

Week 3 Static Analysis Examples

Constant Propagation (demo)

- Flat lattice of integer values
- Rules are same as Sign analysis, but with wider lattice.
- Note: this is a **Forward analysis**

```
var x,y,z;           [x->T, y->T, z->T]
x = 27;              [
y = input;           [
z = 2*x+y;            [
if (x < 0)            ?
    { y = z-3; }      [
else
    { y = 12; }       [
;
output y;
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Live variables analysis

- A variable is *live* at a program point if its current value may be read in the remaining execution.
- (see Slide 7 in 4-flow-sensitive-analysis.pdf)
- **Note: Backward May analysis! JOIN = union.** $L = (2^{\{x,y,z\}}, \subseteq)$

	Iteration 1	Iteration 2
<code>var x,y,z;</code>	{	
<code>x = input;</code>	{	
<code>while (x>1) {</code>	{	
<code>y = x/2;</code>	{	
<code>if (y>3)</code>	{	
<code>x = x-y;</code>	{	
<code>z = x-4;</code>	{	
<code>if (z>0)</code>	{	
<code>x = x/2;</code>	{	
<code>z = z-1;</code>	{	
<code>}</code>		
<code>output x;</code>	{	
<code>exit</code>	{	

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Available expressions analysis

- a nontrivial expression is *available* at a program point if its current value must have already been computed earlier in the execution
- (see Slide 18 in 4-flow-sensitive-analysis.pdf)
- **Note: FORWARDS** analysis! JOIN=intersection (a **MUST** analysis)

$$L = (2^{\{a+b, a*b, y>a+b, a+1\}}, \supseteq)$$

```
var x,y,z,a,b;
```

```
z = a+b;
```

```
y = a+b;
```

```
while (y > a) {
```

```
    a = a+1;
```

```
    x = a*b;
```

```
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Very busy expressions analysis

- a nontrivial expression is *very busy* if it will definitely be evaluated before its value changes
- (see Slide 28 in 4-flow-sensitive-analysis.pdf)
- **Note:** Backwards analysis! JOIN=intersection (a **MUST** analysis)
- (same style of lattice as previous analysis, but with expressions from this program)

$$L = (2^{\{a+b, a*b, y>a+b, a+1\}}, \supseteq)$$

```
var x,a,b;
{
x = input;
{
a = x-1;
{
b = x-2;
{
while (x > 0) {
{
output a*b-x;
{
x = x-1;
{
}
{
output a*b;
{
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Reaching definitions analysis (def-use)

- The *reaching definitions* for a program point are those assignments that may define the current values of variables.
- (see Slide 35 in 4-flow-sensitive-analysis.pdf)
- **Note: FORWARDS** analysis! JOIN=union (a **MAY** analysis)

The powerset lattice of assignments

$$L = (2^{\{x=input, y=x/2, x=x-y, z=x-4, x=x/2, z=z-1\}}, \subseteq)$$

```
var x,y,z;      {  
  
x = input;      {  
  
while (x > 1) {  
    y = x/2;     {  
    if (y>3)  
        x = x-y; {  
    z = x-4;      {  
    if (z>0)  
        x = x/2;  {  
    z = z-1;      {  
    }             {  
    output x;
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder