CS409
Software Testing

TAN, Shin Hwei
陈馨慧
Southern University of Science and Technology

Slides adapted from Introduction to Software Testing, Edition 2 (Ch 7)

Graph lab

Graph lab

- Graph-lab in GitHub Classroom:
- https://classroom.github.com/a/rkg8YIET
- Write all answers in README.md
  - Don’t forget your student id and names!

In Class Extended Exercise 1

- $N = \{1, 2, 3, 4\}$
- $N_0 = \{1\}$
- $N_4 = \{4\}$
- $E = \{(1,2), (2,3), (3,2), (2,4)\}$

Step 1: Draw this graph

work …

In Class Extended Exercise 1

- $N = \{1, 2, 3, 4\}$
- $N_0 = \{1\}$
- $N_4 = \{4\}$
- $E = \{(1,2), (2,3), (3,2), (2,4)\}$

Step 2: If possible, list test-paths that achieve Node Coverage, but not Edge Coverage, explain why not?

work …

In Class Extended Exercise 1

- $N = \{1, 2, 3, 4\}$
- $N_0 = \{1\}$
- $N_4 = \{4\}$
- $E = \{(1,2), (2,3), (3,2), (2,4)\}$

Step 3: If possible, list test-paths that achieve Edge Coverage, but now Edge-Pair Coverage, explain why not?

work …

In Class Extended Exercise 1

- $N = \{1, 2, 3, 4\}$
- $N_0 = \{1\}$
- $N_4 = \{4\}$
- $E = \{(1,2), (2,3), (3,2), (2,4)\}$

Step 4: List test paths that achieve Edge-Pair Coverage

work …

Use Tool

- $N = \{1, 2, 3, 4\}$
- $N_0 = \{1\}$
- $N_4 = \{4\}$
- $E = \{(1,2), (2,3), (3,2), (2,4)\}$

Step 4: Compare your results with
https://cs.gmu.edu:8443/offutt/coverage/GraphCoverage

work …

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Step 1: Draw this graph

work …

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Step 2: List the test requirements for Edge-Pair Coverage. *(Hint: You should get 12 requirements)*

work …

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Consider the following (candidate) test paths:

- p1 = [1, 2, 4, 5, 6, 1, 7]
- p2 = [1, 2, 3, 2, 4, 6, 1, 7]
- p3 = [1, 2, 3, 2, 4, 5, 6, 1, 7]

Step 3: Does these test paths satisfy Edge-Pair Coverage? If not, state what is missing?

work ...

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Step 4: Consider the simple path [3, 2, 4, 5, 6] & test path [1, 2, 3, 2, 4, 6, 1, 2, 4, 5, 6, 1, 7]. Does the test path tour the simple path directly? With a sidetrip? If so, write down the sidetrip.

work ...

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Step 5: List the test requirements for Node Coverage, Edge Coverage, and Prime Path Coverage.

work …

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Consider the following (candidate) test paths:

- p1 = [1, 2, 4, 5, 6, 1, 7]
- p2 = [1, 2, 3, 2, 4, 6, 1, 7]
- p3 = [1, 2, 3, 2, 4, 5, 6, 1, 7]

Step 5: List test paths from the given set that achieve Node Coverage but not Edge Coverage.

work ...

In Class Extended Exercise 2

- $N = \{1, 2, 3, 4, 5, 6, 7\}$
- $N_0 = \{1\}$
- $N_f = \{7\}$
- $E = \{(1, 2),(1, 7),(2, 3),(2, 4),(3, 2),(4, 5),(4, 6),(5, 6),(6, 1)\}$

Consider the following (candidate) test paths:

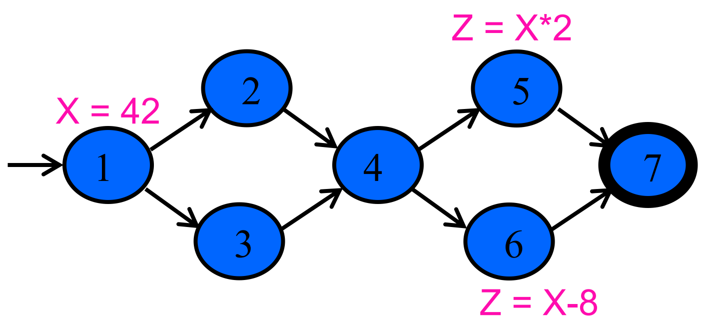
- p1 = [1, 2, 4, 5, 6, 1, 7]
- p2 = [1, 2, 3, 2, 4, 6, 1, 7]
- p3 = [1, 2, 3, 2, 4, 5, 6, 1, 7]

Step 6: List test paths from the given set that achieve Edge Coverage but not Prime Path Coverage on the graph.

work …

Data Flow Analysis Exercise

Data Flow Testing Example in Lecture



All-defs for *X*
[1, 2, 4, 5]

All-uses for *X*
[1, 2, 4, 5]
[1, 2, 4, 6]

All-du-paths for *X*
[1, 2, 4, 5]
[1, 3, 4, 5]
[1, 2, 4, 6]
[1, 3, 4, 6]

Lab: Data flow example

```
int bias;
if(inhibit) {
      bias = down_sep;
}
else {
      bias = up_sep;
}

if(bias > down_sep) {
      return 1;
}
else {
       return 0;
}
```

1. Write the def and use of all variables
2. Are they C-use or P-use?

work ...

