

第十章 目标代码生成

1. 写出下面语句列的带寄存器追踪的代码生成表，可分配寄存器有两个。

$G := C*(A+B)+(A+B);$

$C := A+B;$

$A := (C*D)+(E-F);$

(答案)

中间代码为		R ₁	R ₂
(+, A, B, T ₁)	(+, A, B, T ₁)		
(*, C, T ₁ , T ₂)	● Load A, R ₁	A(L, NS)	
(+, T ₂ , T ₁ , T ₃)	● ADD B, R ₁ [A不需保存]	T ₁ (L, S)	
(=, T ₃ , -, G)	(*, C, T ₁ , T ₂)		
(=, T ₁ , -, C)	● Load C, R ₂	T ₁ (L, S)	C(L, NS)
(*, C, D, T ₄)	● MULT R ₁ , R ₂ [C不需保存]	T ₁ (L, S)	T ₂ (L, S)
(-, E, F, T ₅)	(+, T ₂ , T ₁ , T ₃)		T ₃ (L, S)
(+, T ₄ , T ₅ , T ₆)	● ADD R ₁ , R ₂ [T ₂ 不需保存]	T ₁ (L, S)	
(=, T ₆ , -, A)	(=, T ₃ , -, G)		
	● STORE R ₂ , G	T ₁ (L, S)	T ₃ (D, NS)
	(=, T ₁ , -, C)		
	推迟 T ₁ 的 STORE	C(L, S)	
	(*, C, D, T ₄)		
	● Load D, R ₂	C(L, S)	D(L, NS)
	● MULT R ₁ , R ₂ [D不需保存]	C(D, S)	T ₄ (L, S)
	● STORE R ₁ , C		
	(-, E, F, T ₅)		
	● Load E, R ₁	E(L, NS)	T ₄ (L, S)
	● MIN F, R ₁	T ₅ (L, S)	T ₄ (L, S)
	(+, T ₄ , T ₅ , T ₆)		
	● ADD R ₂ , R ₁ [T ₅ 不需保存]	T ₆ (L, S)	T ₄ (D, NS)
	(=, T ₆ , -, A)		
	● STORE R ₁ , A		

(关闭)

2. 设有下面基本块，试写出各临时变量的活动区间。

(+,X,1,T1)
(-,A,T1,T2)
(*,Y,T2,T3)
(-,T3,T1,T4)
(=>,T3,Y)
(*,T3,T4,T5)
(=>,T5,Z)

(答案)

1:(+,X,1,T1)	临时变量活动区间:
2:(-,A,T1,T2)	T1:[1,4]
3:(*,Y,T2,T3)	T2:[2,3]
4:(-,T3,T1,T4)	T3:[3,6]
5:(=>,T3,Y)	T4:[4,6]
6:(*,T3,T4,T5)	T5:[6,7]
7:(=>,T5,Z)	

(关闭)

3. 设有下面带间接地址的目标机程序，试写出与此等价但没有间接地址的目标机程序(其中M表示绝对地址)。

Load R1, *3[R0]
Add R1, R1
Load R2, *M
Mult R2, R1
Store R2, 5[R3]

(答案)

Load R1, 3[R0]
Load R1, 0[R1]
ADD R1, R1
Load R2, 0[M]
Mult R2, R1
Store R2, 5[R3]

(关闭)

4. 试写出下面函数声明和函数调用 $f(f(a[i]))$ 的目标代码.

```
function fac(x: integer):integer;  
begin  
    if x=0 then fac:=1 else fac:=fac(x-1)*x  
end
```

[\(答案\)](#)

(无)

[\(关闭\)](#)

5. 试写出下面语句的目标代码(其中变量均为实型变量, 且 y 是引用形参).

if $x > 0$ then $y := y + a$ else if $x < 0$ then $z := y + a$

[\(答案\)](#)

($>$, x , 0, T_1)	l_1	LOAD R_1, x
(then, -, -, -)	l_2	MIN $R_1, 0$
(+, y , a , T_2)	l_3	JMP R_1, l_9
(=, T_2 , 0, y)	l_4	LOAD R_1, y
(else, -, -, -)	l_5	LOAD R_2, a
(<, x , 0, T_3)	l_6	ADD R_1, R_2
(then, -, -, -)	l_7	STORE y, R_1
(+, y , a , T_4)	l_8	JMP l_{16}
(=, T_4 , -, z)	l_9	LOAD R_1, x
(ifend, -, -, -)	l_{10}	MIN $R_1, 0$
(ifend, -, -, -)	l_{11}	JMP R_1, l_{15}
	l_{12}	LOAD R_1, y
	l_{13}	LOAD R_2, a
	l_{14}	ADD R_1, R_2
	l_{14}	STORE z, R_1
	l_{15}	JMP l_{16}

[\(关闭\)](#)

6. 试写出下面语句的目标代码, 其中变量的假定同前(无全局变量)。

```
while x<y do
begin
y:=y-1;
if y>0 then y:=y-x else while y<0 do y:=y+x
end
```

