Tutorial 6: Program Verification

CS60030 Formal Systems

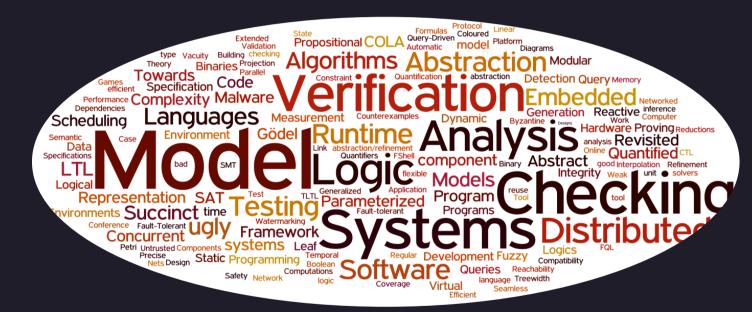
PALLAB DASGUPTA,

FNAE, FASc,

A K Singh Distinguished Professor in Al, **Dept of Computer Science & Engineering Indian Institute of Technology Kharagpur**

Email: pallab@cse.iitkgp.ac.in

Web: http://cse.iitkgp.ac.in/~pallab









Weakest Pre-condition

1) Starting with the post condition and statements, fill in the intermediate assertions and weakest precondition:

1) if(
$$x > y$$
)

2)
$$w = 2 * w$$
;

3) if(
$$x > y$$
)

4)
$$x = 2 * x$$
;

$$x = x-y$$
;

$$z = v - 2$$
;

$$z = x;$$

$$y = x + 2;$$

else

$$y = -w;$$

else

$$z = y/2;$$

$$y = y-x;$$

$$x = min(y,z);$$

$$z = y;$$

$$\{z > x\}$$

$$\{\gcd(x, y) = z\}$$

$${x < 0}$$

$${z = max(x, y)}$$

Weakest Pre-condition

- 5) Compute wp(A, x < y) in as simple form as you can, where A is the following code fragment
- 6) Compute weakest precondition for the following loop code

```
if (x > y)
    tmp = x;
    x = y;
    y = tmp;
```

Program Abstraction

Check if the following code computes the maximum of two no.s for any given set of input

```
{ true }
if (x > y) : r = x;
else : r = y;
```

Analyze the program using the domain Parity and then Sign.

```
y = 5;
x = -2 * y;
if (x > 0){
x = x - (y%2 - 1);
y = x * (y - 1);
else
y = -1;
```

Program Verification

```
L1:
       a = b = i = 0;
      while (a <= 1000) {
L2:
L3:
       a = b + i;
L4:
              b = a + 1;
L5:
              i = i + 1;
L6:
L7:
       if (b > 2000) {
L8:
      error: exit(-1);
L9:
```

- 1. Construct a Boolean program corresponding to this program P, using only the predicates (a ≤ 1000), (b ≥ 0), (i = 0), and (b ≤ 2000)
- 2. Show that the error location is reachable in the Boolean program you constructed
- 3. Explain whether the error identified in this Boolean program is a spurious counter-example.
- 4. If the counterexample is spurious can you suggest refined predicates such that the counterexample disappears?

Program Verification

Use abstract interpretation on the following program to determine whether the assertion is true. Use the sign domain as your abstract domain, which has only three elements, [+, -, 0], representing +ve, -ve, and zero respectively. Show the values collected in each location. Also indicate the outcome:

```
L1: x = 1;
     if (y < = 10){
L2:
L3:
L3:
       else{
L5:
               while (x < y){
L6:
                      x = 2 * x;
L7:
                       y = y - 1;
L8:
               assert (x > 0);
L9:
```