Lab: Data flow example

```
int bias;
if(inhibit) {
      bias = down_sep;
}
else {
      bias = up_sep;
}

if(bias > down_sep) {
      return 1;
}
else {
      return 0;
}
```

A definition of bias

A C-use of down_sep

Definitions of bias

A C-use of up_sep

P-uses of bias

Lab: Data flow example

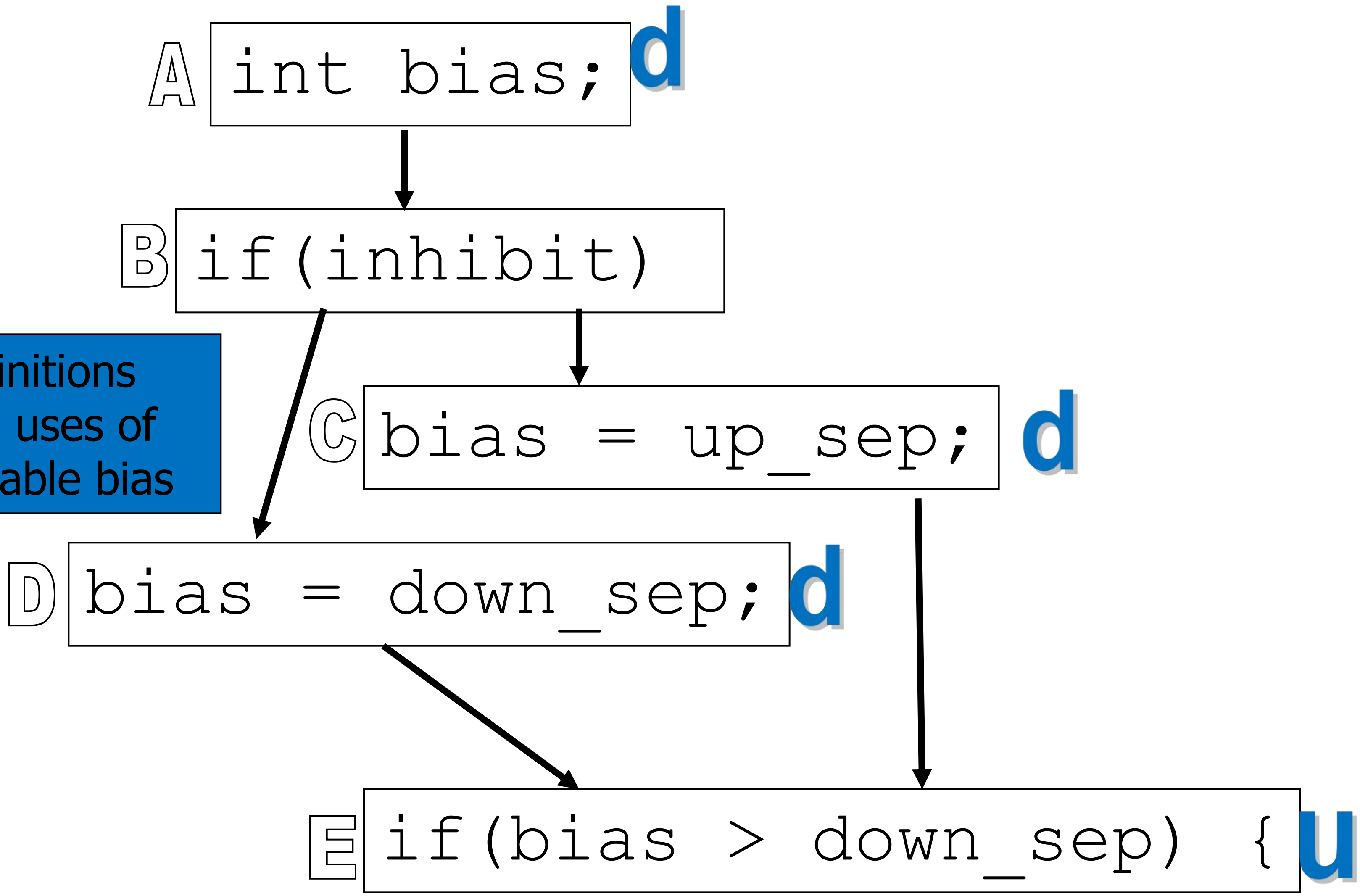
```
int bias;
if(inhibit) {
      bias = down_sep;
}
else {
      bias = up_sep;
}

if(bias > down_sep) {
      return 1;
}
else {
       return 0;
}
```

- Mark the statement that are definition of bias with **d**
- Mark the statement that are usage of bias with **u**

work ...

Definitions and uses of variable bias



What to submit?

Part 1:

- Pictures (on paper or using power point/paint or any painting tools) showing the graph that you draw
  - In Class Extended Exercise 1
  - In Class Extended Exercise 2

Part 2:

1. Draw the data flow graph for the variable “bias” for the “*Lab: Data flow example*”
2. Write down All-defs for “bias”
3. Write down All-uses for “bias”
4. Write down All-du-paths for “bias”

- Write all answers in **README.md**
  - Don’t forget your student id and names!