CS 4120
Introduction to Compilers

Ross Tate
Cornell University

Lecture 35: Interprocedural Analysis

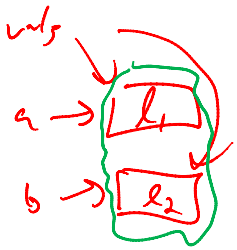
Aliasing

```
sort(int[] vals) {…}
a = new int[100] {…};
b = new int[100] {…};
sort(a);
sort(b);
for (0 <= i < 100) {
    a[i] = b[i] + a[i];
    b[i] = b[i] + 1;
}
```

a and b do not alias

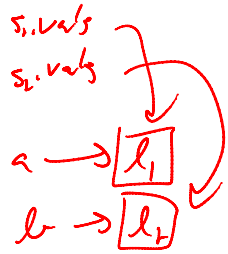
Imprecision

```
sort(int[] vals) {…}
a = new int[100] {…};
b = new int[100] {…};
sort(a);
sort(b);
```



Using Context

```
sort(int[] vals) {…}
a = new int[100] {…};
b = new int[100] {…};
sort(a);
sort(b);
```



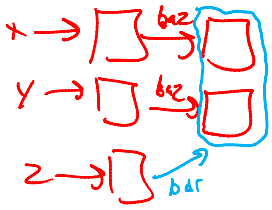
kCFA

```
int f() {
    g(1);
    return h(2); }
int g(int x) {
    if (x < 3)
        return g(x+1);
    return h(x); }
int h(int x) {
    return x; }
```

$s_1.x = 1$
$s_1.s_3.x = 2$
$\ldots s_3.s_3.x = ?$
$\ldots s_3.s_4.x = ?$
$s_0.x = 0$
$s_0.s_3.x = 1$

Summaries

```
foo(x, y, z) {
    if (…)
        z.bar = x.baz;
    else
        z.bar = y.baz;
}
```



Object-Sensitive Analysis

- distinguish call context based on "this"
  - context includes abstract location for "this"

```
class Y { X mX; setX(X x) {mX = x;} }
set(Y y, X x) { y.setX(x); }
x1 = new X(); x2 = new X();
y1 = new Y(); y2 = new Y();
set(y1, x1); set(y2, x2);
y1.x.i = y1.x.i + y2.x.i;
y2.x.i = y2.x.i + 1;
```