(答案)

```
l1 LOAD R1,x
l2 LOAD R2,y
l3 CMP R1,R2
l4 CJ>= l26
l5 LOAD R1,y
l6 LOAD R2,1
l7 SUB R1,R2
l8 STORE y,R2
l9 LOAD R1,y
l10 CMP R1,0
l11 CJ<= l17
l12 LOAD R1,y
l13 LOAD R2,x
l14 SUB R1,R2
l15 STORE y,R2
l16 JMP l1
l17 LOAD R1,y
l18 CMP R1,0
l19 CJ>= l1
l20 LOAD R1,y
l21 LOAD R2,x
l23 ADD R1,R2
l24 STORE y,R2
l25 JMP l1
l26
```

(关闭)

7. 写出下面语句列的目标代码(优化后的):

```
a[i][j] := a[i][j] + 1;
i := j;
a[i][j] := a[i][j] + a[i][j];
```

(答案)

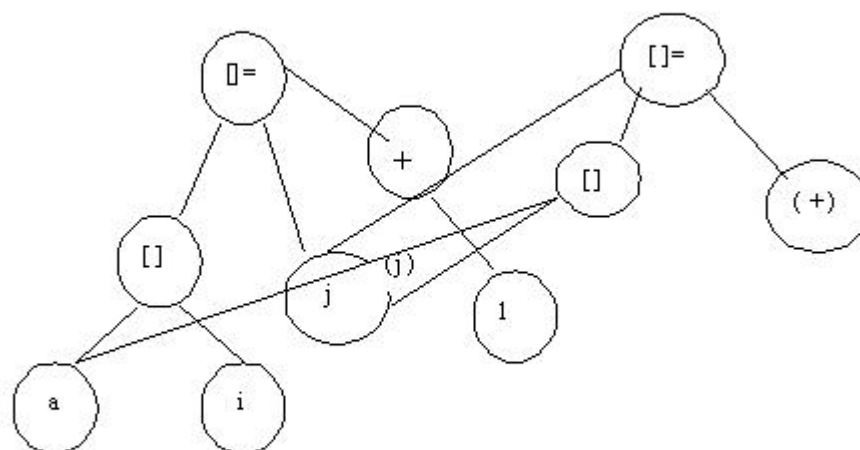
中间代码(优化后)		R ₁	R ₂	R ₃	R ₄	R ₅
(-, i, 1, T ₁)	(-, i, 1, T ₁)					
(*, T ₁ , 10, T ₂)	● LOAD R ₁ , i	i(L, NS)				
([], a, T ₂ , T ₃)	● LOAD R ₃ , #1	i(L, NS)	#1(L, NS)			
(-, j, 1, T ₄)	● MIN R ₁ , R ₂	T ₁ (L, S)	#1(L, NS)			
(*, j, 1, T ₅)	(*, T ₁ , 10, T ₂)					
([], T ₃ , T ₅ , T ₆)	● LOAD R ₃ , #10	T ₁ (L, S)	#1(L, NS)	#10(L, NS)		
(+, T ₆ , 1, T ₇)	● MULT R ₁ , R ₃	T ₃ (L, S)	#1(L, NS)	#10(L, NS)		
(=, T ₇ , -, T ₆)	([], a, T ₂ , T ₃)					
(=, j, -, i)	● LOAD R ₄ , a	T ₃ (L, S)	#1(L, NS)	#10(L, NS)	a(L, NS)	
(*, T ₄ , 10, T ₈)	● ADD R ₁ , R ₄	T ₃ (L, S)	#1(L, NS)	#10(L, NS)	a(L, NS)	
([], a, T ₈ , T ₉)	(-, j, 1, T ₄)					
([], T ₉ , T ₅ , T ₁₀)	● LOAD R ₅ , j	T ₃ (L, S)	#1(L, NS)	#10(L, NS)	a(L, NS)	j(L, NS)
(+, T ₁₀ , T ₁₀ , T ₁₁)	● MIN R ₅ , R ₂	T ₃ (L, S)	#1(L, NS)	#10(L, NS)	a(L, NS)	T ₄ (L, S)
(=, T ₁₁ , -, T ₁₀)						

