



中国科学技术大学
University of Science and Technology of China



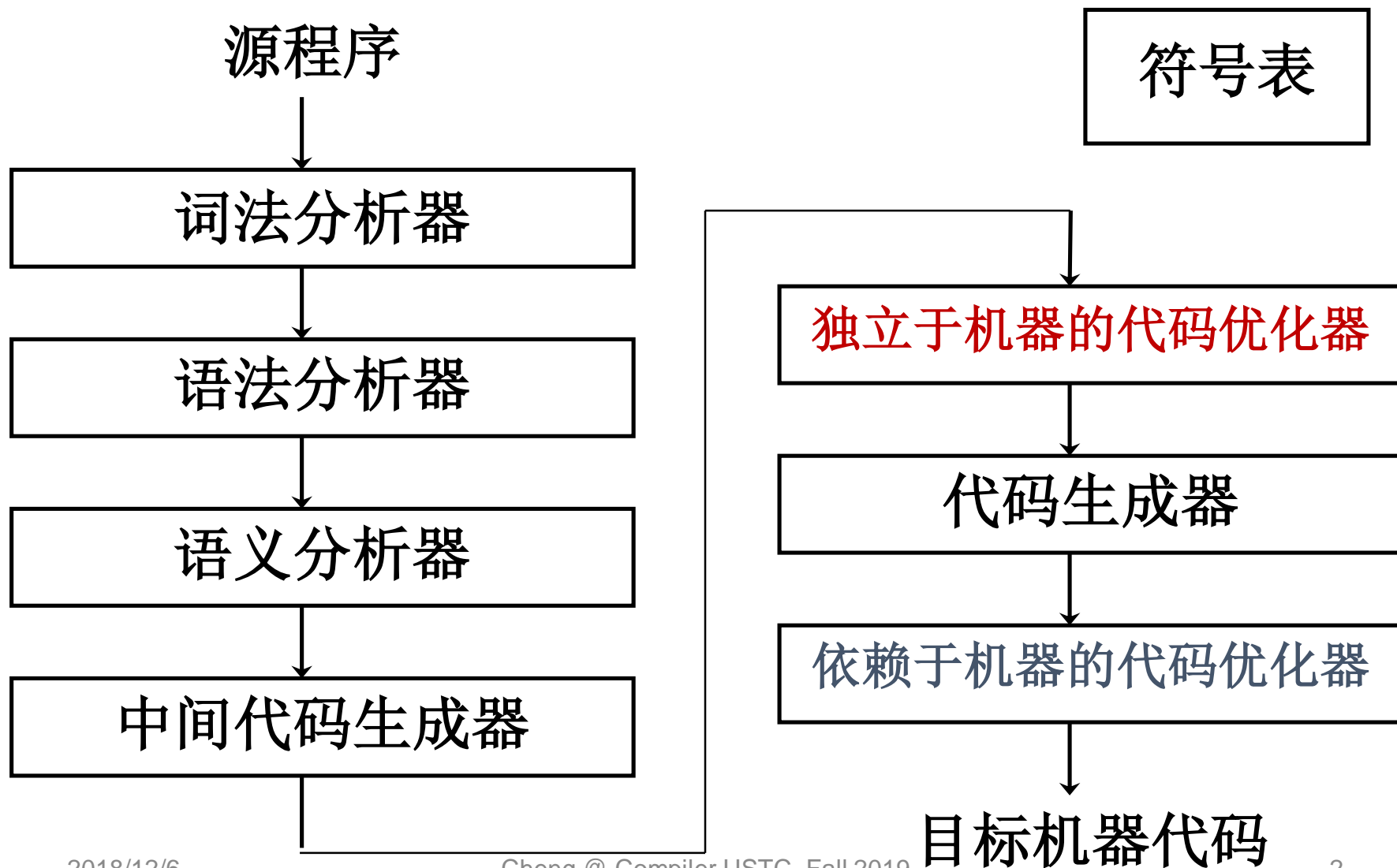
《编译原理与技术》

独立于机器的优化III

计算机科学与技术学院

李 诚

1/12/2019





□标识循环并对循环专门处理的重要性

- ❖ 程序执行的大部分时间消耗在循环上，改进循环性能的优化会对程序执行产生显著影响
- ❖ 循环也会影响程序分析的运行时间

□本节介绍

- ❖ 支配结点、深度优先排序、回边、自然循环
- ❖ 这些知识点可用来识别循环和分析迭代数据流分析的收敛速度(**不做要求和讲解**)

