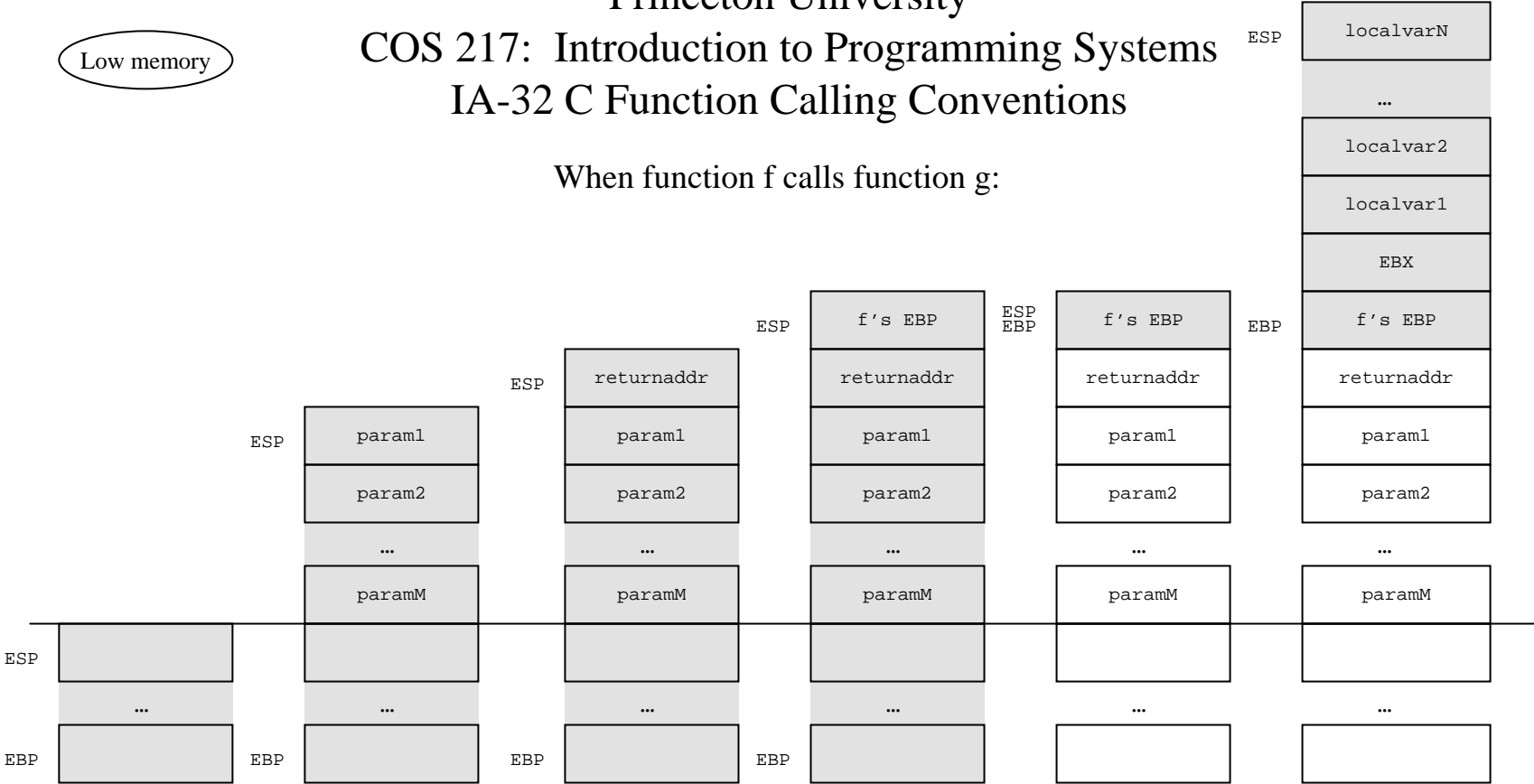# Princeton University
COS 217: Introduction to Programming Systems
IA-32 C Function Calling Conventions

Low memory

When function f calls function g:

					ESP	localvarN
						...
						localvar2
						localvar1
						EBX
		ESP	f's EBP	ESP EBP	f's EBP	EBP	f's EBP
	ESP	returnaddr	returnaddr	returnaddr	returnaddr
ESP	param1	param1	param1	param1	param1
	param2	param2	param2	param2	param2
	...	...	...	...	...
	paramM	paramM	paramM	paramM	paramM
ESP					
...	...	...	...	...	...
EBP	EBP	EBP	EBP		

