

```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

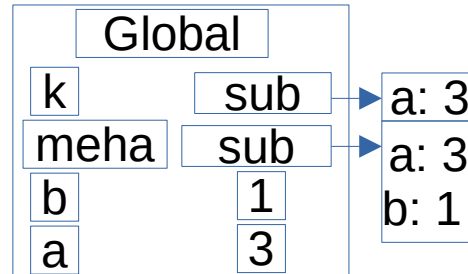
k(a, meha, meha);
print(a, b)

```

Alcance Dinamico y Clausura con Asociacion Profunda

Inicialmente las variables Globales son declaradas.

Las clausuras se realizan Al definir las subrutinas



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

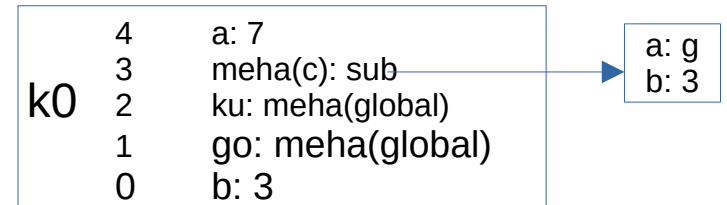
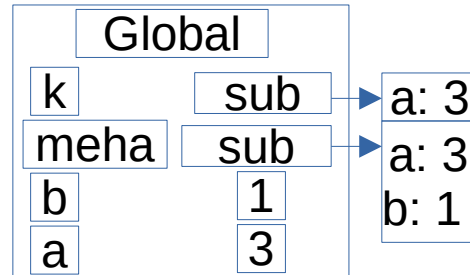
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

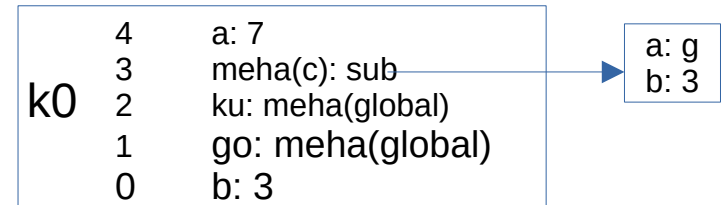
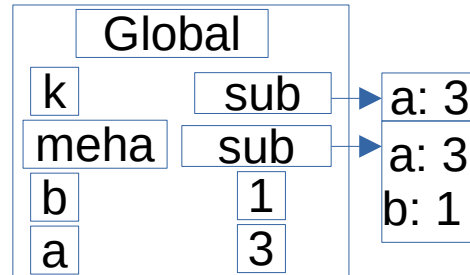
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

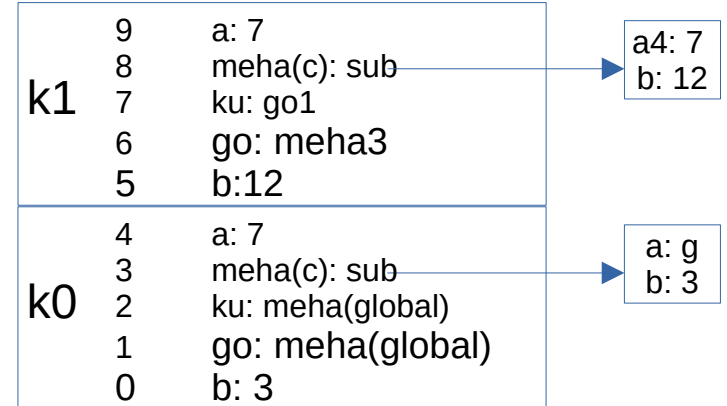
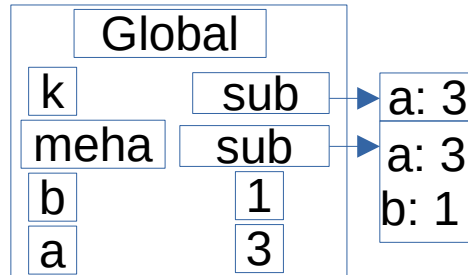
sub meha(int c) {
    b := a + c;
}
sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}
k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

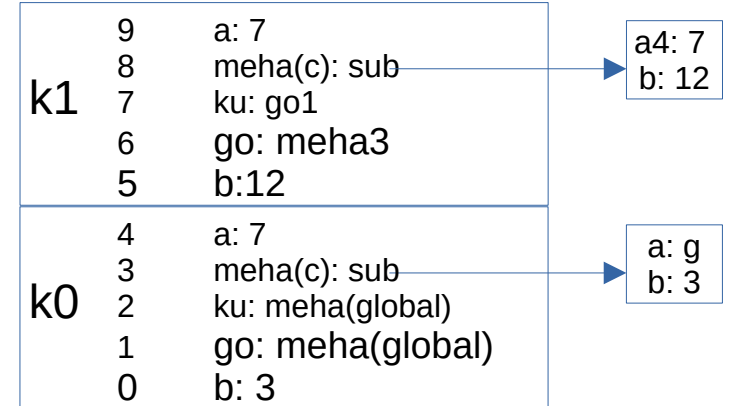
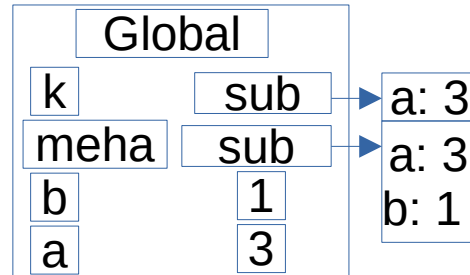
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

