

```
int x=0
```

```
f()
```

```
{
```

```
  int k;  
  read k;
```

```
  if (x == 0)  
    x = g();  
  else  
    x = 10;
```

```
  int g()
```

```
  {
```

```
    if (k != 0 )  
      return k;  
    else  
      return 5;
```

```
  }
```

```
}
```

```
main()
```

```
{
```

```
  int x= 3;  
  f();
```

```
}
```

0 set 0, 3	Beginning of AR(main)
1 set 1, 7	End of AR(main)
2 set 2, 1	global x=0
3 set 6, 2	main: x=3
4 set D[1], 5 + 5	RP
5 set D[1]+1, D[0]	DL
6 set D[1]+2, fp(1)	SL
7 set 0, D[1]	CP
8 set 1, D[1] + 4	FP  AR(f) =4
9 jump 11	jump to code for f
10 halt	STOP
11 set fp(0)+ 3, READ	Code for f(): read k
12 jumpt 22, D[2] <> 0	Jump if x <>0
13 set 1, D[1] + 1	space for ret. value
14 set D[1], 15 + 5	RP
15 set D[1]+1, D[0]	DL
16 set D[1]+2, fp(0)	SL
17 set 0, D[1]	CP
18 set 1, D[1] + 3	FP  AR(g) =3
19 jump 26	jump to code for g
20 set 2, D[D[1]]	x = g()
21 jump 23	skip else
22 set 2, 10	x = 2
23 set 1, D[0]	FP
24 set 0, D[D[0]+1]	CP
25 jump D[D[1]]	Return to caller
26 jumpt 31, D[fp(1) + 3] = 0	Jump if k = 0
27 set D[0]-1, D[fp(1) + 3]	Save return value
28 set 1, D[0]-1	FP
29 set 0, D[D[0]+1]	CP
30 jump D[D[1]+1]	Return to caller
31 set D[0]-1, 5	Save return value
32 set 1, D[0]-1	FP
33 set 0, D[D[0]+1]	CP
34 jump D[D[1]+1]	Return to caller