

Liveness Analysis

Akim Demaille Étienne Renault Roland Levillain
first.last@lrde.epita.fr

EPITA — École Pour l'Informatique et les Techniques Avancées

January 22, 2016

Liveness Analysis

- 1 Control Flow Graph
- 2 Liveness
- 3 Interference Graph

Control Flow Graph

1 Control Flow Graph

2 Liveness

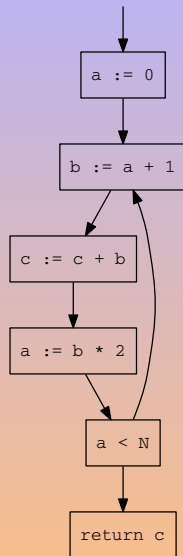
3 Interference Graph

Control Flow Graph [Appel, 1998]

```
    a := 0
L1:  b := a + 1
    c := c + b
    a := b * 2
    if a < N goto L1
    return c
```

Control Flow Graph [Appel, 1998]

```
a := 0
L1: b := a + 1
   c := c + b
   a := b * 2
   if a < N goto L1
   return c
```



$1 + 2 * 3$

7's Pre-Assembly

```
tc_main:
```

```
# Allocate frame
```

```
    move    $x13, $ra
    move    $x5, $s0
    move    $x6, $s1
    move    $x7, $s2
    move    $x8, $s3
    move    $x9, $s4
    move    $x10, $s5
    move    $x11, $s6
    move    $x12, $s7
```

```
l0:
```

```
    li      $x1, 1
    li      $x2, 2
    mul     $x3, $x2, 3
    add     $x4, $x1, $x3
```

```
l1:
```

```
    move    $s0, $x5
    move    $s1, $x6
    move    $s2, $x7
    move    $s3, $x8
    move    $s4, $x9
    move    $s5, $x10
    move    $s6, $x11
    move    $s7, $x12
    move    $ra, $x13
```

```
# Deallocate frame
```

```
    jr      $ra
```

7's Flowgraph



1 | 2 & 3

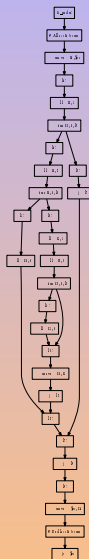
7000's Pre-Assembly

```
tc_main:
# Allocate frame

    move    $x6, $ra
18:
    li      $x3, 1
    bne     $x3, 0, 15
16:
    li      $x4, 2
    bne     $x4, 0, 10
11:
    li      $x0, 0
12:
17:
    j       19
```

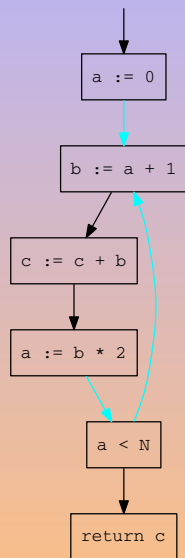
```
10:
    li      $x1, 1
    li      $x5, 3
    bne     $x5, 0, 13
14:
    li      $x1, 0
13:
    move    $x0, $x1
    j       12
15:
    j       17
19:
    move    $ra, $x6
# Deallocate frame
    jr      $ra
```

7000's Flowgraph

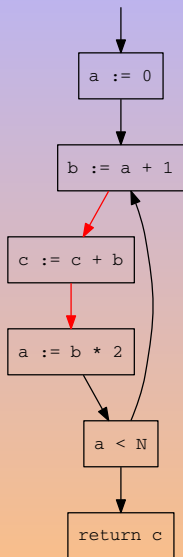
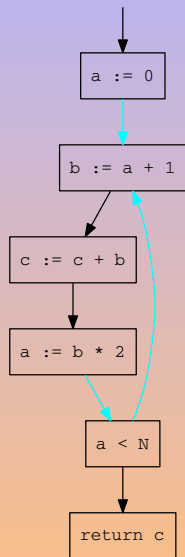