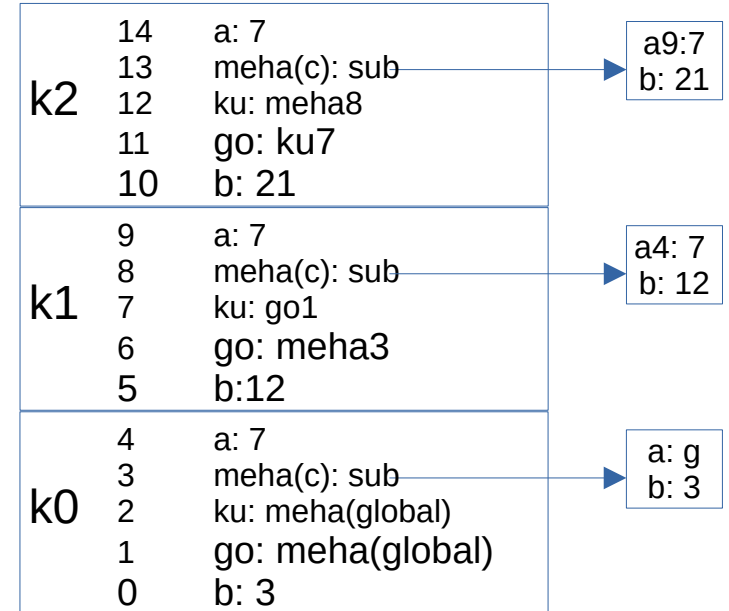
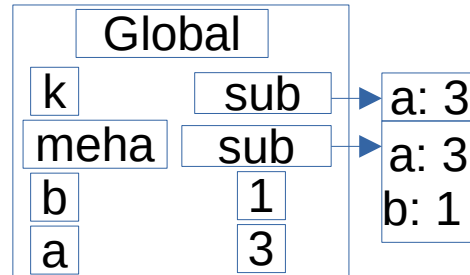
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

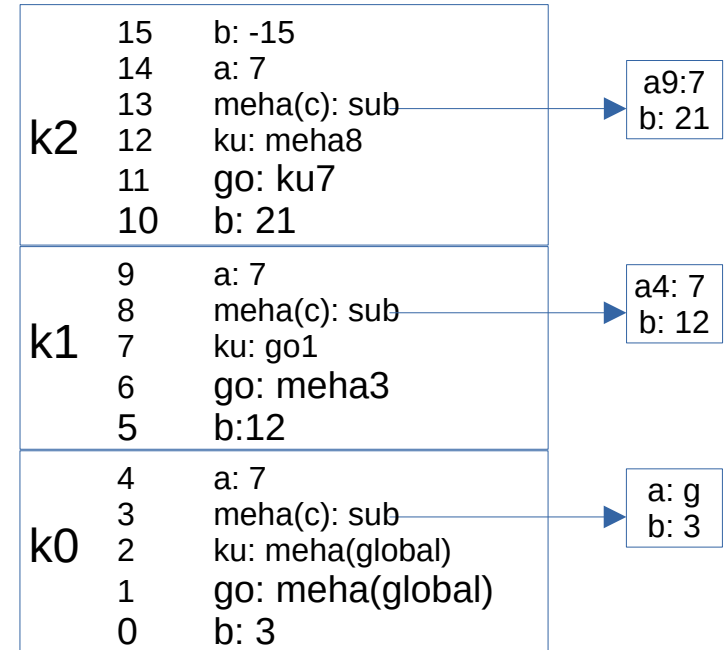
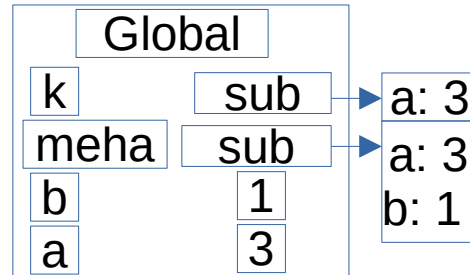
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

