

Compiler Construction

~ Home Run Example ~

Forewords about Pre-colored nodes

Some nodes are precolored: the real registers

- the frame pointer (\$fp)
- the stack pointer (\$sp)
- the argument registers (\$a0, \$a1, etc.)
- the return value (\$v0, \$v1)
- the return address (\$ra)

They all interfere with each other
They cannot be simplified (infinite degree)

Example (in C)

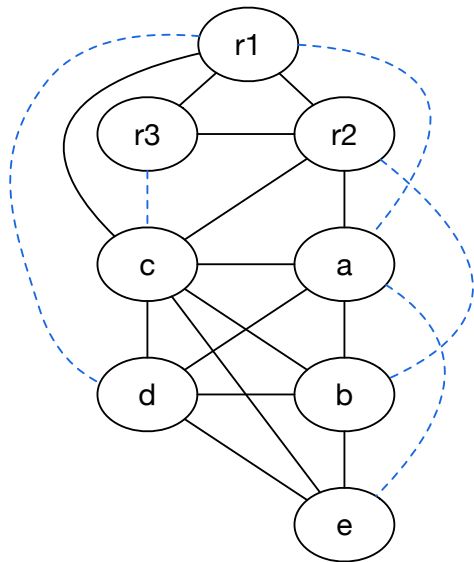
```
int
f (int a, int b) {
    int d = 0;
    int e = a;
    do
    {
        d += b;
        --e;
    } while (e > 0);
    return d;
}
```

1
2
3
4
5
6
7
8
9
10
11

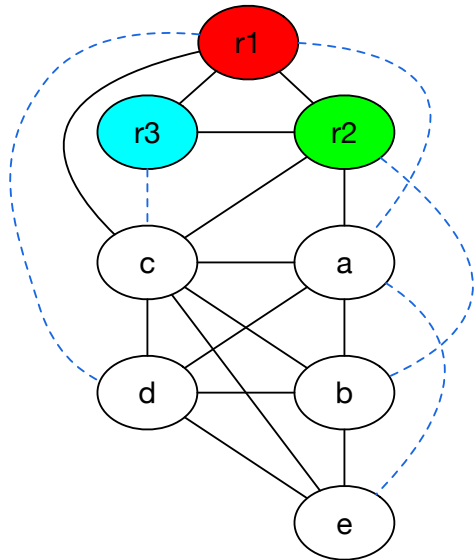
Example cont'd

```
enter: 1
    c := r3 2
    a := r1 3
    b := r2 4
    d := 0 5
    e := a 6
loop: 7
    d := d + b 8
    e := e - 1 9
    if e > 0 goto loop 10
    r1 := d 11
    r3 := c return 12
# liveout: r1, r3 13
```