- 1 Control Flow Graph
- 2 Liveness**
- 3 Interference Graph

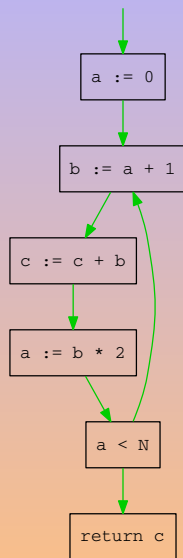
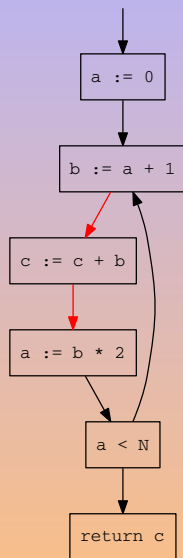
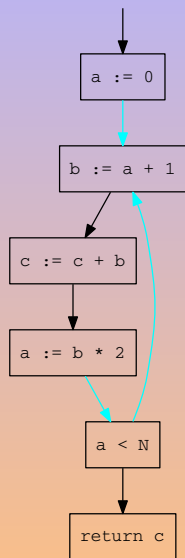
Liveness



Liveness



Liveness



Dataflow Equations for Liveness Analysis

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a								
2	a	b								
3	bc	c								
4	b	a								
5	a									
6	c									

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a						
2	a	b						
3	bc	c						
4	b	a						
5	a							
6	c							

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

1st step

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a								
2	a	b	a							
3	bc	c	bc							
4	b	a	b							
5	a		a	a						
6	c		c							

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a						
2	a	b						
3	bc	c						
4	b	a						
5	a							
6	c							

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

	<i>use</i>	<i>def</i>	1st step		2nd step		<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
			<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>				
1		a				a				
2	a	b	a		a	bc				
3	bc	c	bc		bc	b				
4	b	a	b		b	a				
5	a		a	a	a	ac				
6	c		c		c					

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a						
2	a	b						
3	bc	c						
4	b	a						
5	a							
6	c							

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

	<i>use</i>	<i>def</i>	1st step		2nd step		3rd step		<i>in</i>	<i>out</i>
			<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>		
1		a				a		a		
2	a	b	a		a	bc	ac	bc		
3	bc	c	bc		bc	b	bc	b		
4	b	a	b		b	a	b	a		
5	a		a	a	a	ac	ac	ac		
6	c		c		c		c			

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a						
2	a	b						
3	bc	c						
4	b	a						
5	a							
6	c							

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

	<i>use</i>	<i>def</i>	1st step		2nd step		3rd step		4th step	
			<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a				a		a		ac
2	a	b	a		a	bc	ac	bc	ac	bc
3	bc	c	bc		bc	b	bc	b	bc	c
4	b	a	b		b	a	b	a	b	ac
5	a		a	a	a	ac	ac	ac	ac	ac
6	c		c		c		c		c	

	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a						
2	a	b						
3	bc	c						
4	b	a						
5	a							
6	c							

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

			1st step		2nd step		3rd step		4th step	
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a				a		a		ac
2	a	b	a		a	bc	ac	bc	ac	bc
3	bc	c	bc		bc	b	bc	b	bc	c
4	b	a	b		b	a	b	a	b	ac
5	a		a	a	a	ac	ac	ac	ac	ac
6	c		c		c		c		c	

			5th step					
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a	c	ac				
2	a	b	ac	bc				
3	bc	c	bc	b				
4	b	a	bc	ac				
5	a		ac	ac				
6	c		c					

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

			1st step		2nd step		3rd step		4th step	
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a				a		a		ac
2	a	b	a		a	bc	ac	bc	ac	bc
3	bc	c	bc		bc	b	bc	b	bc	c
4	b	a	b		b	a	b	a	b	ac
5	a		a	a	a	ac	ac	ac	ac	ac
6	c		c		c		c		c	

			5th step		6th step			
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a	c	ac	c	ac		
2	a	b	ac	bc	ac	bc		
3	bc	c	bc	b	bc	bc		
4	b	a	bc	ac	bc	ac		
5	a		ac	ac	ac	ac		
6	c		c		c			

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation

			1st step		2nd step		3rd step		4th step	
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a				a		a		ac
2	a	b	a		a	bc	ac	bc	ac	bc
3	bc	c	bc		bc	b	bc	b	bc	c
4	b	a	b		b	a	b	a	b	ac
5	a		a	a	a	ac	ac	ac	ac	ac
6	c		c		c		c		c	

