

第七章 中间代码生成

1、写出 if A and B and C > D then

if A < B then F:=1

else F:=0

else G:=G+1; 的四元式序列, 翻译过程中, 采用 then 与 else 的最近匹配原则。

答:

- | | | |
|------|----------------|--------------------|
| (1) | (jnz, A, _, 3) | //当 A 为真时跳转到地址 3 |
| (2) | (j, _, _, 13) | //当 A 为假时跳转到地址 13 |
| (3) | (jnz, B, _, 5) | //当 B 为真时跳转到地址 5 |
| (4) | (j, _, _, 13) | //当 B 为假跳转到地址 13 |
| (5) | (j>, C, D, 7) | //当 C>D 时跳转到地址 7 |
| (6) | (j, _, _, 13) | //当 C<=D 时跳转到地址 13 |
| (7) | (j<, A, B, 9) | //当 A<B 时跳转到地址 9 |
| (8) | (j, _, _, 11) | //当 A>=B 时跳转到地址 11 |
| (9) | (:=, 1, _, F) | //F:=1 |
| (10) | (j, _, _, 15) | |
| (11) | (:=, 0, _, F) | //F:=0 |
| (12) | (j, _, _, 15) | |
| (13) | (+, G, 1, T) | //计算 G+1 的值 |
| (14) | (:=, T, _, G) | //将计算结果赋给 G |
| (15) | | |

2、写出 WHILE A<C AND B<D DO

IF A=1 THEN C:=C+1 ELSE

WHILE A<=D DO A:=A+2; 的四元式序列。

答:

- | | | |
|------|-----------------|--------------------|
| (1) | (j<, A, C, 3) | //当 A<C 时跳转到地址 3 |
| (2) | (j, _, _, 15) | //当 A>=C 时跳转到地址 15 |
| (3) | (j<, B, D, 5) | //当 B<D 时跳转到地址 5 |
| (4) | (j, _, _, 15) | //当 B>=D 时跳转到地址 15 |
| (5) | (j=, A, 1, 7) | //当 A=1 时跳转到地址 7 |
| (6) | (j, _, _, 10) | //当 A!=1 时跳转到地址 10 |
| (7) | (+, C, 1, T) | //计算 C+1 的值 |
| (8) | (:=, T, _, C) | //将结果赋给 C |
| (9) | (j, _, _, 1) | //继续外层循环 |
| (10) | (j<=, A, D, 12) | //当 A<=D 时跳转到地址 12 |
| (11) | (j, _, _, 1) | //继续外层循环 |
| (12) | (+, A, 2, T) | //计算 A+2 的值 |
| (13) | (:=, T, _, A) | //将计算结果赋给 A |
| (14) | (j, _, _, 10) | //继续内层循环 |
| (15) | | |