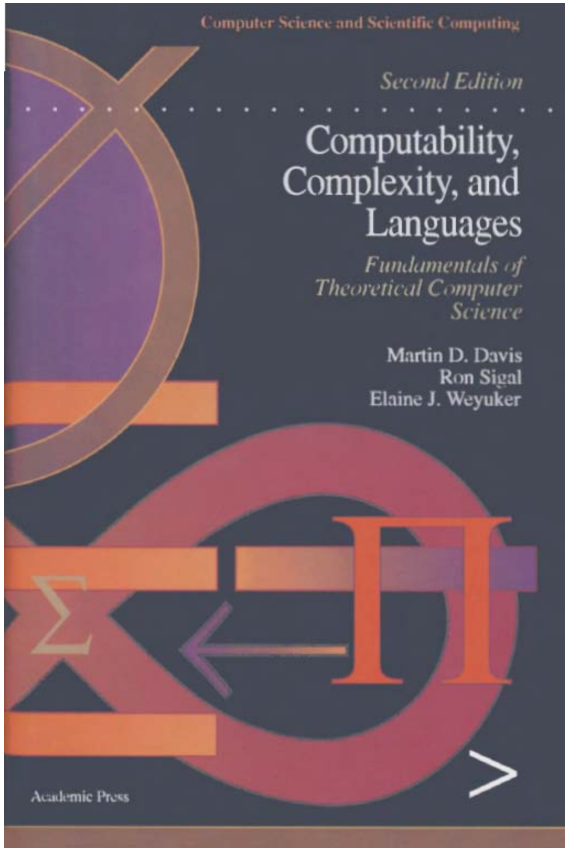
Theory of Computation

Lecture 03

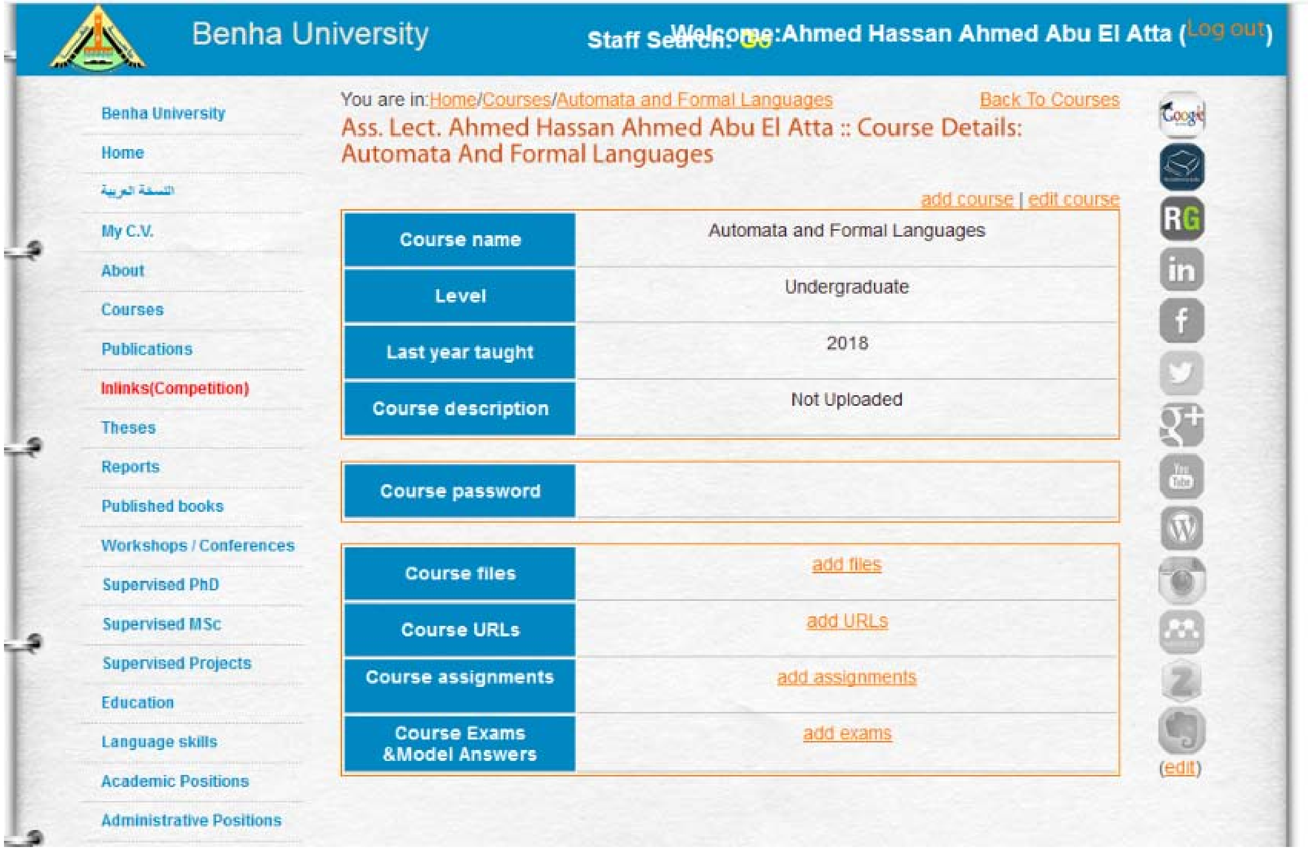
Books


THEORY OF
COMPUTATION
AN INTRODUCTION
JAMES L. HEIN


Computer Science and Scientific Computing
Second Edition
Computability, Complexity, and Languages
Fundamentals of Theoretical Computer Science
Martin D. Davis
Ron Sigal
Elaine J. Weyuker
Academic Press

PowerPoint

http://www.bu.edu.eg/staff/ahmedaboalatah14-courses/14767



You are in:Home/Courses/Automata and Formal Languages

Back To Courses

Ass. Lect. Ahmed Hassan Ahmed Abu El Atta :: Course Details: Automata And Formal Languages

add course | edit course

Course name	Automata and Formal Languages
Level	Undergraduate
Last year taught	2018
Course description	Not Uploaded

Course password	

Course files	add files
Course URLs	add URLs
Course assignments	add assignments
Course Exams &Model Answers	add exams

Programs and Computable Functions

SIMPLE LANGUAGE

Agenda

- A Programming Language
- Some Examples of Programs
- Macro & Macro Expansion

A Programming Language

Our development of computability theory will be based on a specific programming language $\mathscr{S}$.

We will use certain letters as variables whose values are numbers.

In this programming language $\mathscr{S}$ The word number will always mean nonnegative integer, unless the contrary is specifically stated.

Variables

The letters

$$X_1 \ X_2 \ X_3 \ \cdots$$

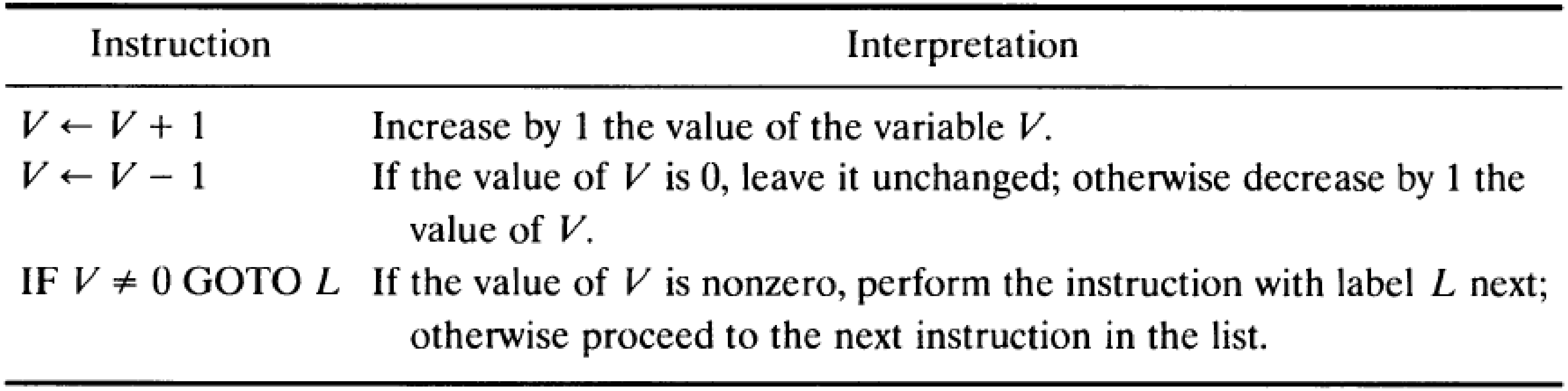
will be called the *input variables of* $\mathscr{S}$*,* the letter **Y** will be called the output variable of $\mathscr{S}$.

