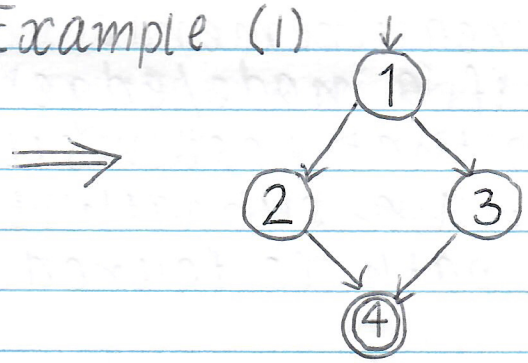
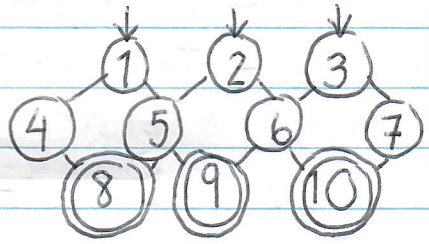
Lecture 3. Graphs (1)

- Outcomes
  - Create control flow graphs from source code
  - Identify test requirements for:
    - Node & Edge coverage
    - Prime path coverage
    - All-DU-Path, All-Uses, All-defs coverage
  - Idenify ~~fast~~ test paths for the above

- Graph Drawing Example (I)
  - $N = \{1, 2, 3, 4\}$
  - $N_0 = \{1\}$
  - $N_f = \{4\}$



- Graph Drawing Example (II)
  - $N = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
  - $N_0 = \{1, 2, 3\}$
  - $N_f = \{8, 9, 10\}$

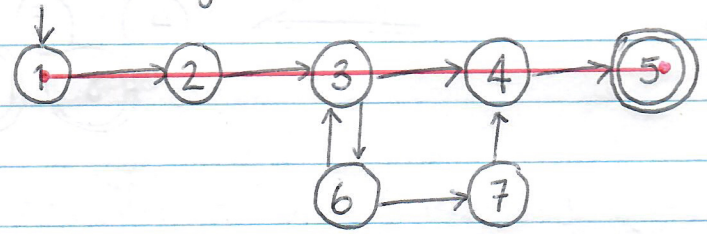


- Graph Terminology (1)
  - Definition
    - Graphs are effectively made up of sets:
      - A set of nodes
      - A set of initial nodes
      - A set of final nodes
      - A set of edges
    - These sets cannot be empty
  - Path: a sequence of nodes
  - Test path: a path that starts at an

intial node and ends at a final node
- Lenth: number of edges in a path
- Subpath: a path that makes up part of a larger path

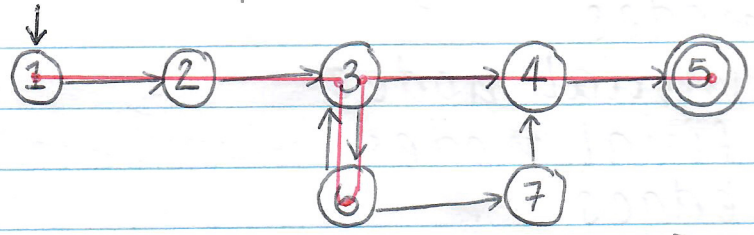
- Test Paths
  - Whenever you execute your code, it will run through a test path
  - A single test path may satisfy multiple test requirements
  - Visit: if a node/edge N exists in a test path, that test visits that node/edge
  - Tour: A subpath that is part of a test path is toured by that test path

- Touring, Side-trips and Detours
  - Test requirement: [1, 2, 3, 4, 5]
  - Touring



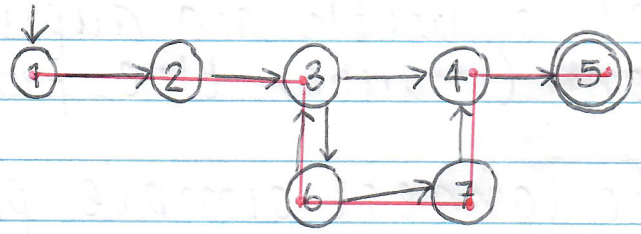
If our test path was also [1,2,3,4,5], it tours our requirement

  - side-trips



If our test path was [1,2,3,6,3,4,5], it tours our requirement with a side trip

- Detours



If our test path was [1, 2, 3, 6, 7, 4, 5], it tours our requirement with a detour

- Graph Terminology (II)
  - Lowercase (t) denotes an element of a set
  - Uppercase (T) denotes an entire set
  - Path(t): a test path executed by test t
  - Path(T): the set of test paths executed by test set T

- Test Requirements
  - Test criterion: the rules by which we design our tests (e.g. NC, EC, etc)
  - Test requirements: a set of paths that test paths should cover, based on a criterion
  - Satisfaction: make sure you have test paths that cover all the test requirements for a given criteria (e.g. visiting every node if node coverage is criterion)

- Types of Criterion for Graphs
  - Structural: criteria that investigate the structure of the graph and the paths
  - Data flow: criteria that investigate the variables and how data is actually used

- Simple Paths and Prime Paths
  - Simple paths: paths with no duplicate nodes apart from (maybe) the first and last nodes
  - Prime paths: the longest simple paths you can make out of the graph

- Realistically, test criteria will end up having infeasible test requirements

Lecture 3. Static Code Metrics (ISAD3002)

- Calculating Cyclomatic Complexity
  - Frequently calculated/represented using a control flow graph
  - There are two equations based on a CFG
    1) Count the number of decision points:
       $cc = d + 1$
    2) Count the number of edges (e) and the number of nodes (n):
       $cc = e - n + 2$