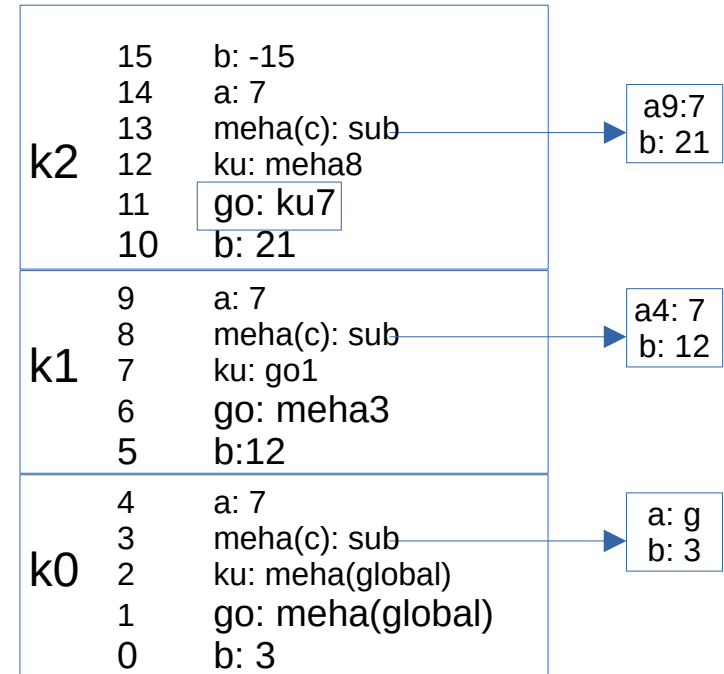
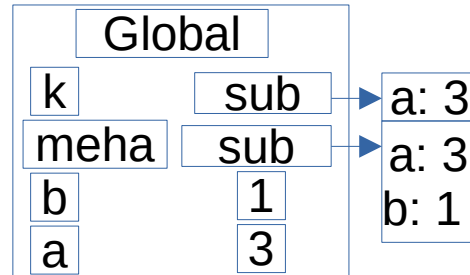
sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }
    print(a, b)
}

k(a, meha, meha);
print(a, b)

```




```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

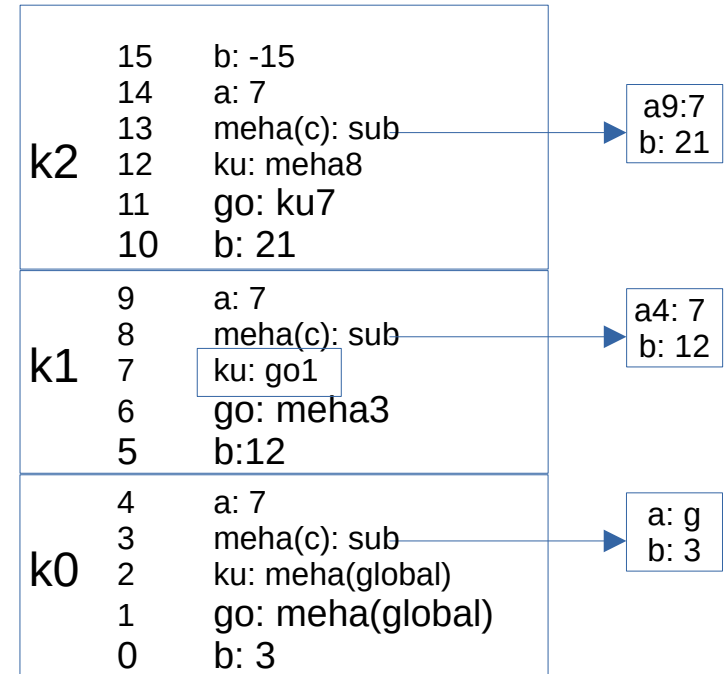
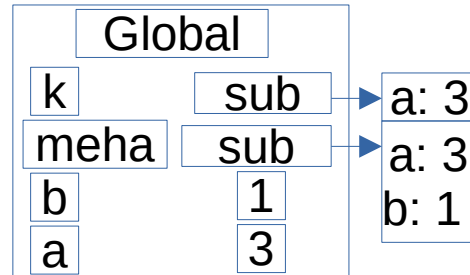
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

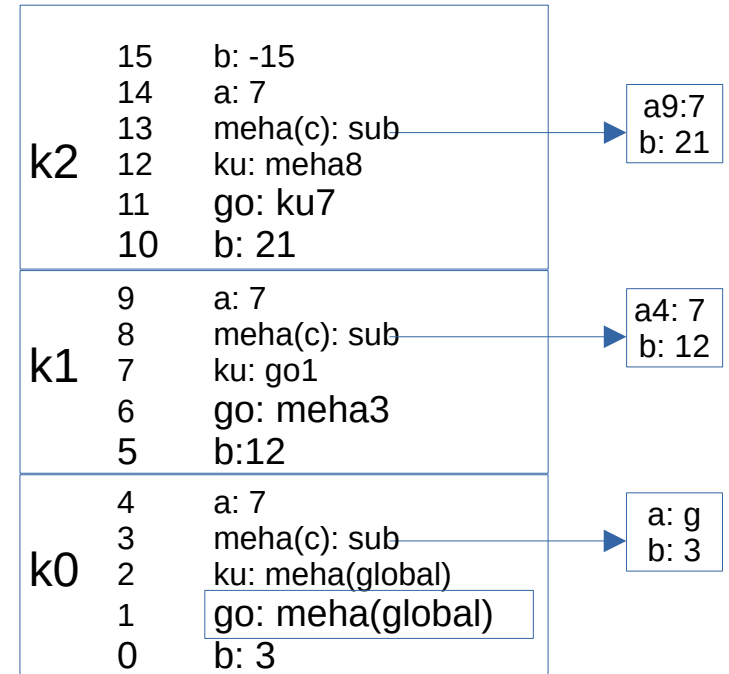
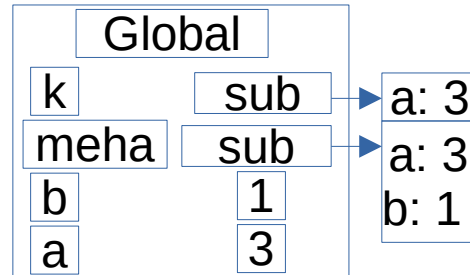
sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }
    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```
int a = 2 + 1, b = 1;
```

```
sub meha(int c) {  
    b := a + c;  
}
```

```
sub k (int b, sub go, sub ku) {
```

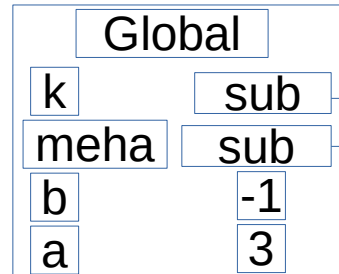
```
    sub meha(int c) {  
        a := b - c;  
    }
```

```
    int a = 6 + 1;  
    if (b < 3 * (2 + 1)) {  
        k(b + 3 * (2 + 1), meha, go);  
    } else if (b < 6 * (2 + 1)) {  
        k(b + 3 * (2 + 1), ku, meha);  
    } else {  
        int b = 6 - b  
        go(a + b);  
        ku(a - b);  
    }
```

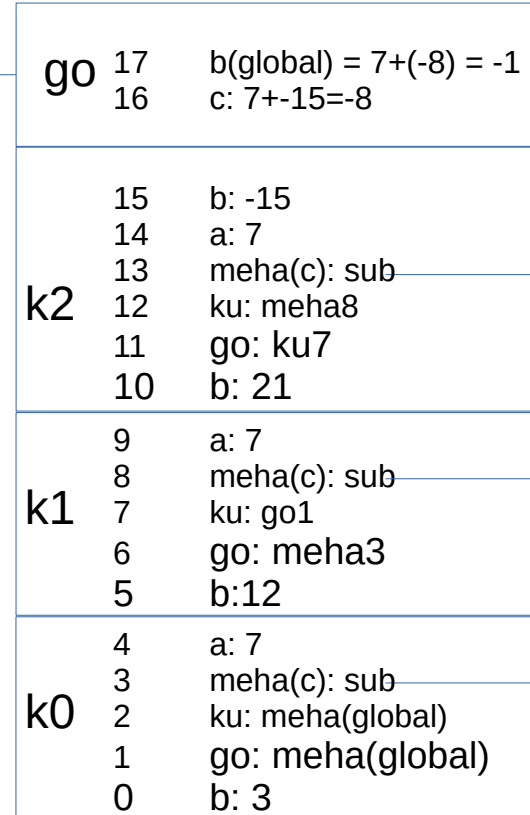
```
    print(a, b)
```

```
}  
k(a, meha, meha);  
print(a, b)
```

b(global) como estaba en la clausura
De meha entonces cambia su valor



a: 3
a: 3
b: 1



a9: 7
b: 21

a4: 7
b: 12

a: g
b: 3

```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

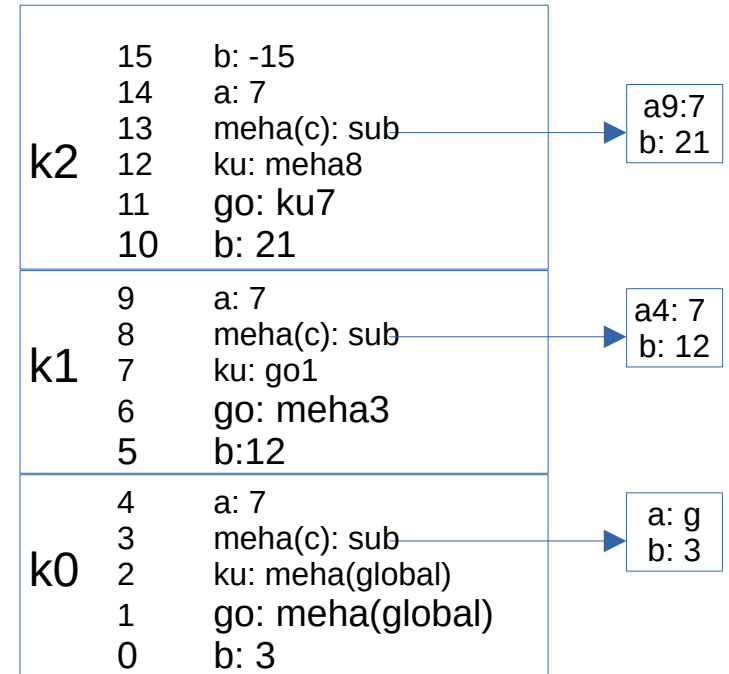
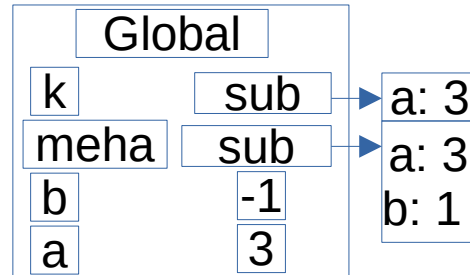
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

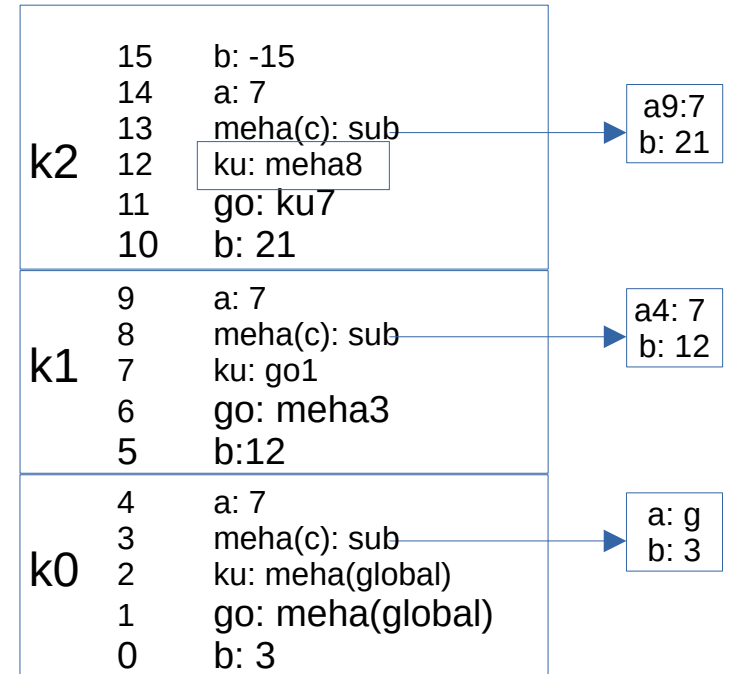
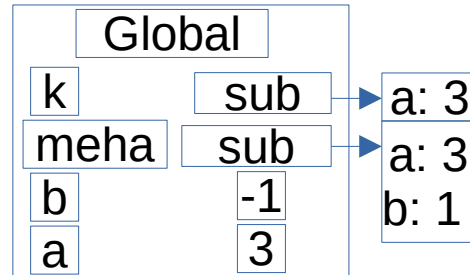
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

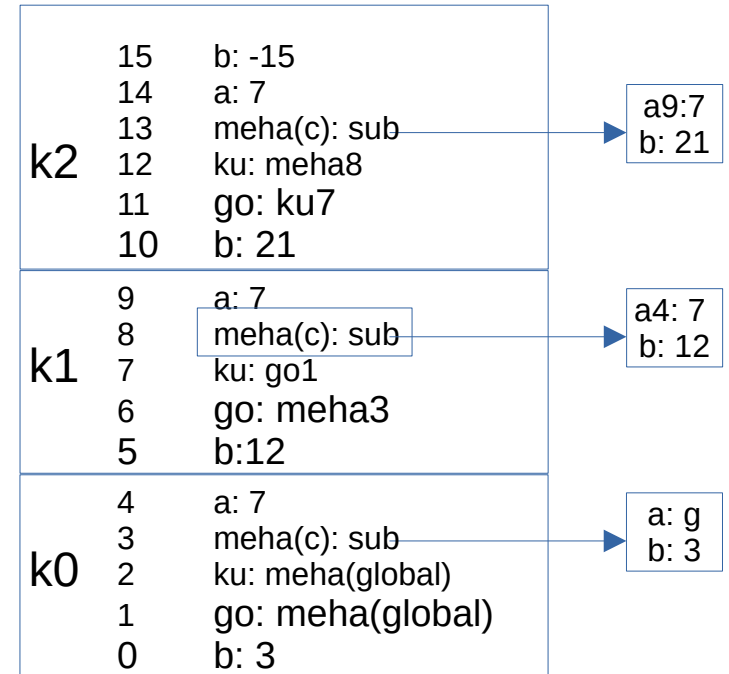
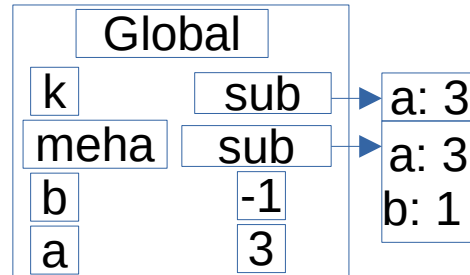
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

```