The letters

$$Z_1 \ Z_2 \ Z_3 \ \cdots$$

will be called the *local variables of*

Instructions



Instruction	Interpretation
$V \leftarrow V + 1$	Increase by 1 the value of the variable V.
$V \leftarrow V - 1$	If the value of V is 0, leave it unchanged; otherwise decrease by 1 the value of V.
IF $V \neq 0$ GOTO L	If the value of V is nonzero, perform the instruction with label L next; otherwise proceed to the next instruction in the list.

Program

A "program" of $\mathscr{S}$ will then consist of a list *(i.e., a finite sequence)* of instructions.

A simple example of a program of *.9' is*

$$X \leftarrow X+ 1$$

$$X \leftarrow X+1$$

"Execution" of this program has the effect of increasing the value of *X by* 2.

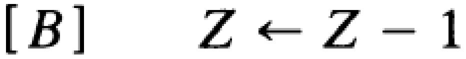
Labels

$$A_1 \;\; B_1 \;\; C_1 \;\; D_1 \;\; E_1 \;\; A_2 \;\; B_2 \;\; C_2 \;\; D_2 \;\; E_2 \;\; A_3 \;\; \cdots$$

Initial Values

We will use the special convention that *the output variable* **Y** *and the local variables* $\mathbf{Z}_i$ *initially have the value* **0**.

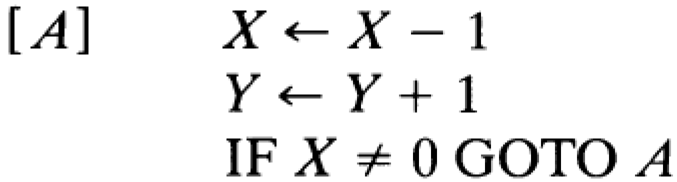
Instructions may or may not have labels. When an instruction is labeled, the label is written to its left in square brackets. For example,



$$[B] \qquad Z \leftarrow Z - 1$$

Some Examples of Programs

Our first example is the program

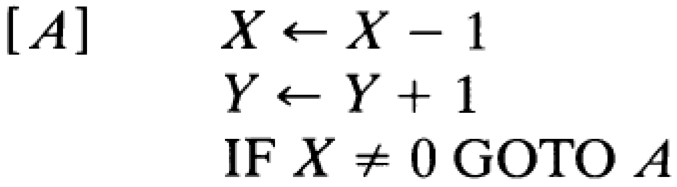


```
[A]    $X \leftarrow X - 1$
       $Y \leftarrow Y + 1$
       IF $X \neq 0$ GOTO $A$
```

Some Examples of Programs

Our first example is the program

- When an attempt is made to move on to the nonexistent fourth instruction, the program halts.
- A program will also halt if an instruction labeled *L is to be executed, but there is no* instruction in the program with that label.
- In this case, we usually will use the letter *E (for "exit") as the label which labels no instruction.*



```
[$A$]    $X \leftarrow X - 1$
         $Y \leftarrow Y + 1$
         IF $X \neq 0$ GOTO $A$
```

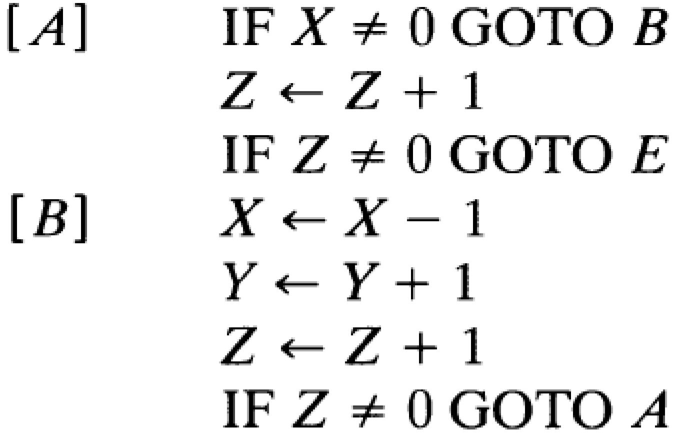
Some Examples of Programs

Our first example is the program

```
[A]    $X \leftarrow X - 1$
       $Y \leftarrow Y + 1$
       IF $X \neq 0$ GOTO $A$
```

$$f(x) = \begin{cases} 1 & \text{if} \quad x = 0 \\ x & \text{otherwise}. \end{cases}$$

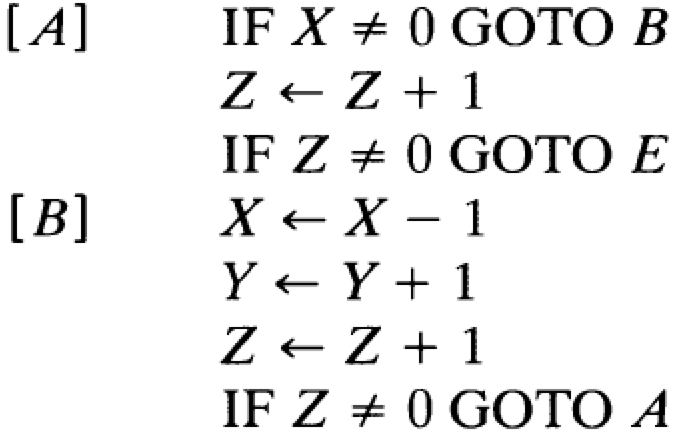
Some Examples of Programs



```
[A]     IF $X \neq 0$ GOTO $B$
        $Z \leftarrow Z + 1$
        IF $Z \neq 0$ GOTO $E$
[B]     $X \leftarrow X - 1$
        $Y \leftarrow Y + 1$
        $Z \leftarrow Z + 1$
        IF $Z \neq 0$ GOTO $A$
```

Some Examples of Programs

$$f(x) = x$$



```
[A]    IF $X \neq 0$ GOTO $B$
       $Z \leftarrow Z + 1$
       IF $Z \neq 0$ GOTO $E$
[B]    $X \leftarrow X - 1$
       $Y \leftarrow Y + 1$
       $Z \leftarrow Z + 1$
       IF $Z \neq 0$ GOTO $A$
```

Macro & Macro Expansion

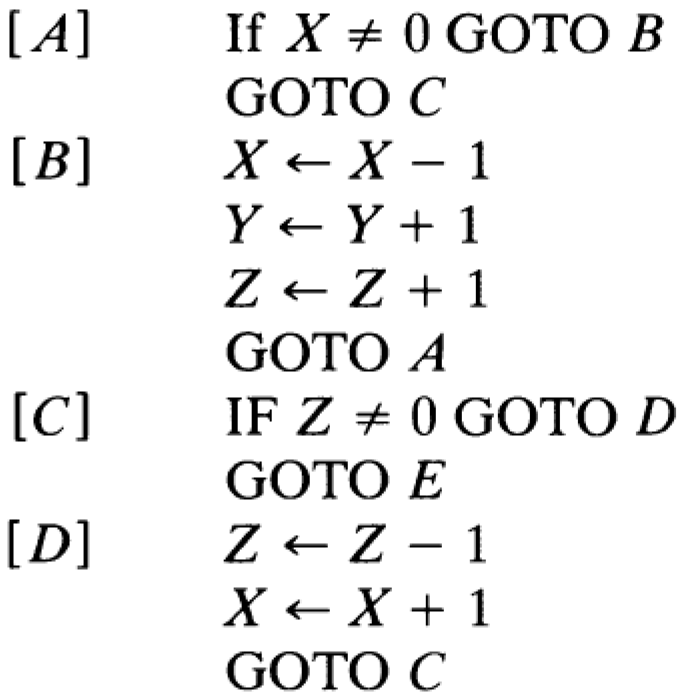
MACRO	MACRO EXPANSION
GOTO L	$Z \leftarrow Z + 1$ IF $Z \neq 0$ GOTO L