			5th step		6th step		7th step	
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a	c	ac	c	ac	c	ac
2	a	b	ac	bc	ac	bc	ac	bc
3	bc	c	bc	b	bc	bc	bc	bc
4	b	a	bc	ac	bc	ac	bc	ac
5	a		ac	ac	ac	ac	ac	ac
6	c		c		c		c	

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Liveness Calculation (Forward)

			1st step		2nd step		3rd step		4th step	
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a				a		a		ac
2	a	b	a		a	bc	ac	bc	ac	bc
3	bc	c	bc		bc	b	bc	b	bc	c
4	b	a	b		b	a	b	a	b	ac
5	a		a	a	a	ac	ac	ac	ac	ac
6	c		c		c		c		c	

			5th step		6th step		7th step	
	<i>use</i>	<i>def</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>
1		a	c	ac	c	ac	c	ac
2	a	b	ac	bc	ac	bc	ac	bc
3	bc	c	bc	b	bc	bc	bc	bc
4	b	a	bc	ac	bc	ac	bc	ac
5	a		ac	ac	ac	ac	ac	ac
6	c		c		c		c	

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Calculation done following forward control-flow edges.

Liveness Calculation (Backward)

	<i>use</i>	<i>def</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>
6	c							
5	a							
4	b	a						
3	bc	c						
2	a	b						
1		a						

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Calculation done following *reverse* control-flow edges.

Liveness Calculation (Backward)

	<i>use</i>	<i>def</i>	1st step		<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>
			<i>out</i>	<i>in</i>				
6	c			c				
5	a		c	ac				
4	b	a	ac	bc				
3	bc	c	bc	bc				
2	a	b	bc	ac				
1		a	ac	c				

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Calculation done following *reverse* control-flow edges.

Liveness Calculation (Backward)

	<i>use</i>	<i>def</i>	1st step		2nd step		<i>out</i>	<i>in</i>
			<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>		
6	c			c		c		
5	a		c	ac	ac	ac		
4	b	a	ac	bc	ac	bc		
3	bc	c	bc	bc	bc	bc		
2	a	b	bc	ac	bc	ac		
1		a	ac	c	ac	c		

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

Calculation done following *reverse* control-flow edges.

Liveness Calculation (Backward)

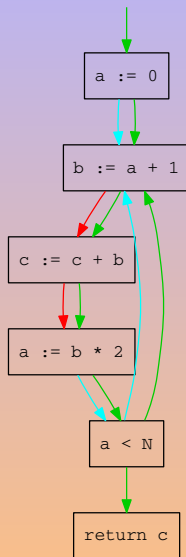
	<i>use</i>	<i>def</i>	1st step		2nd step		3rd step	
			<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>	<i>out</i>	<i>in</i>
6	c			c		c		c
5	a		c	ac	ac	ac	ac	ac
4	b	a	ac	bc	ac	bc	ac	bc
3	bc	c	bc	bc	bc	bc	bc	bc
2	a	b	bc	ac	bc	ac	bc	ac
1		a	ac	c	ac	c	ac	c

$$\begin{aligned} \text{in}[n] &= \text{use}[n] \cup (\text{out}[n] \setminus \text{def}[n]) \\ \text{out}[n] &= \bigcup_{s \in \text{succ}[n]} \text{in}[s] \end{aligned}$$

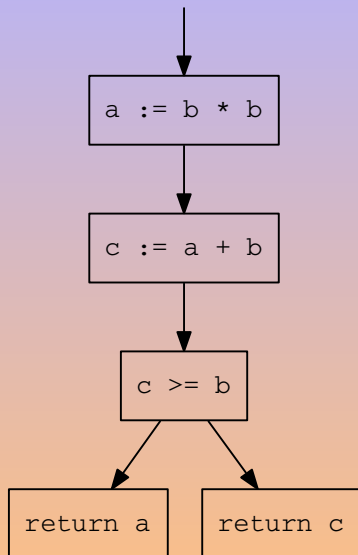
Calculation done following *reverse* control-flow edges.

Control Flow Graph [Appel, 1998]

```
a := 0
L1: b := a + 1
   c := c + b
   a := b * 2
   if a < N goto L1
   return c
```

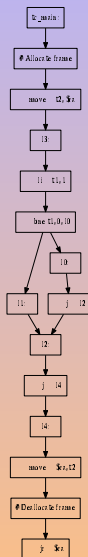


Conservative Approximation

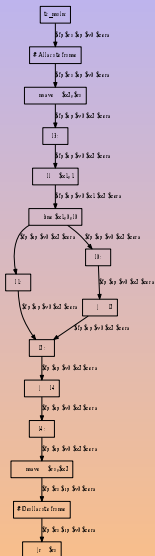


1 | 2

ors' Flowgraph



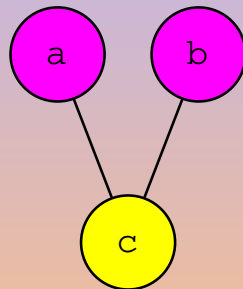
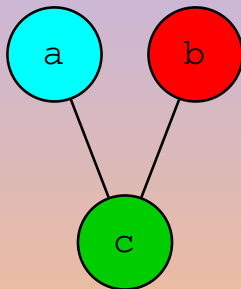
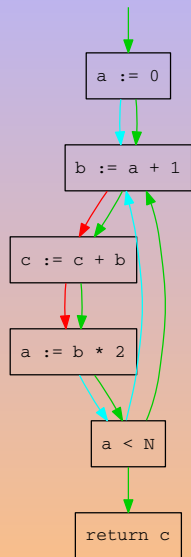
ors' Liveness Graph



Interference Graph

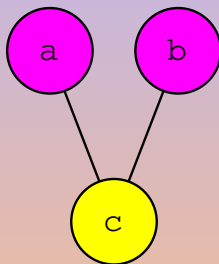
- 1 Control Flow Graph
- 2 Liveness
- 3 Interference Graph**

Interference Graph



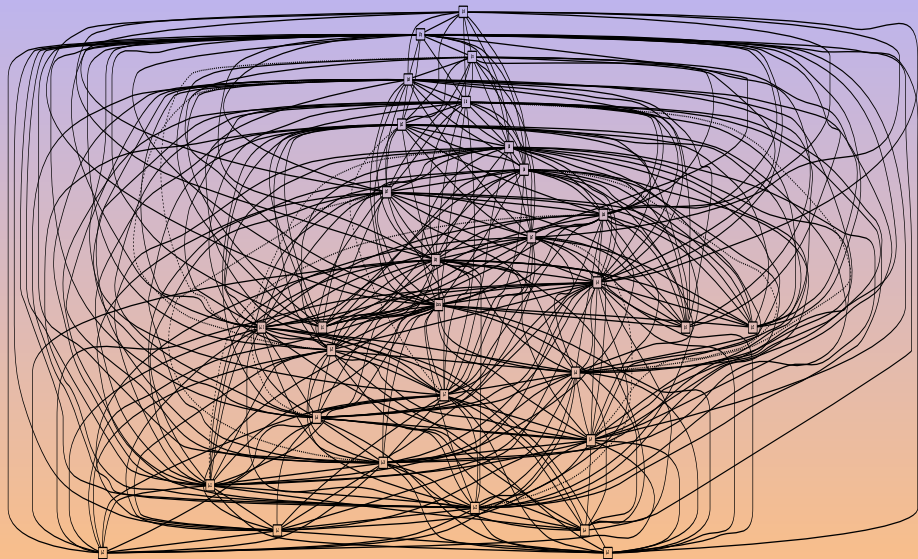
Register Allocation

```
a := 0  
L1: b := a + 1  
   c := c + b  
   a := b * 2  
   if a < N goto L1  
   return c
```