	(*, j, 1, T ₅)	R ₁	R ₂	R ₃	R ₄	R ₅	R ₆
	● LOAD R ₆ , j	T ₃ (L, S)	#1(L, NS)	#10(L, NS)	a(L, NS)	T ₄ (L, S)	T ₅ (L, S)
	([], T ₃ , T ₅ , T ₆)						
	● ADD R ₁ , R ₆	T ₄ (L, S)	#1(L, NS)	#10(L, NS)	a(L, NS)	T ₄ (L, S)	T ₅ (L, S)
	(+, T ₆ , 1, T ₇)						
	● ADD R ₂ , R ₁	T ₄ (L, S)	T ₅ (L, S)	#10(L, NS)	a(L, NS)	T ₄ (L, S)	T ₅ (L, S)
	(=, T ₇ , -, T ₆)						
	● STORE R ₁ , R ₂			#10(L, NS)	a(L, NS)	T ₄ (L, S)	T ₅ (L, S)
	(=, j, -, i)						
	● STORE i, j			#10(L, NS)	a(L, NS)	T ₅ (L, S)	T ₅ (L, S)
	(*, T ₄ , 10, T ₈)						
	● MULT R ₅ , R ₃			#10(D, NS)	a(L, NS)	T ₅ (L, S)	T ₅ (L, S)
	([], a, T ₈ , T ₉)						
	● ADD R ₅ , R ₄				a(D, NS)	T ₅ (L, S)	T ₅ (L, S)
	([], T ₉ , T ₅ , T ₁₀)						
	● ADD R ₅ , R ₄					T ₁₀ (L, S)	T ₅ (L, S)
	(+, T ₁₀ , T ₁₀ , T ₁₁)						
	● LOAD R ₁ , R ₇	T ₁₀ (L, S)				T ₁₀ (L, S)	
	● ADD R ₁ , R ₅	T ₁₁ (L, S)				T ₁₀ (L, S)	
	(=, T ₁₁ , -, T ₁₀)						
	● STORE R ₅ , R ₁						

(关闭)

8. 画出 7 题的 DAG 图。

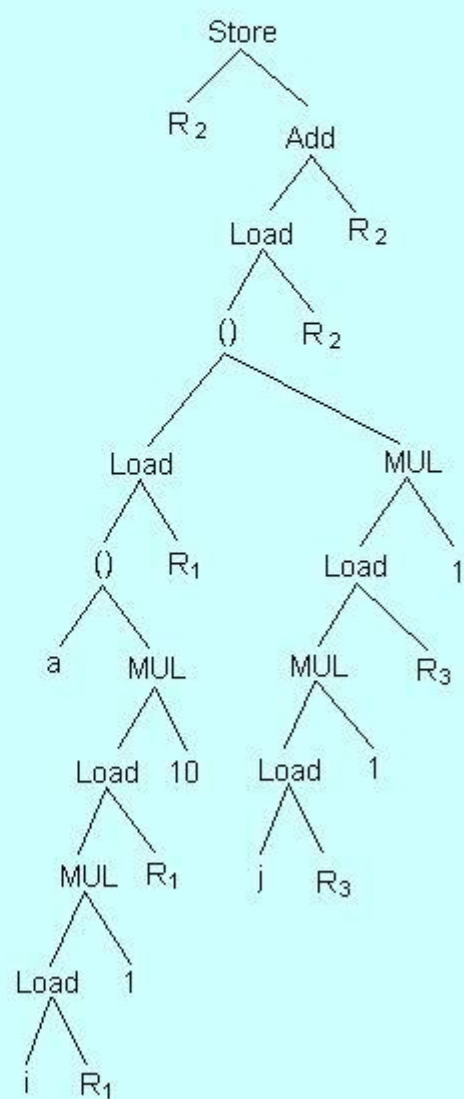
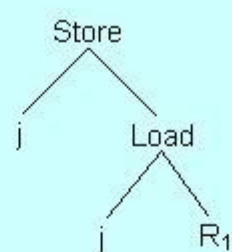
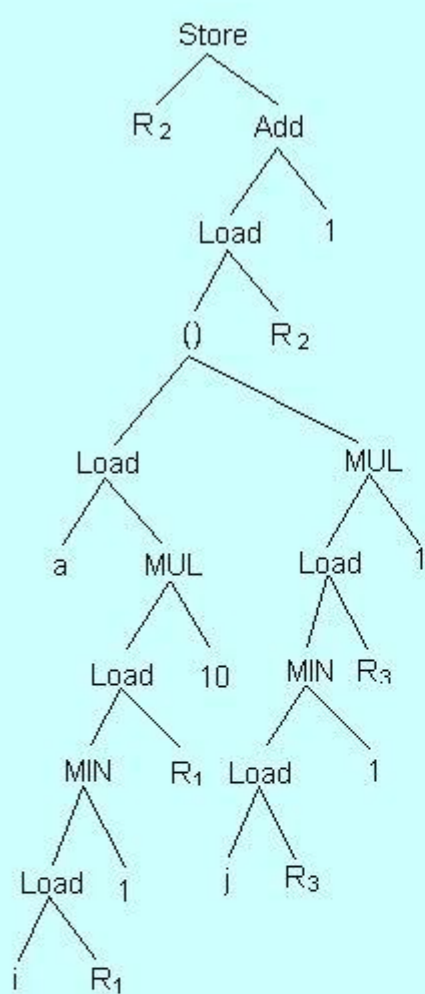
[\(答案\)](#)



[\(关闭\)](#)

9. 画出 7 题的目标树。

[\(答案\)](#)



(关闭)

[<<上一章](#) [◎](#) [◎](#)
[回](#)
[页](#)
[首](#)
[◎](#)