Copy X into Y

The value of *X is "destroyed" and the program* terminates with *X having the value 0.*

Copy X into Y

The value of *X is "destroyed" and the program* terminates with *X having the value 0.*



```
[$A$]    If $X \neq 0$ GOTO $B$
         GOTO $C$
[$B$]    $X \leftarrow X - 1$
         $Y \leftarrow Y + 1$
         $Z \leftarrow Z + 1$
         GOTO $A$
[$C$]    IF $Z \neq 0$ GOTO $D$
         GOTO $E$
[$D$]    $Z \leftarrow Z - 1$
         $X \leftarrow X + 1$
         GOTO $C$
```

Macro & Macro Expansion

MACRO

$$V \leftarrow 0$$

MACRO EXPANSION

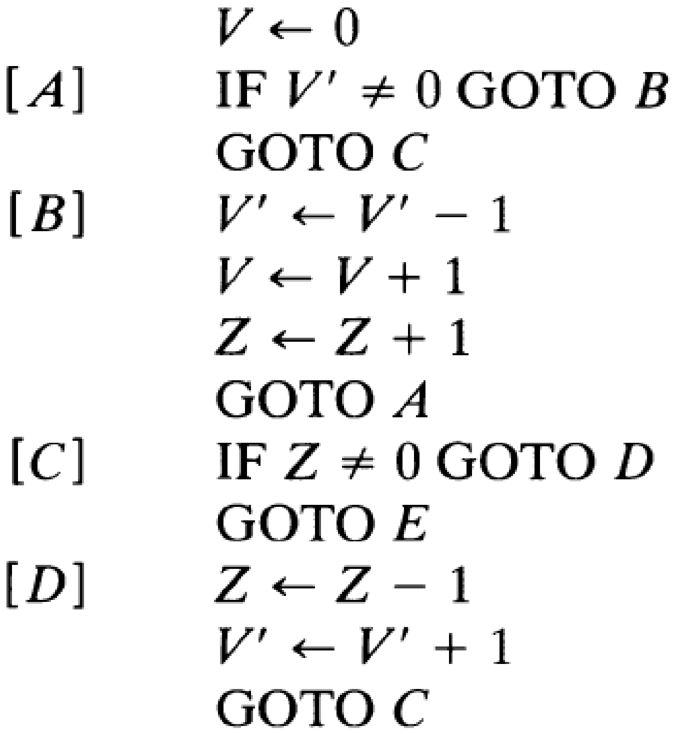
```
[$L$]    $V \leftarrow V - 1$
         IF $V \neq 0$ GOTO $L$
```

Macro & Macro Expansion

MACRO

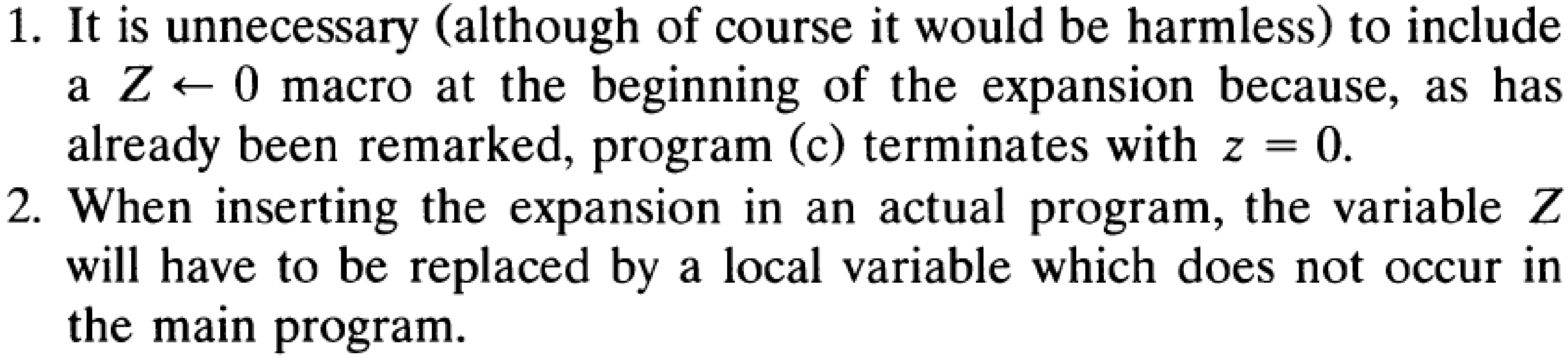
$V \leftarrow V'$

MACRO EXPANSION

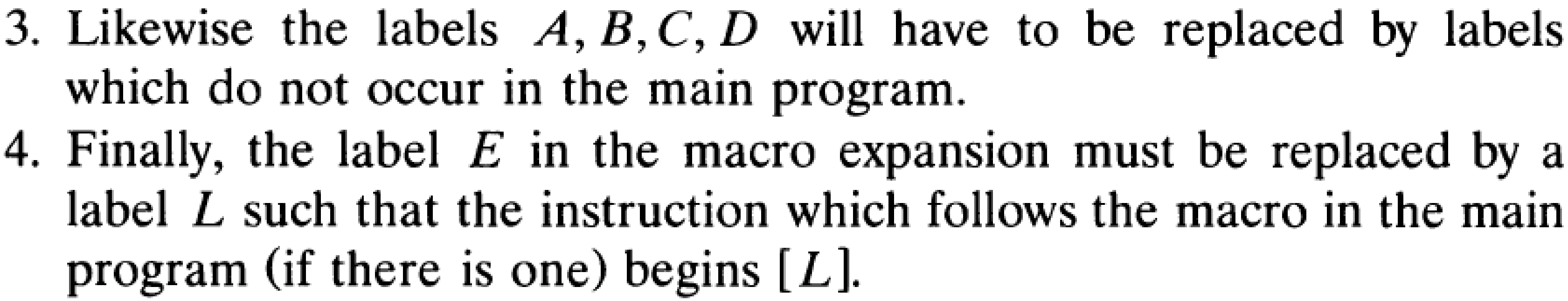


```
        $V \leftarrow 0$
[A]     IF $V' \neq 0$ GOTO $B$
        GOTO $C$
[B]     $V' \leftarrow V' - 1$
        $V \leftarrow V + 1$
        $Z \leftarrow Z + 1$
        GOTO $A$
[C]     IF $Z \neq 0$ GOTO $D$
        GOTO $E$
[D]     $Z \leftarrow Z - 1$
        $V' \leftarrow V' + 1$
        GOTO $C$
```

Notes



1. It is unnecessary (although of course it would be harmless) to include a $Z \leftarrow 0$ macro at the beginning of the expansion because, as has already been remarked, program (c) terminates with $z = 0$.
2. When inserting the expansion in an actual program, the variable Z will have to be replaced by a local variable which does not occur in the main program.



3. Likewise the labels A, B, C, D will have to be replaced by labels which do not occur in the main program.
4. Finally, the label E in the macro expansion must be replaced by a label L such that the instruction which follows the macro in the main program (if there is one) begins $[L]$.

Add Two Variables

$$f(x_1, x_2) = x_1 + x_2$$

Add Two Variables

$f(x_1, x_2) = x_1 + x_2$

```
        $Y \leftarrow X_1$
        $Z \leftarrow X_2$
[B]     IF $Z \neq 0$ GOTO $A$
        GOTO $E$
[A]     $Z \leftarrow Z - 1$
        $Y \leftarrow Y + 1$
        GOTO $B$
```