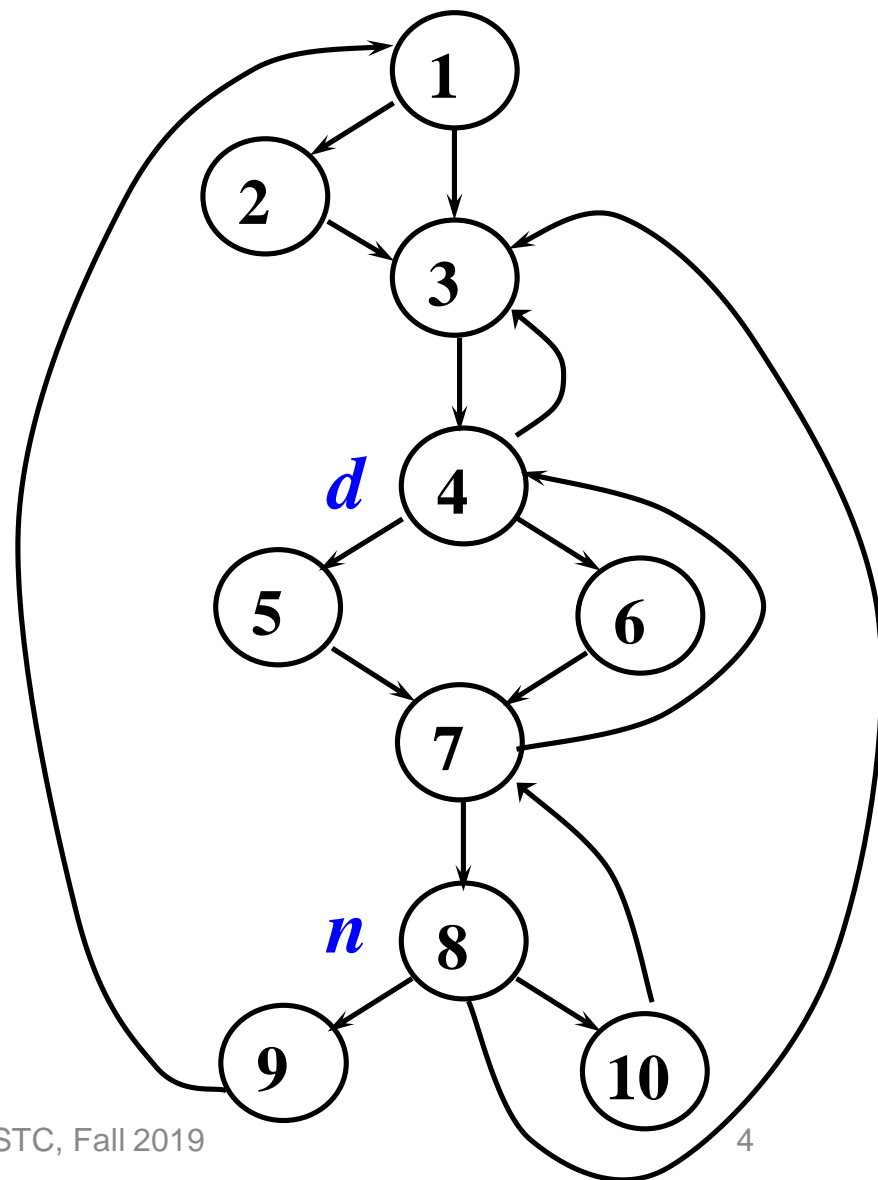


□ d 是 n 的支配结点:

❖ 若从初始结点起, 每条到达 n 的路径都要经过 d , 写成 $d \text{ dom } n$

□ 结点是他本身的支配结点

□ 循环的入口是循环中所有结点的支配结点

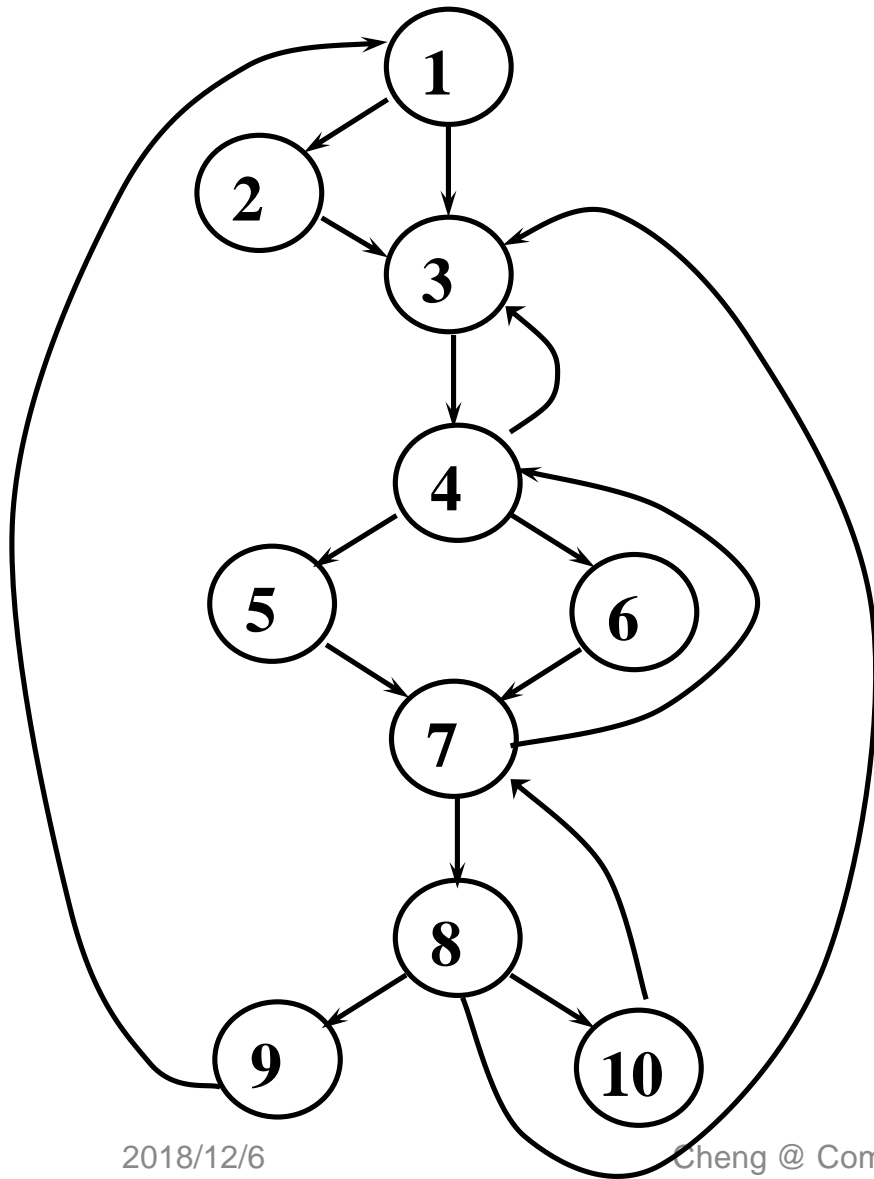




支配结点(Dominators)-举例



中国科学技术大学
University of Science and Technology of China



支配结点	支配对象
1	1-10
2	2
3	3-10
4	4-10
5	5
6	6
7	7-10
8	8-10
9	9
10	10

支配结点集的计算可以形式化为一个数据流问题，参考龙书算法9.38



□先序遍历

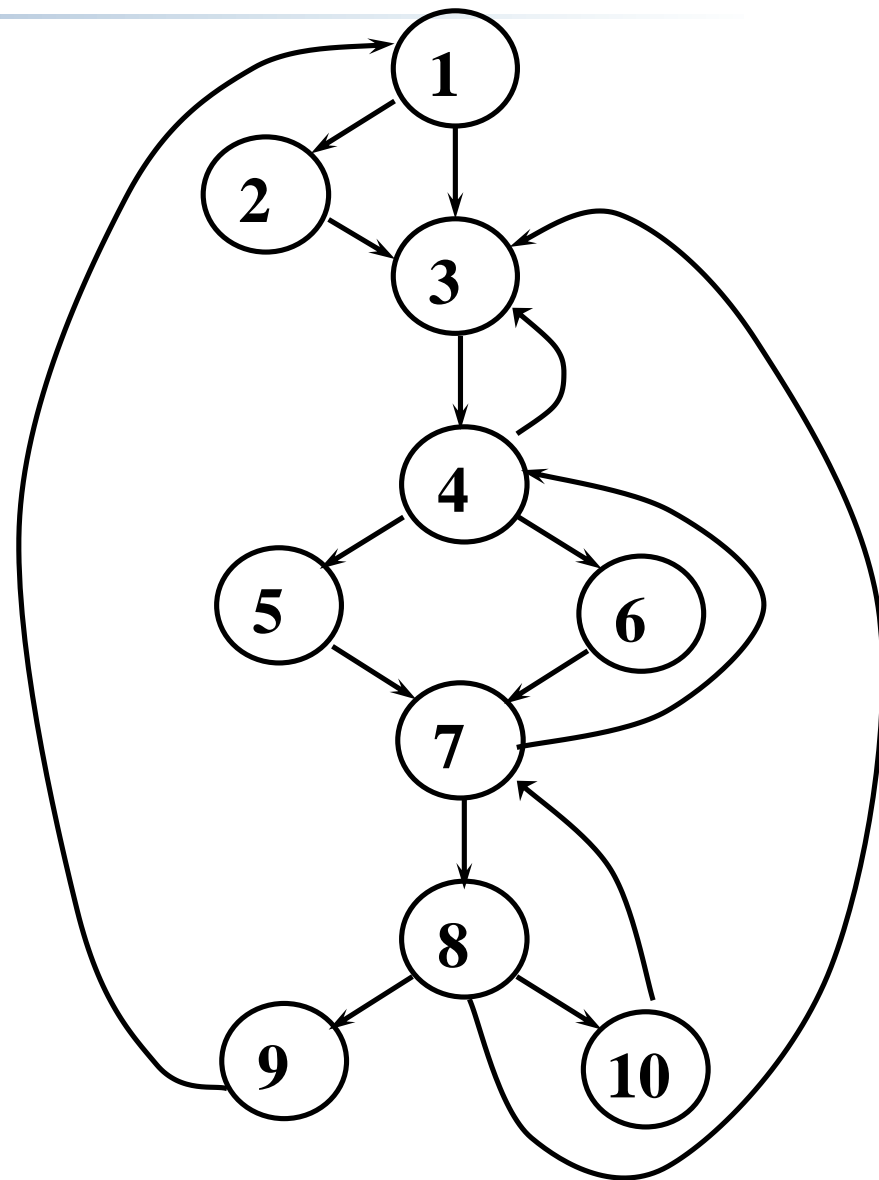
❖ 1,3,4,6,7,8,10,9,5,2

□后序遍历(左右根)