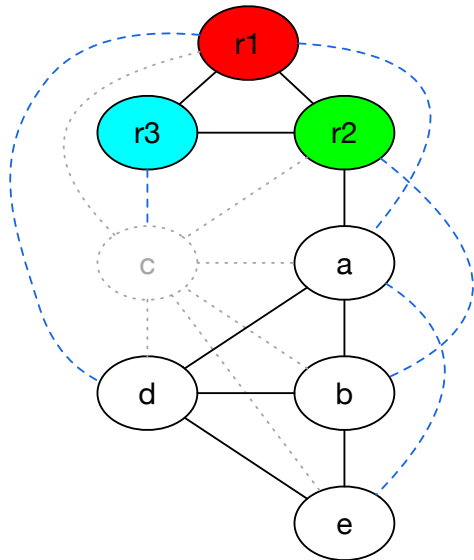
Example



Example

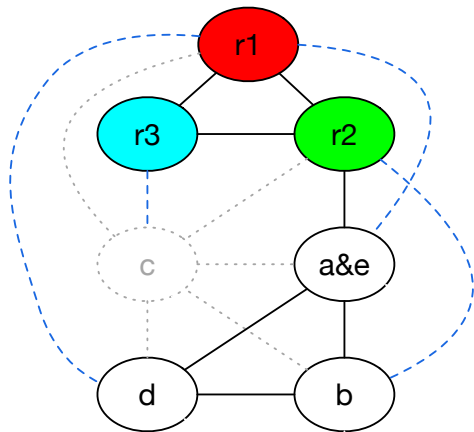


Example



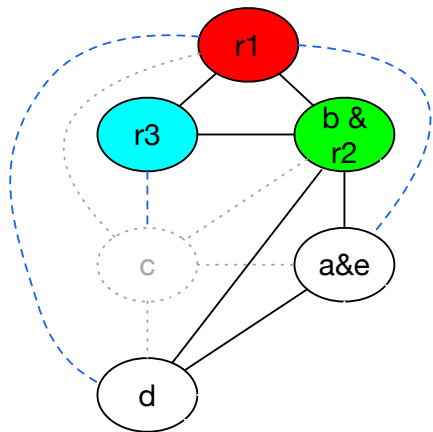
Remove c

Example



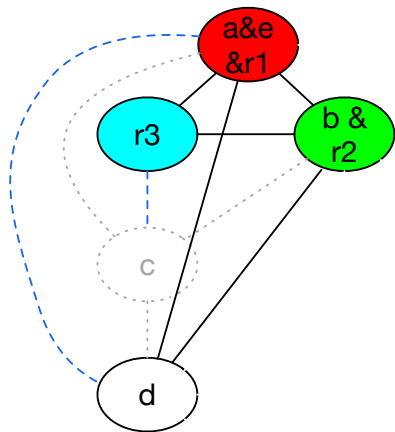
Merge a & e
Remove c

Example



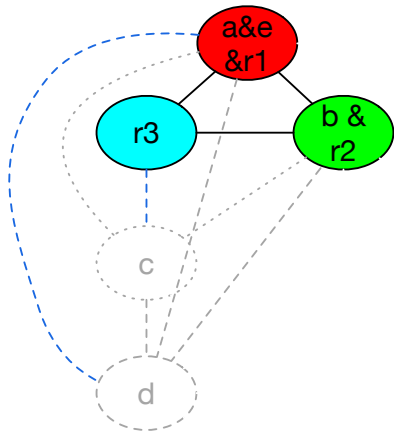
Merge b & r2
Merge a & e
Remove c

Example



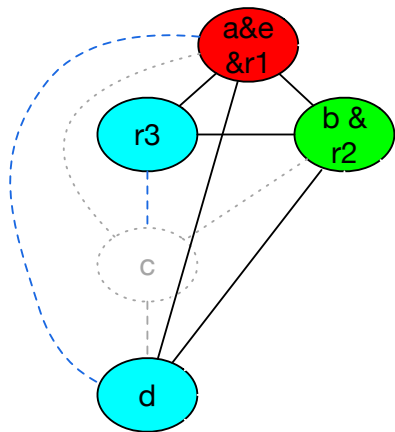
Merge a&e&r1
Merge b & r2
Merge a & e
Remove c

Example



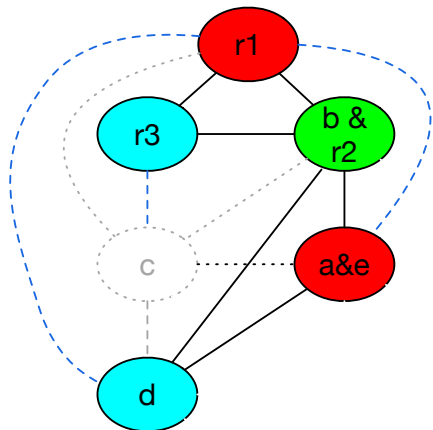
Remove d
Merge a&e&r1
Merge b & r2
Merge a & e
Remove c

Example



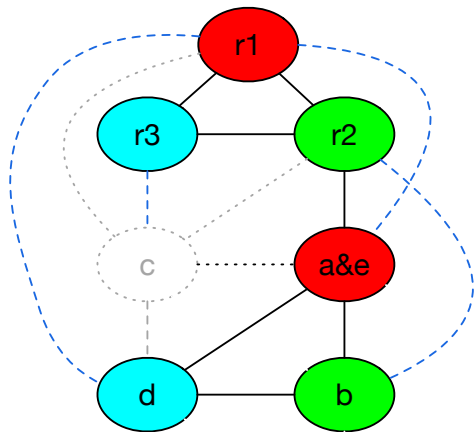
Merge a&e&r1
Merge b & r2
Merge a & e
Remove c

Example



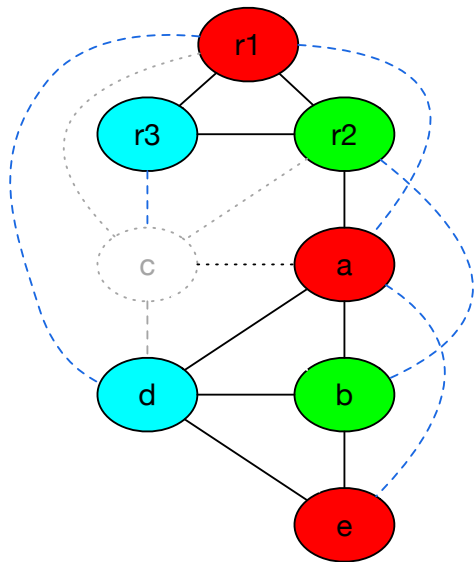
Merge b & r2
Merge a & e
Remove c

Example



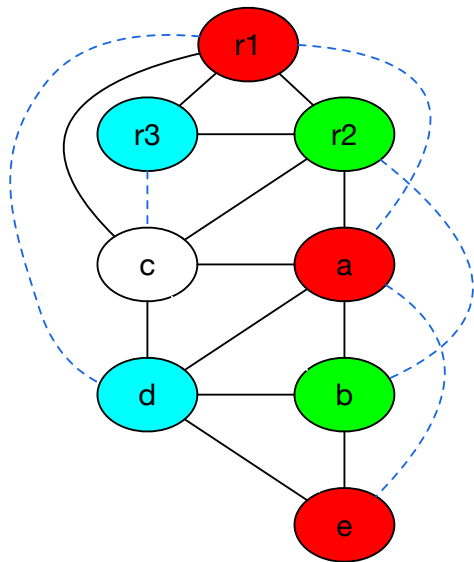
Merge a & e
Remove c

Example



Remove c

Example



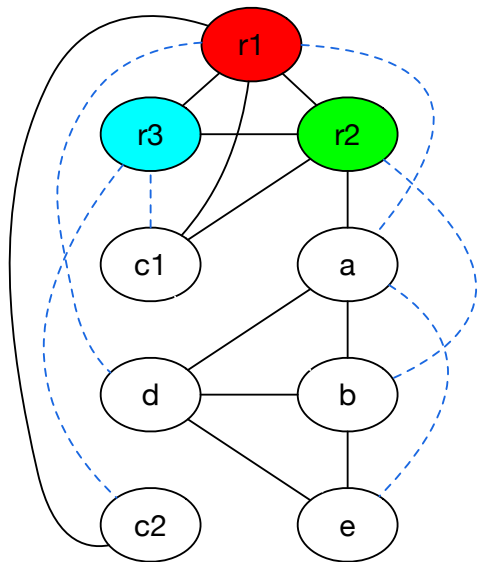
Remove c

Example: rewrite

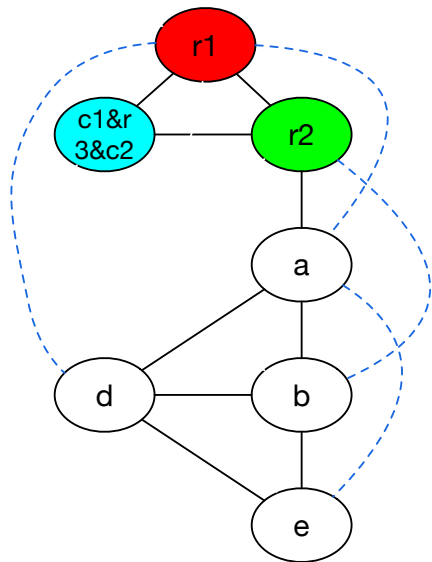
```
enter:
    c1 :=r3
    [sp+8] := c1
    a :=r1
    b :=r2
    d :=0
    e :=a
loop:
    d :=d+b
    e :=e-1
    if e > 0 goto loop
    r1 :=d
    c2 := [sp+8]
    r3 :=c2
    return
# liveout: r1, r3
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

Example

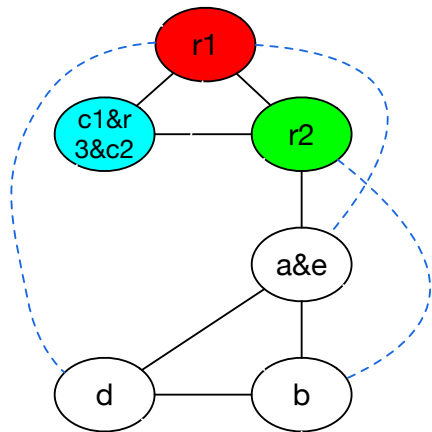


Example



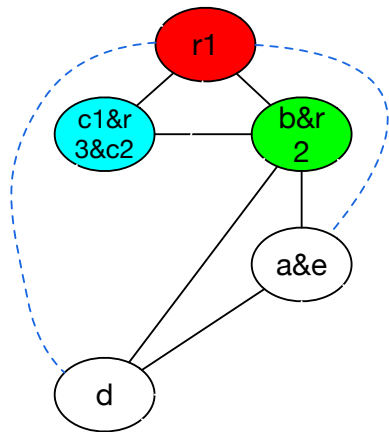
Merge c1&r3&c2
Merge c1&r3

Example



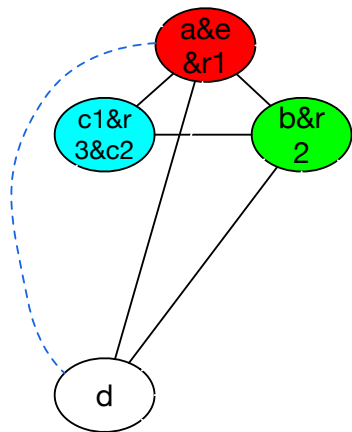
Merge a&e
Merge c1&r3&c2
Merge c1&r3

Example



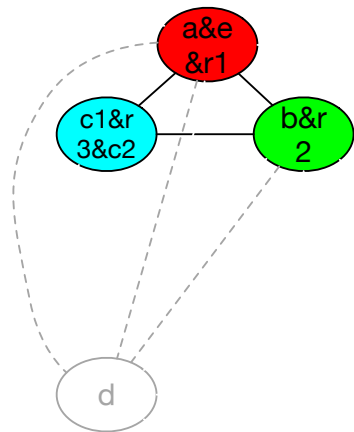
Merge b & r2
Merge a&e
Merge c1&r3&c2
Merge c1&r3

Example



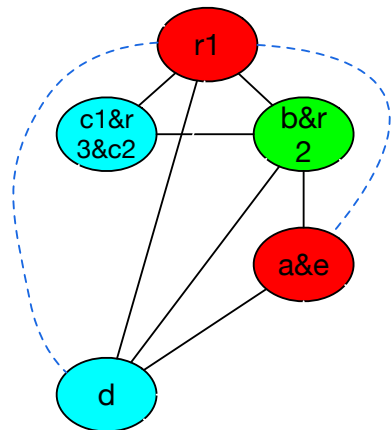
Merge a&e&r1
Merge b & r2
Merge a&e
Merge c1&r3&c2
Merge c1&r3

Example



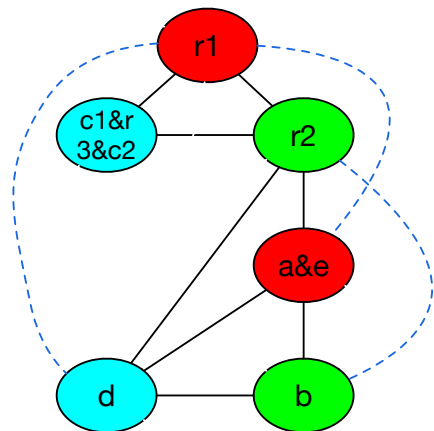
Remove d
Merge a&e&r1
Merge b & r2
Merge a&e
Merge c1&r3&c2
Merge c1&r3

Example



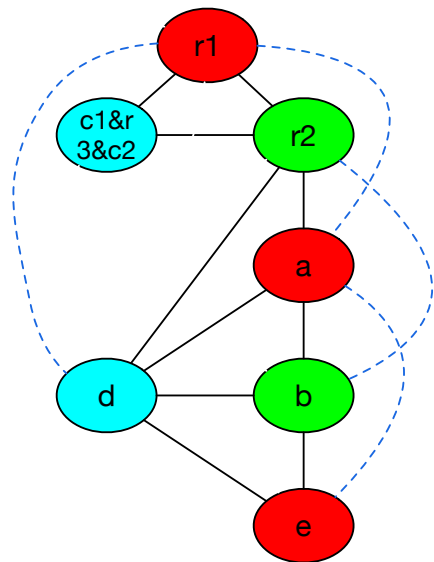
Merge b & r2
Merge a&e
Merge c1&r3&c2
Merge c1&r3