```

    sub meha(int c) {
        a := b - c;
    }

```

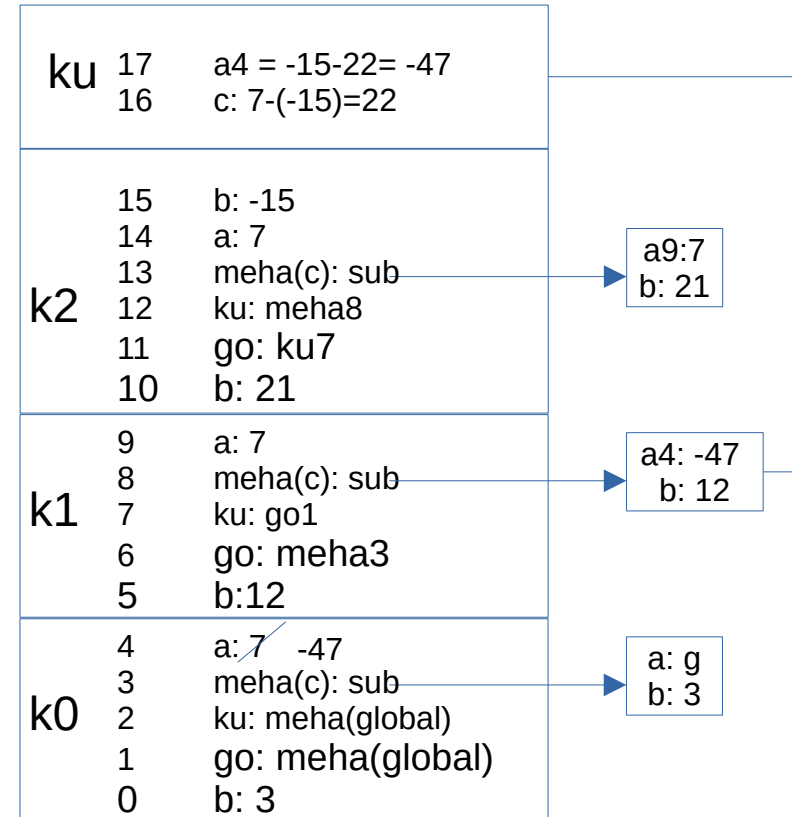
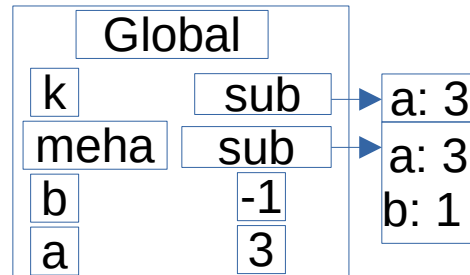
```

int a = 6 + 1;
if (b < 3 * (2 + 1)) {
    k(b + 3 * (2 + 1), meha, go);
} else if (b < 6 * (2 + 1)) {
    k(b + 3 * (2 + 1), ku, meha);
} else {
    int b = 6 - b
    go(a + b);
    ku(a - b);
}

print(a, b)
}

k(a, meha, meha);
print(a, b)

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

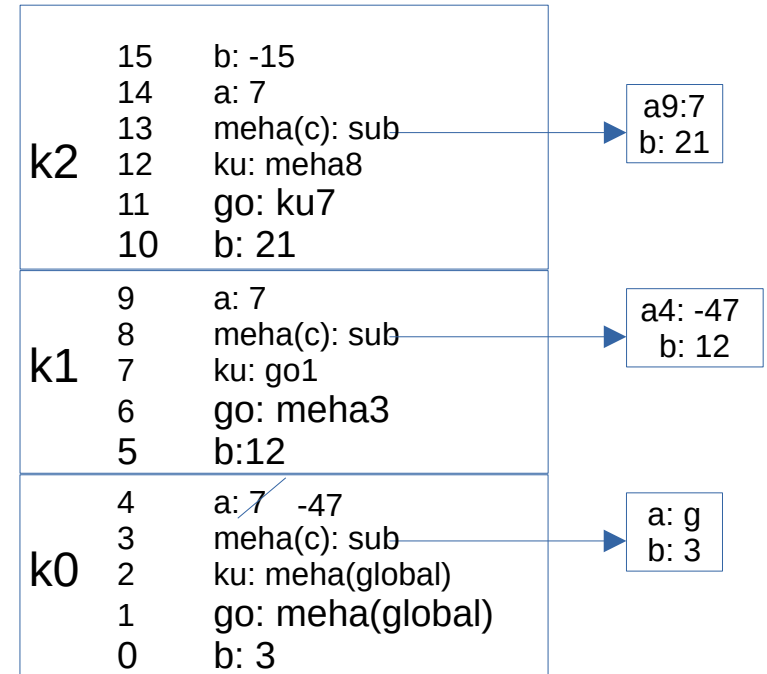
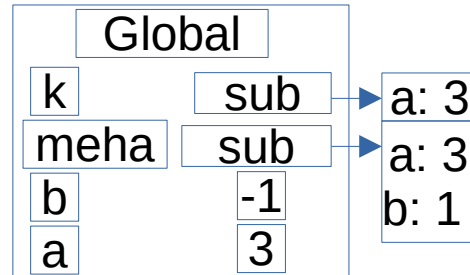
    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

```