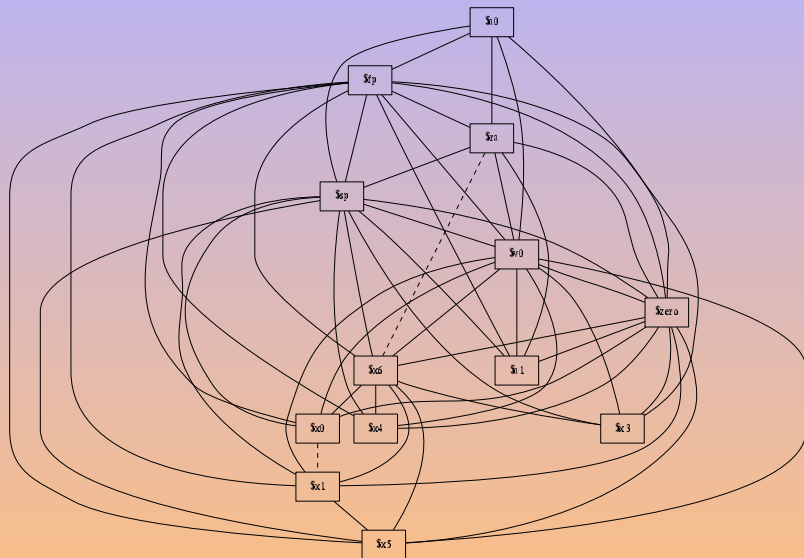


```
r1 := 0  
L1: r1 := r1 + 1  
   r2 := r2 + r1  
   r1 := r1 * 2  
   if r1 < N goto L1  
   return r2
```

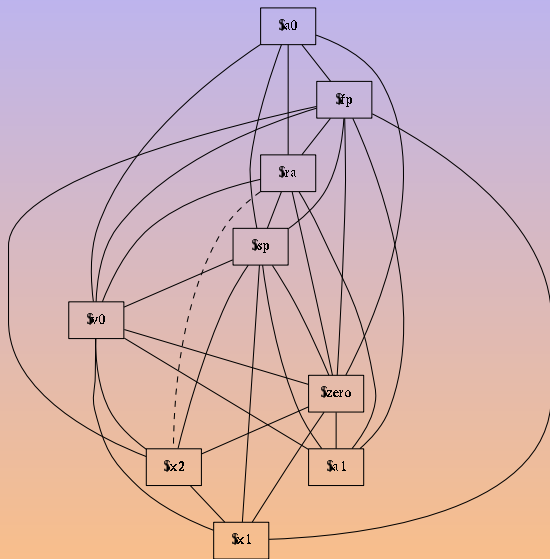
7's Interference Graph



7000's Interference Graph



ors' Interference Graph

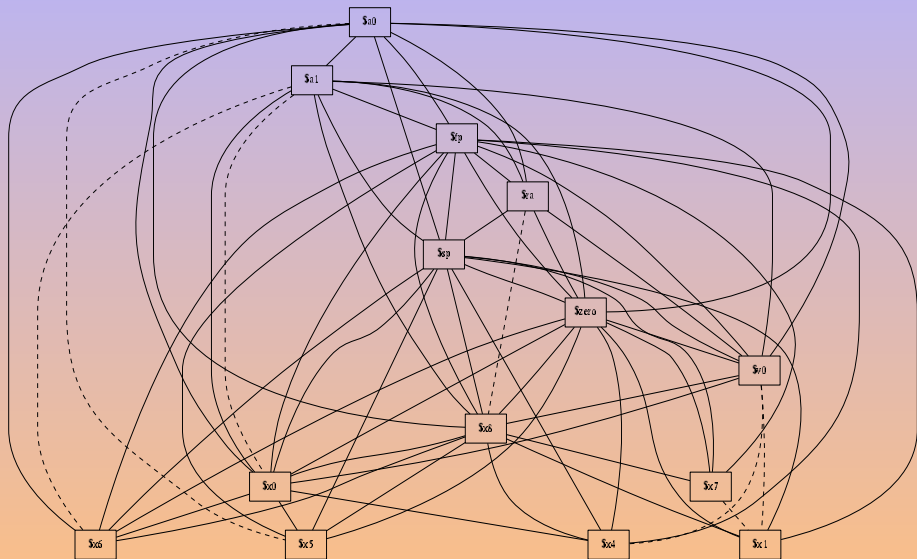



```
let function fact (n : int) : int =  
  if n = 0 then  
    1  
  else  
    n * fact (n - 1)  
in  
  fact (12)  
end
```

fact's Liveness Graph



fact's Interference Graph





Appel, A. W. (1998).

Modern Compiler Implementation in C, Java, ML.

Cambridge University Press.