❖ 10,9,8,7,6,5,4,3,2,1

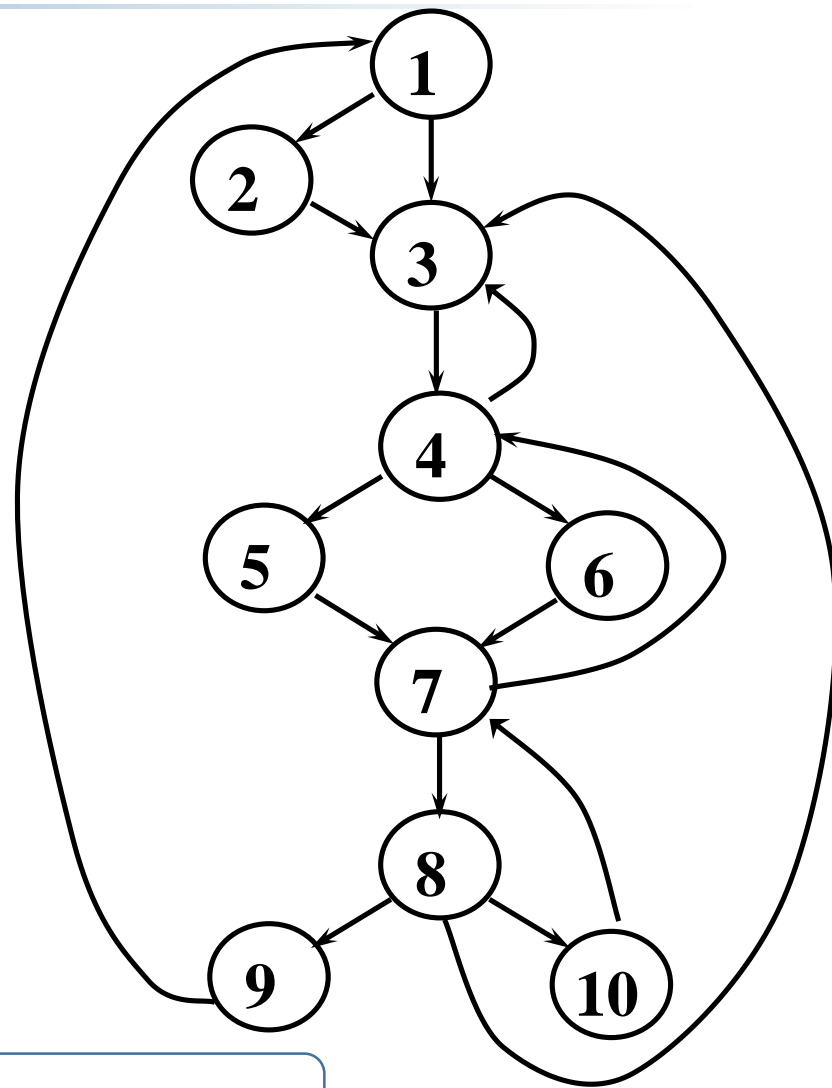
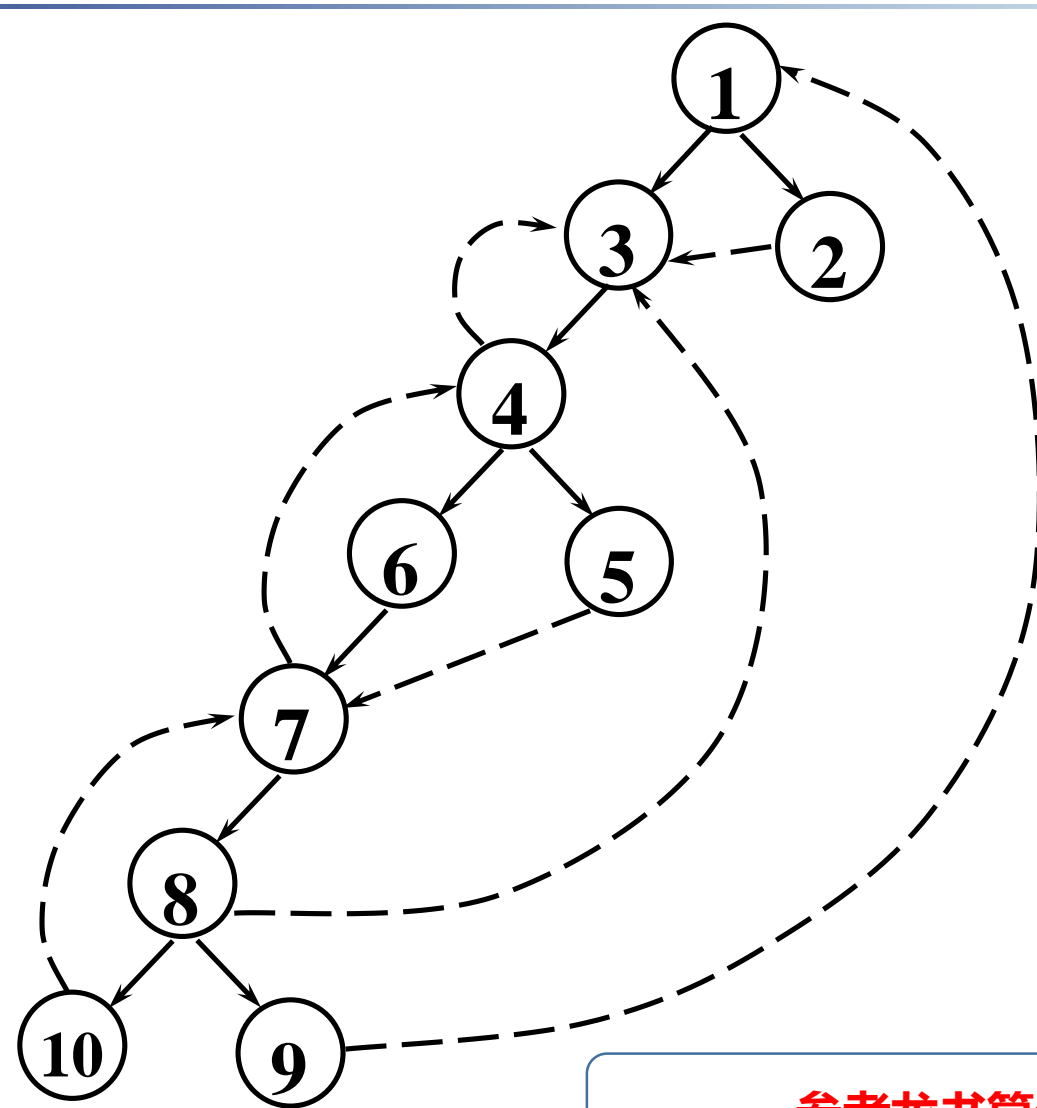
□深度优先排序正好与 后序遍历相反

❖ 1,2,3,4,5,6,7,8,9,10





深度优先表示



参考龙书算法9.41



□前进边

❖深度优先生成树的边

□后撤边

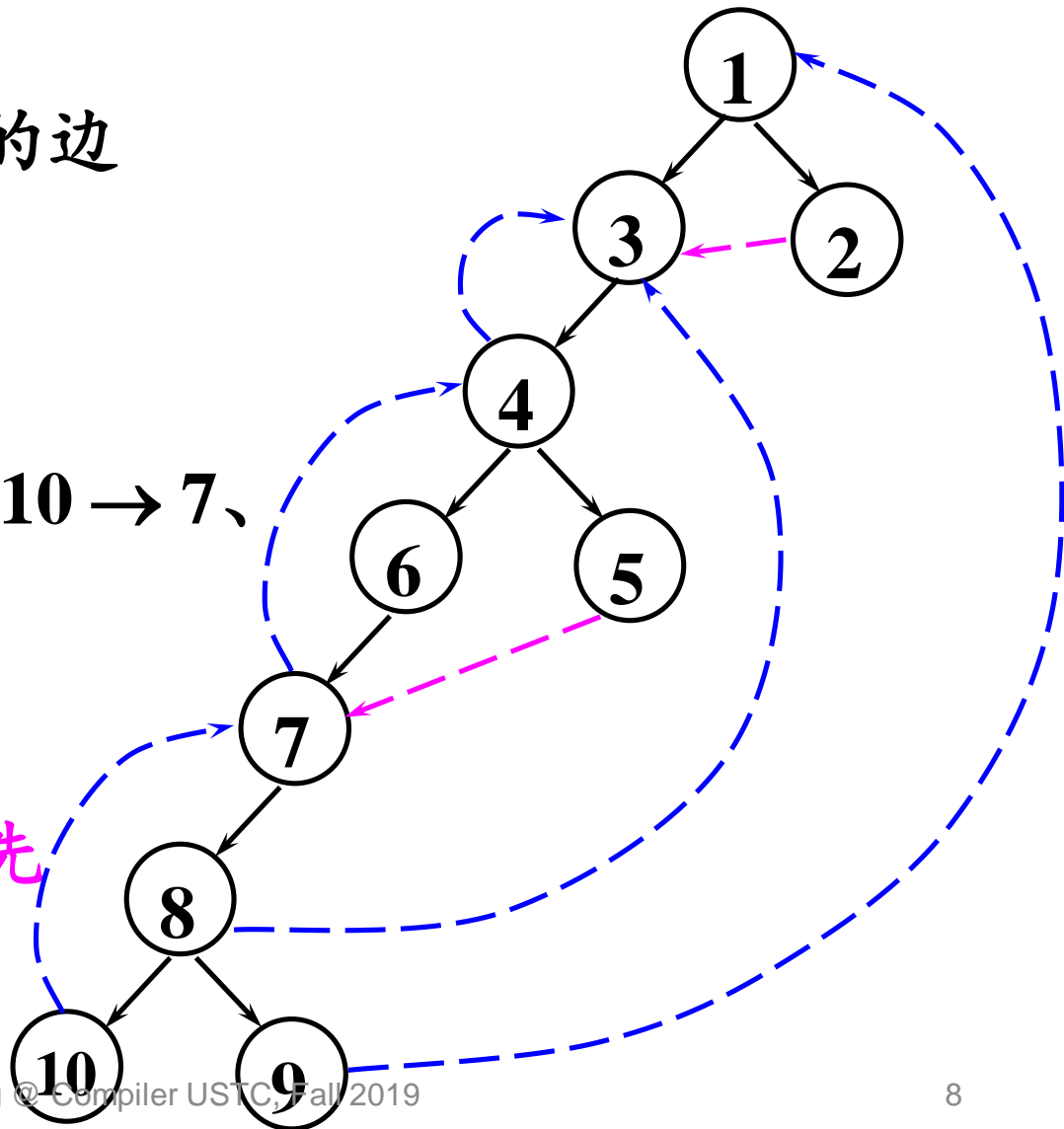
❖指向祖先节点

$4 \rightarrow 3$ 、 $7 \rightarrow 4$ 、 $10 \rightarrow 7$ 、
 $8 \rightarrow 3$ 和 $9 \rightarrow 1$

