

题目 1

对以下中间代码序列 G:

```
T1 := B - C
T2 := A * T1
T3 := D + 1
T4 := E - F
T5 := T3 * T4
W := T2 / T5
```

假设可用寄存器为 R0 和 R1, W 是基本块出口的活跃变量, 用简单代码生成算法生成其目标代码, 同时列出代码生成过程中的寄存器描述和地址描述。

解答：步骤：变量活跃与待用信息标注 \Rightarrow 目标代码生成 \Rightarrow 寄存器与地址描述

1	T1 := B - C	(2, y) (^, ^) (^, ^)	LD R0, B SUB R0, C	R[R0] = {T1}	A[T1] = {R0}
2	T2 := A * T1	(6, y) (^, ^) (^, ^)	LD R1, A MUL R1, R0	R[R1] = {T2}	A[T2] = {R1}
3	T3 := D + 1	(5, y) (^, ^) (^, ^)	LD R0, D ADD R0, 1	R[R0] = {T3}	A[T3] = {R0}
4	T4 := E - F	(5, y) (^, ^) (^, ^)	ST R1, T2 LD R1, E SUB R1, F	R[R0] = {T3}	A[T2] = {R1}
5	T5 := T3 * T4	(6, y) (^, ^) (^, ^)	MUL R0, R1	R[R0] = {T5}	A[T5] = {R0}
6	W := T2 / T5	(^, y0) (^, ^) (^, ^)	LD R1, T2 DIV R1, R0 ST R1, W	R[R1] = {W}	A[W] = {R1}

题目 2

对以下中间代码序列:

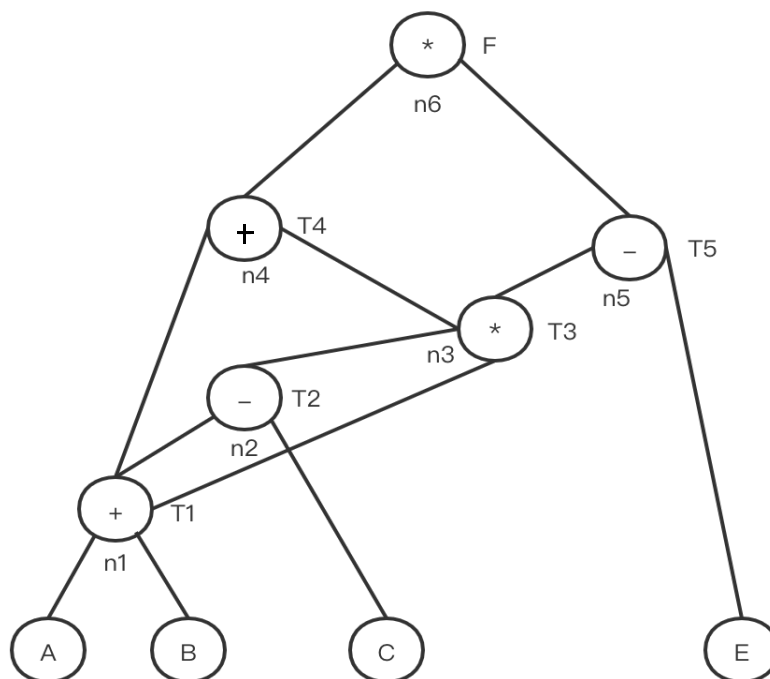
```
T1 := A + B
T2 := T1 - C
T3 := T2 * T1
T4 := T1 + T3
T5 := T3 - E
F := T4 * T5
```

(1) 应用 DAG 结点排序算法重新排序

(2) 假设可用寄存器为 R0, F 是基本块出口处的活跃变量, 应用简单代码生成算法分别生成排序前后的中间代码序列的目标代码, 并比较其优劣

解答:

(1) 根据上述代码序列, 可得到如下 DAG 图:



根据 DAG 图即可得到代码重排结果为: n1、n2、n3、n5、n4、n6

(2) 重排前:

1	T1 := A + B	(2, y) (^, ^) (^, ^)	LD R0, A ADD R0, B
2	T2 := T1 - C	(3, y) (3, y) (^, ^)	ST R0, T1 SUB R0, C
3	T3 := T2 * T1	(4, y) (^, ^) (4, y)	MUL R0, T1
4	T4 := T1 + T3	(6, y) (^, ^) (5, y)	ST R0, T3 LD R0, T1 ADD R0, T3
5	T5 := T3 - E	(6, y) (^, ^) (^, ^)	ST R0, T4 LD R0, T3 SUB R0, E
6	F := T4 * T5	(^, y) (^, ^) (^, ^)	ST R0, T5 LD R0, T4 MUL R0, T5 ST R0, F

重排后：

1	$T1 := A + B$	$(2, y) (\wedge, \wedge) (\wedge, \wedge)$	LD R0, A ADD R0, B
2	$T2 := T1 - C$	$(3, y) (3, y) (\wedge, \wedge)$	ST R0, T1 SUB R0, C
3	$T3 := T2 * T1$	$(4, y) (\wedge, \wedge) (5, y)$	MUL R0, T1
4	$T5 := T3 - E$	$(6, y) (5, y) (\wedge, \wedge)$	ST R0, T3 SUB R0, E
5	$T4 := T1 + T3$	$(6, y) (\wedge, \wedge) (\wedge, \wedge)$	ST R0, T5 LD R0, T1 ADD R0, T3
6	$F := T4 * T5$	$(\wedge, y) (\wedge, \wedge) (\wedge, \wedge)$	MUL R0, T5 ST R0, F

不难发现，DAG 重排后的目标代码长度明显减少。