Example



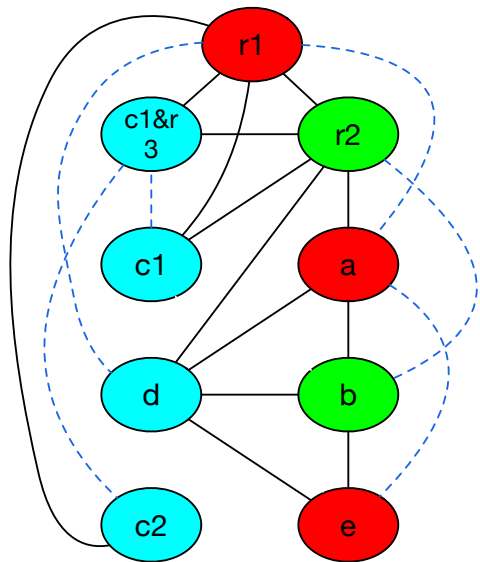
Merge a&e
Merge c1&r3&c2
Merge c1&r3

Example



Merge c1&r3&c2
Merge c1&r3

Example



Result

enter:

```
c1 := r3
[sp+8] := c1
a := r1
b := r2
d := 0
e := a
```

loop:

```
d:=d+b
e:=e-1
if e > 0 goto loop
r1 := d
c2 := [sp+8]
r3 := c2
return
```

liveout: r1, r3

enter:

```
r3 := r3
[sp+8] := r3
r1 := r1
r2 := r2
r3 := 0
r1 := r1
```

loop:

```
r3 := r3 + r2
r1 := r1 - 1
if r1 > 0 goto loop
r1 := r3
r3 := [sp+8]
r3 := r3
return
```

liveout: r1, r3

enter:

```
[sp+8] := r3

r3 := 0
```

loop:

```
r3 := r3 + r2
r1 := r1 - 1
if r1 > 0 goto loop
r1 := r3
r3 := [sp+8]

return
```

liveout: r1, r3

Summary