□交叉边

❖在树中互不为祖先

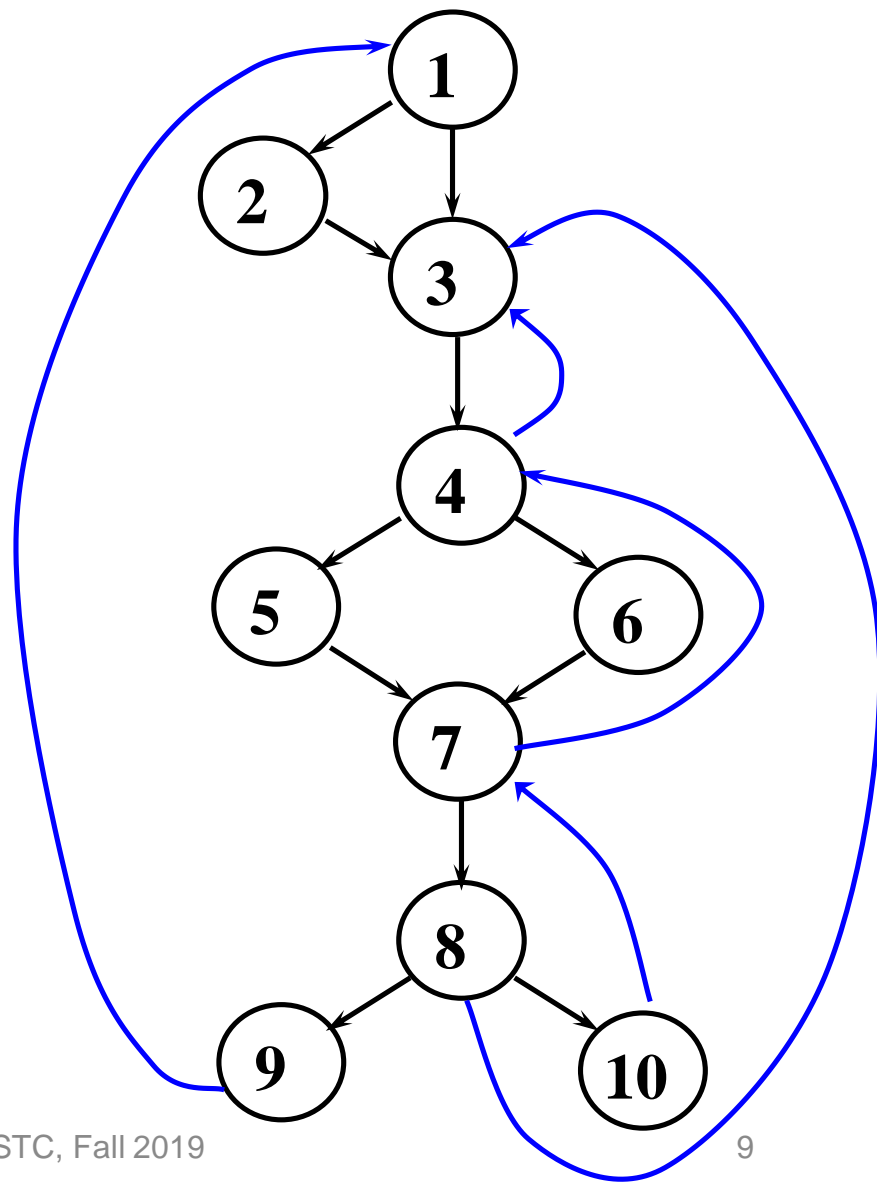
$2 \rightarrow 3$ 和 $5 \rightarrow 7$





□回边

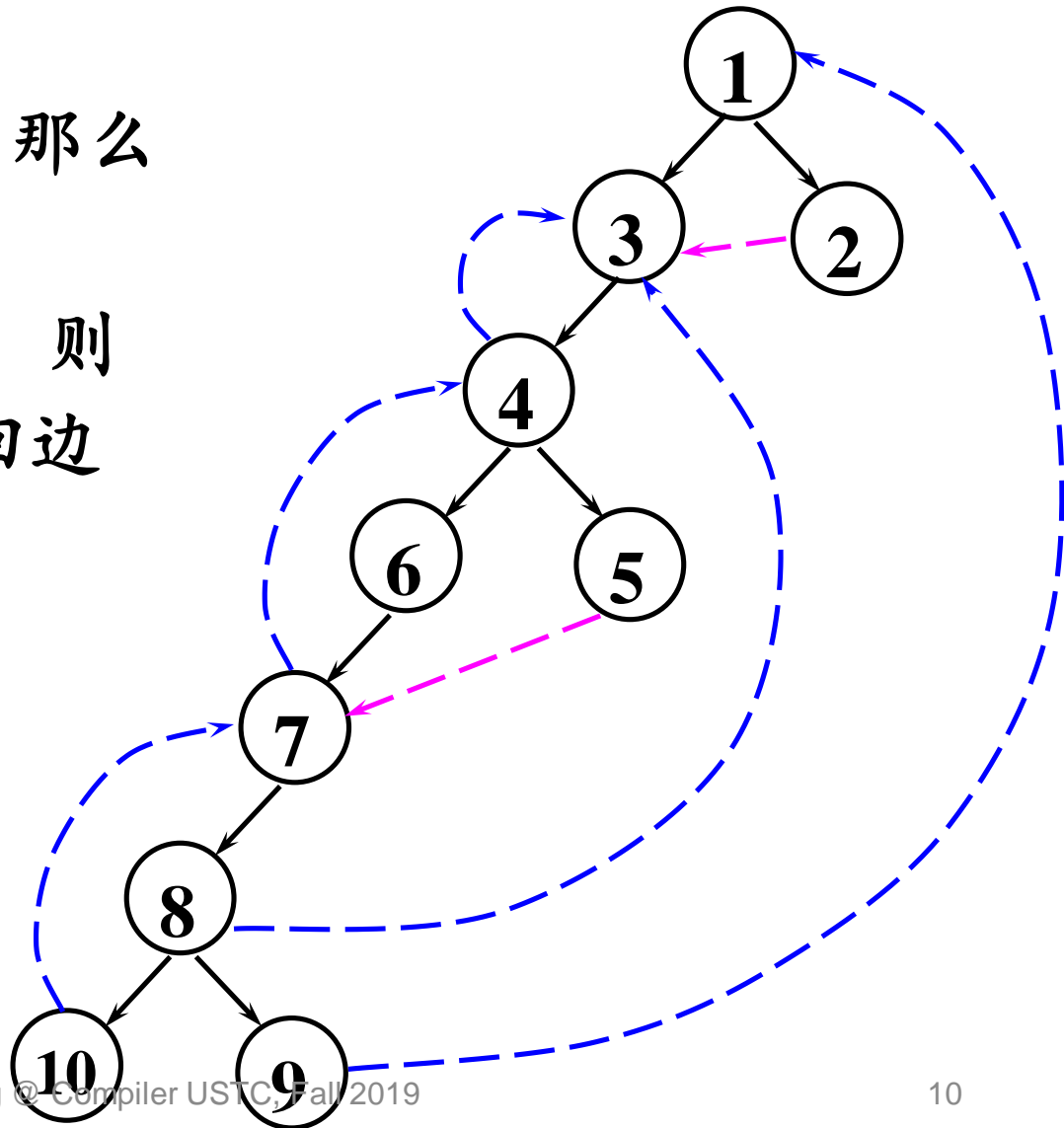
❖如果有 $a \text{ dom } b$, 那么
边 $b \rightarrow a$ 叫做回边