```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

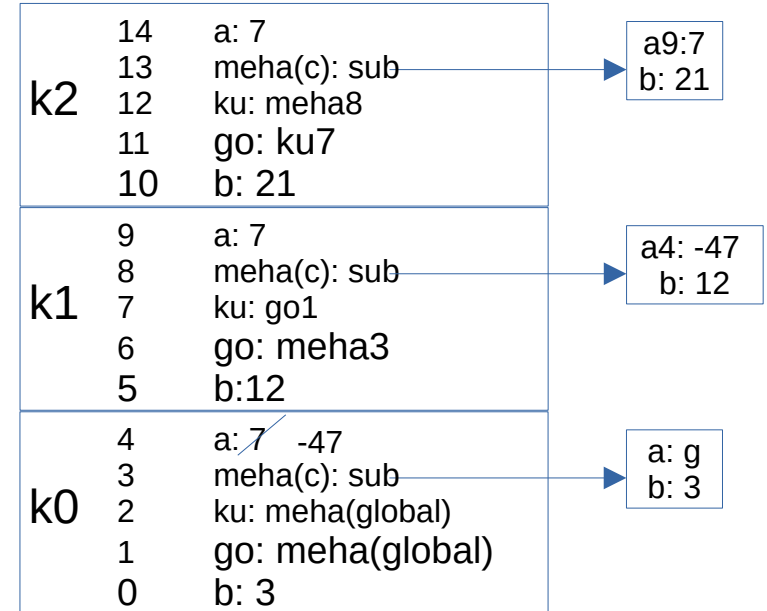
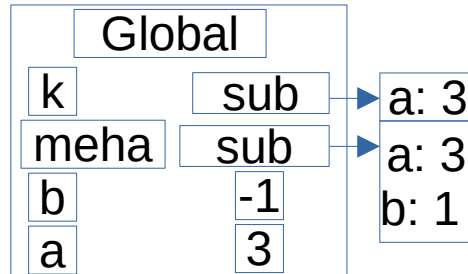
    print(a, b)
}

k(a, meha, meha);
print(a, b)

```

Salida

7 21



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

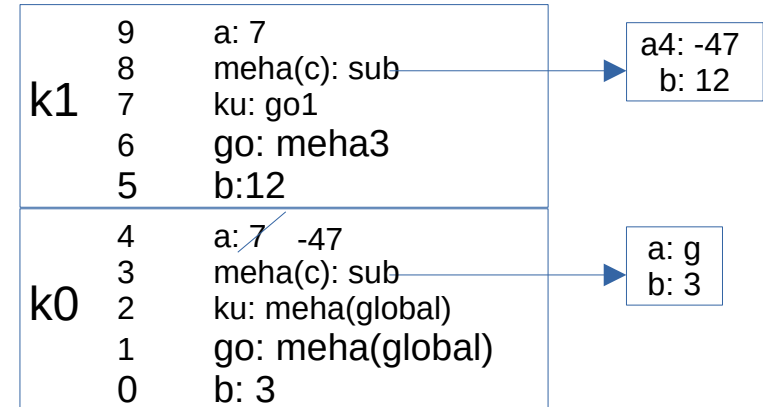
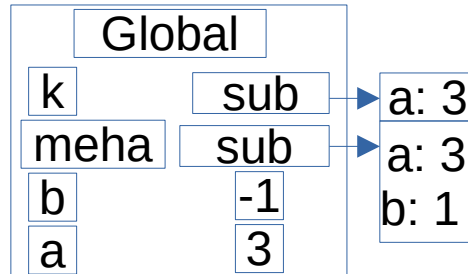
    print(a, b)
}

k(a, meha, meha);
print(a, b)

```

Salida

7 21



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }
    print(a, b)
}
k(a, meha, meha);
print(a, b)

