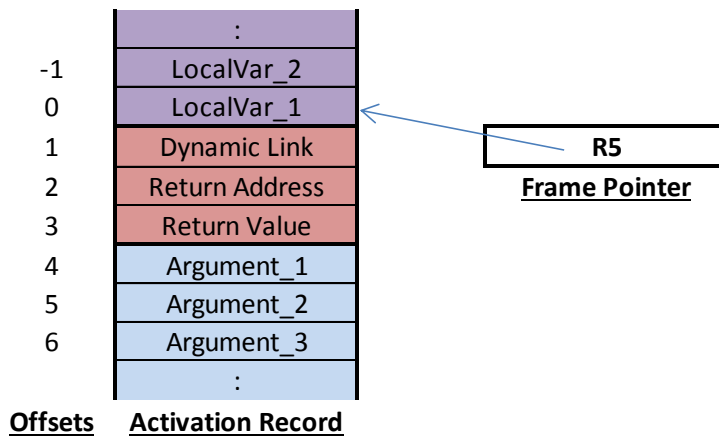


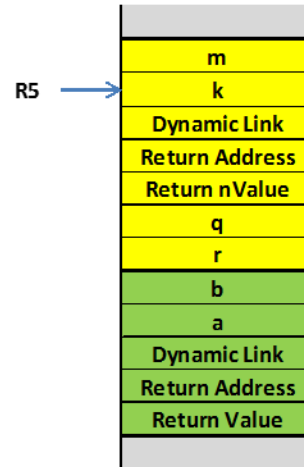
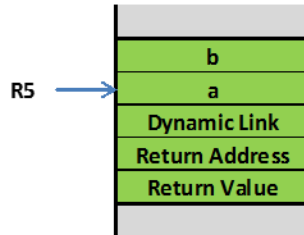
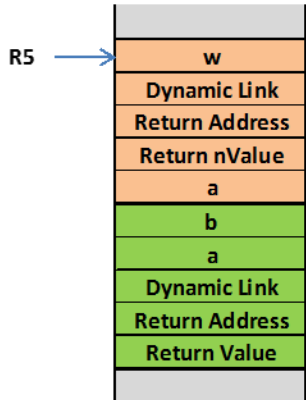
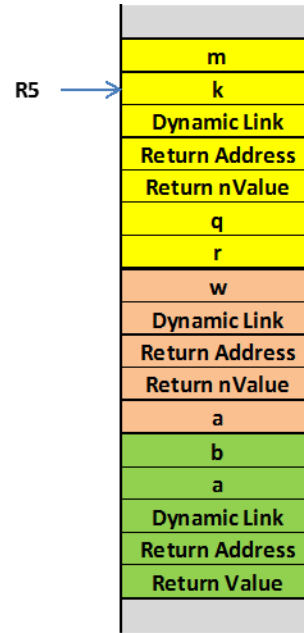
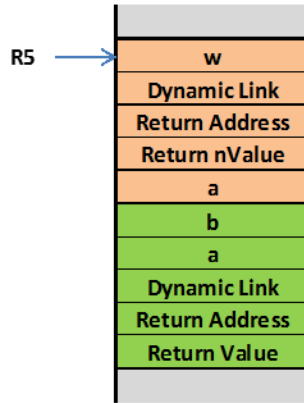
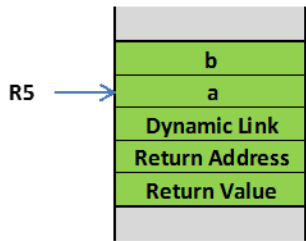
Figure 14.4 Page 386

```
int main()
{
    int a;
    int b;
    ;
    ;
    b = Watt(a);
    b = Volta(a, b);
}
```

```
int Watt(int a)
{
    int w;
    ;
    ;
    w = Volta(w, 10);
    return w;
}
```

```
int Volta(int q, int r)
{
    int k;
    int m;
    ;
    return k
}
```





	<u>Activation Records</u>		<u>R5</u> <u>Offsets</u>
xDFEF	m	<-- R6	-1
xDFF0	K	<-- R5	0
xDFF1	Dynamic Link = xDFF6		1
xDFF2	Return Address		2
xDFF3	Return Value		3
xDFF4	q		4
xDFF5	r	Volta()	5
xDFF6	w		0
xDFF7	Dynamic Link = xDFFC		1
xDFF8	Return Address		2
xDFF9	Return Value		3
xDFFA	a	Watt()	4
xDFFB	b		-1
xDFFC	a		0
xDFFD	Dynamic Link		1
xDFFE	Return Address		2
xDFFF	Return Value	main()	3
xE000			

After function Watt() calls function Volta()

```

int Watt(int a)
{
    int w;
    ;
    ;
    w = Volta(w, 10);
    return w;
}

```

Watt

```

AND  R0, R0, #0
ADD  R0, R0, #10
ADD  R6, R6, #-1
STR  R0, R6, #0    ;Push 10
LDR  R0, R5, #0
ADD  R6, R6, #-1
STR  R0, R6, #0    ;Push w
JSR  VOLTA
-----
LDR  R0, R6, #0
ADD  R6, R6, #1
STR  R0, R5, #0    ;Pop w
ADD  R6, R6, #2    ;Clear 2 arguments

```

Volta

```

ADD  R6, R6, #-1    ;for ret value
ADD  R6, R6, #-1
STR  R7, R6, #0    ;Push R7
ADD  R6, R6, #-1
STR  R5, R6, #0    ;Push R5
ADD  R5, R6, #-1    ;Reset R5
ADD  R6, R6, #-2    ;locals k, m
-----
LDR  R0, R5, #0
STR  R0, R5, #3    ;Ret value
ADD  R6, R5, #1    ;Clear locals
LDR  R5, R6, #0
ADD  R6, R6, #1    ;Pop R5
LDR  R7, R6, #0
ADD  R6, R6, #1    ;Pop R7
RET

```