□回边

- ❖ 如果有 $a \text{ dom } b$, 那么边 $b \rightarrow a$ 叫做回边
- ❖ 如果流图可归约, 则后撤边正好就是回边





□可归约流图

❖ 如果把一个流图中所有回边删掉后，剩余的图无环

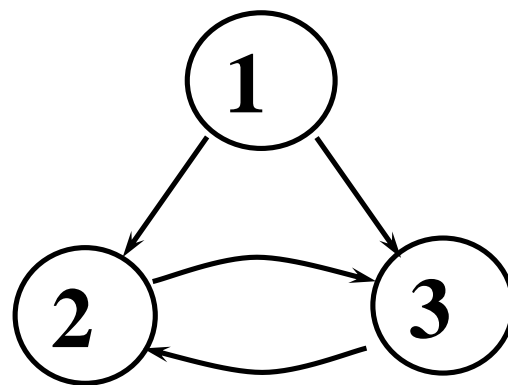
□例：不可归约流图

❖ 开始结点是1

❖ $2 \rightarrow 3$ 和 $3 \rightarrow 2$ 都不是回边

❖ 该图不是无环的

❖ 从结点2和3两处都能进入
由它们构成的环



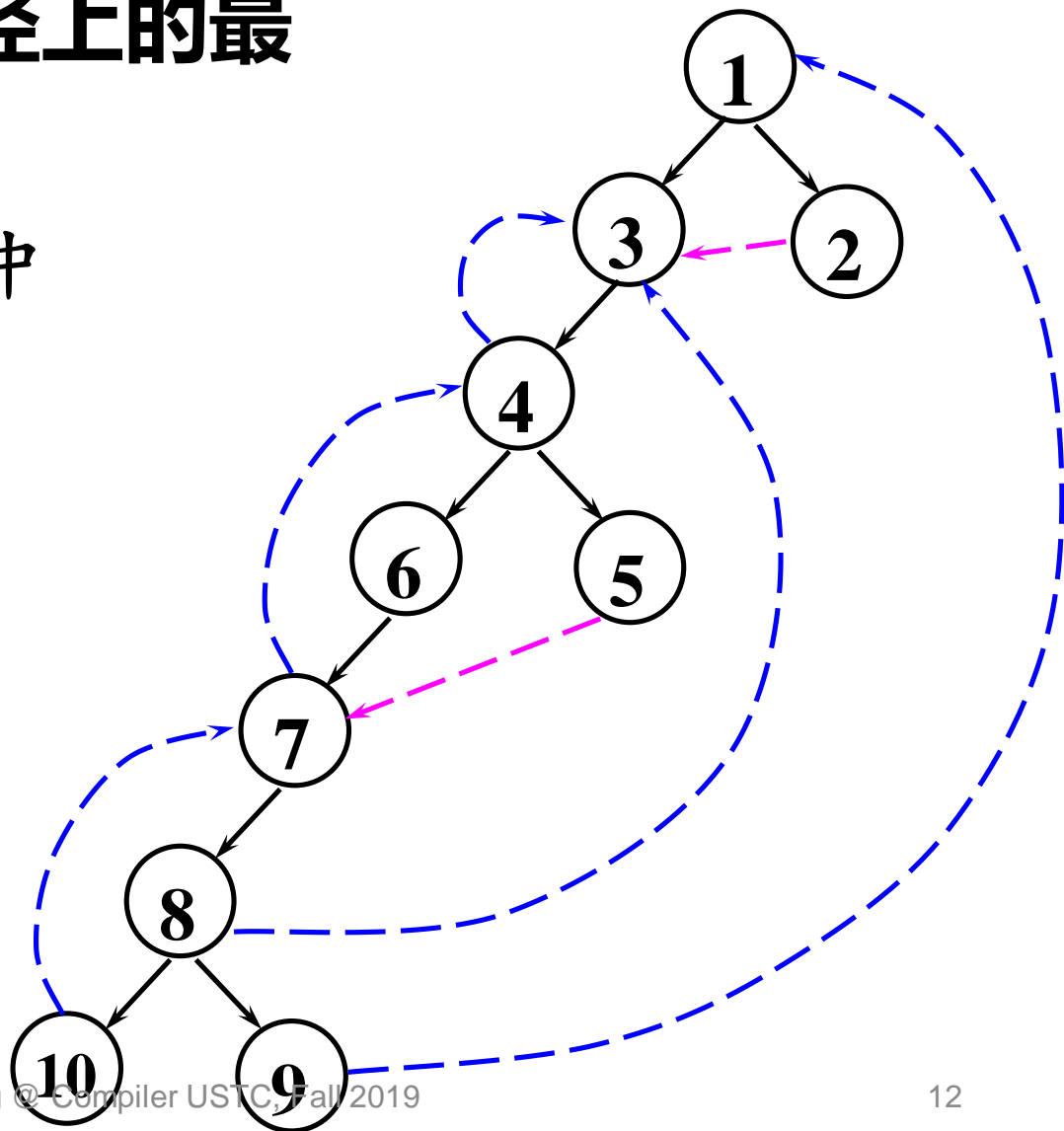


□深度是在无环路径上的最大后撤边数

❖深度不大于流图中
循环嵌套的层数

❖该例深度为3

$10 \rightarrow 7 \rightarrow 4 \rightarrow 3$





□在源程序中，循环可以用多种形式描述

❖for, while, goto等

□但从程序分析的角度来看，循环的代码形式并不重要，重要的是它是否具有一些易于优化的性质。

❖如果循环的入口结点唯一，那么我们就可以假设某些初始条件在循环的每一次迭代开头成立。



□ 自然循环的性质

- ❖ 有唯一的入口结点，叫做首结点，首结点支配该循环中所有结点
- ❖ 至少存在一条回边进入该循环首结点

□ 回边 $n \rightarrow d$ 确定的自然循环

- ❖ d 加上不经过 d 能到达 n 的所有结点
- ❖ 结点 d 是该循环的首结点



□回边 $10 \rightarrow 7$

循环 $\{7, 8, 10\}$

□回边 $7 \rightarrow 4$

循环 $\{4, 5, 6, 7, 8, 10\}$

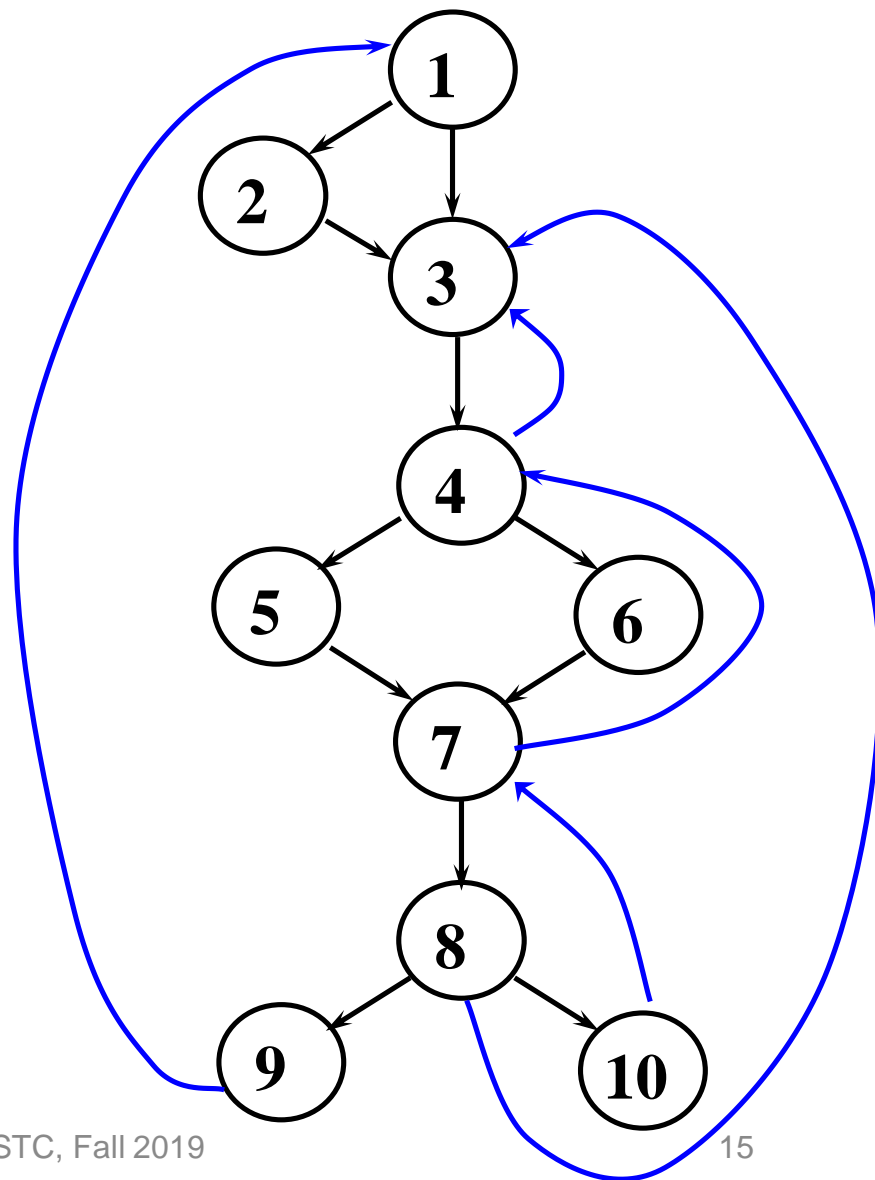
□回边 $4 \rightarrow 3$ 和 $8 \rightarrow 3$

循环 $\{3, 4, 5, 6, 7, 8, 10\}$

□回边 $9 \rightarrow 1$

$\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

参考龙书算法9.46

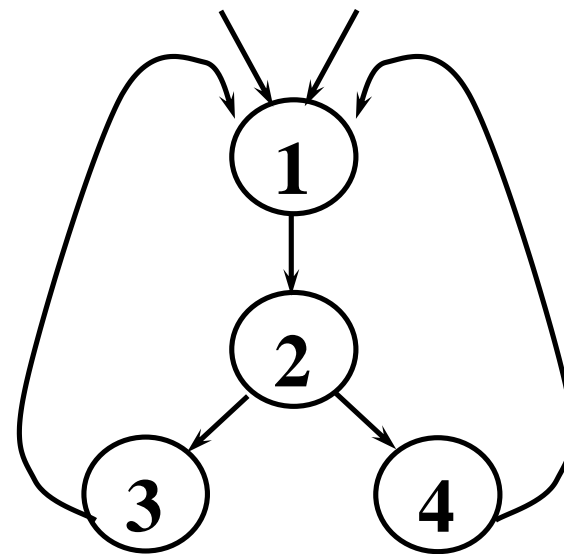




□内循环

❖若一个循环的结点集合是另一个循环的结点集合的子集

❖两个循环有相同的首结点，
但并非一个结点集是另一个
的子集，则看成一个循环





□ Notes on Graph Algorithms Used in Optimizing Compilers

❖ https://www.cs.umb.edu/~offner/files/flow_graph.pdf

□ 控制流分析

❖ <https://bbs.pediy.com/thread-247201.htm>



中国科学技术大学
University of Science and Technology of China



《编译原理与技术》

独立于机器的优化III

The end!