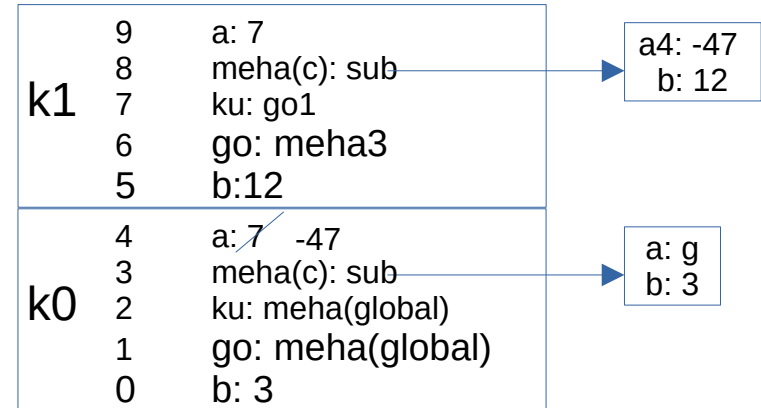
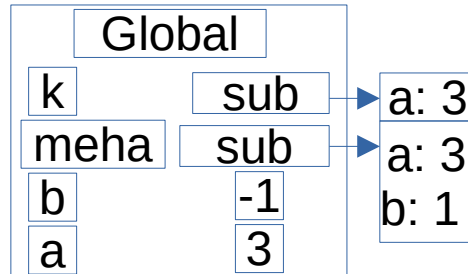
```

Salida

```

7  21
7  12

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

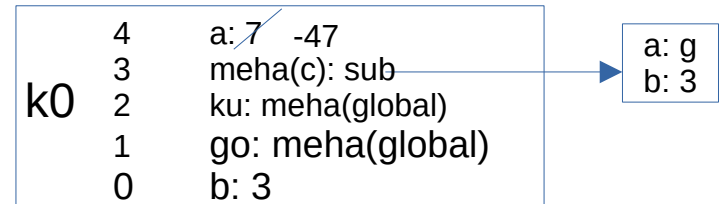
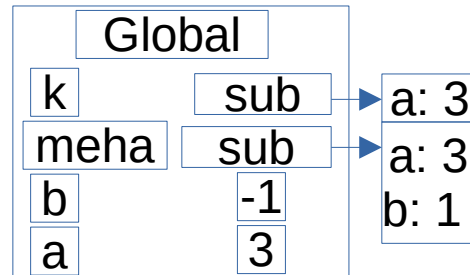
```

Salida

```

7 21
7 12

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }
    print(a, b)
}

k(a, meha, meha);
print(a, b)

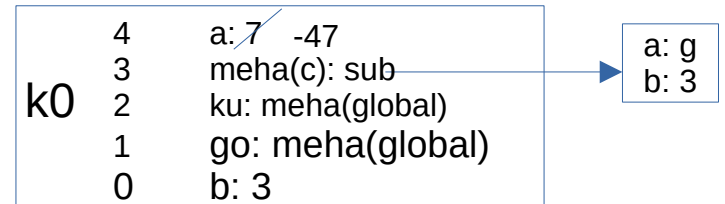
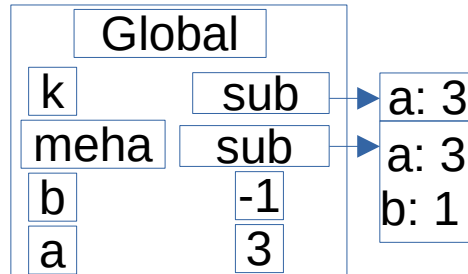
```

Salida

```

7  21
7  12
-47 3

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

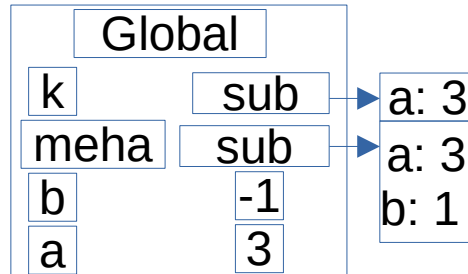
```

Salida

```

7  21
7  12
-47 3

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}

sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }

    print(a, b)
}

k(a, meha, meha);
print(a, b)

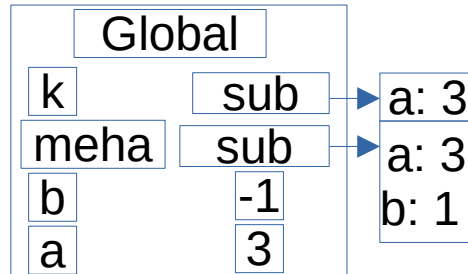
```

Salida

```

7  21
7  12
-47 3
3  -1

```



```

int a = 2 + 1, b = 1;

sub meha(int c) {
    b := a + c;
}
sub k (int b, sub go, sub ku) {

    sub meha(int c) {
        a := b - c;
    }

    int a = 6 + 1;
    if (b < 3 * (2 + 1)) {
        k(b + 3 * (2 + 1), meha, go);
    } else if (b < 6 * (2 + 1)) {
        k(b + 3 * (2 + 1), ku, meha);
    } else {
        int b = 6 - b
        go(a + b);
        ku(a - b);
    }
    print(a, b)
}
k(a, meha, meha);
print(a, b)

```

Salida

```

7  21
7  12
-47 3
3  -1

```

Finalizado