(1) f pushes parameters:

```
pushl paramM
...
pushl param2
pushl param1
```
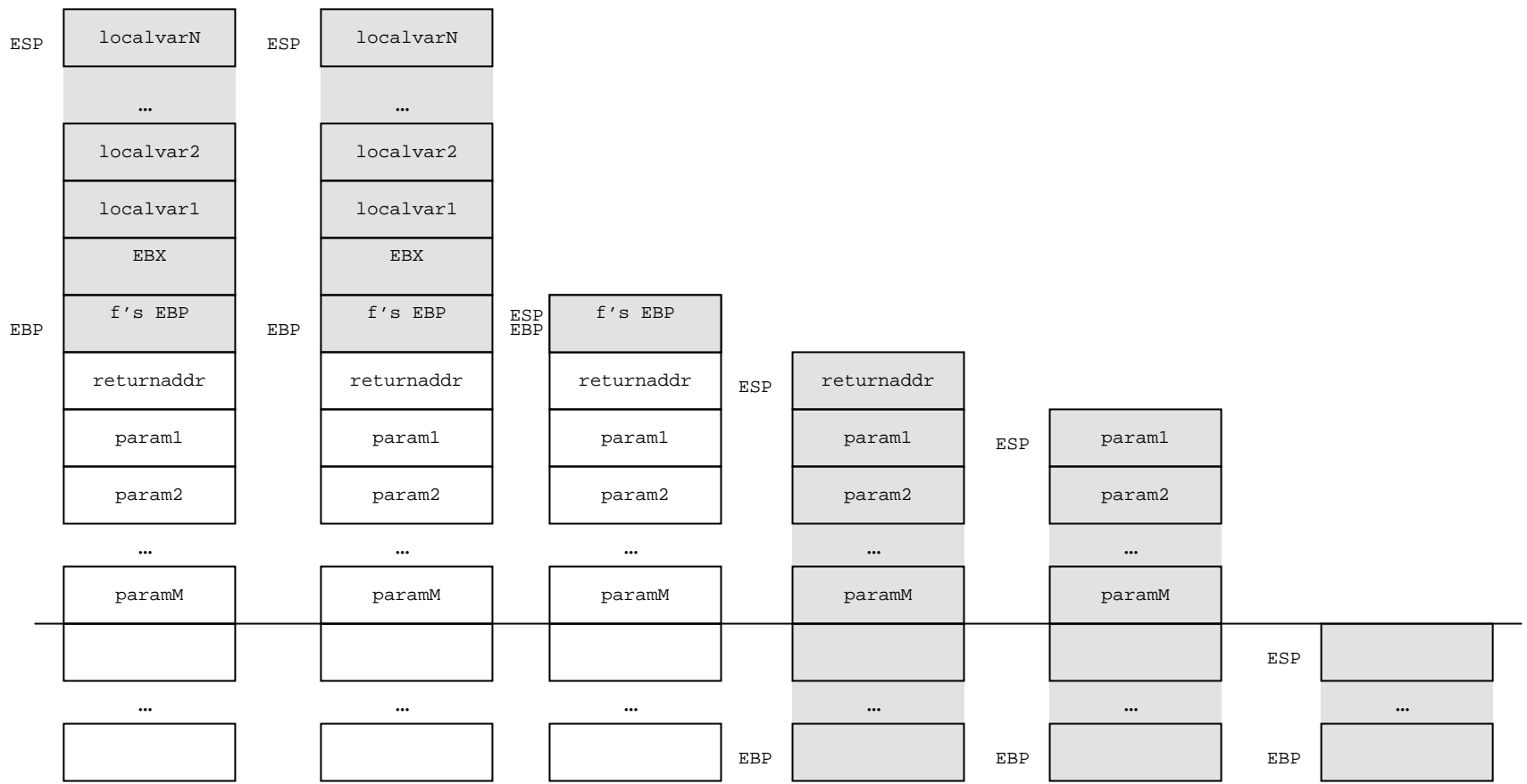
(2) f executes call instruction:

```
call g
```

(3) g pushes f's EBP:

```
pushl %ebp
```

(4) g moves ESP to EBP:

```
movl %esp,%ebp
```

(5) If necessary, g pushes EBX, ESI, or EDI and local variables:

```
pushl %ebx
pushl localvar1
pushl localvar2
...
pushl localvarN
```

High memory

ESP | localvarN
…
localvar2
localvar1
EBX
EBP | f's EBP
returnaddr
param1
param2
…
paramM
…

ESP | localvarN
…
localvar2
localvar1
EBX
EBP | f's EBP
returnaddr
param1
param2
…
paramM
…

ESP EBP | f's EBP
returnaddr
param1
param2
…
paramM
…

ESP | returnaddr
param1
param2
…
paramM
…
EBP

ESP | param1
param2
…
paramM
…
EBP

param1
param2
…
paramM
ESP
…
EBP

(6) g uses parameters and local variables to compute a return value, and moves that value to EAX:

```
movl 8(%ebp), ???
movl 12(%ebp), ???
…
movl –8(%ebp), ???
movl –12(%ebp), ???
…
movl ???, %eax
```

(7) If necessary, g restores EDI, ESI, or EBX

```
movl –4(%ebp),%ebx
```

(8) g moves EBP to ESP:

```
movl %ebp,%esp
```

(9) g pops from stack into EBP:

```
popl %ebp
```

(10) g executes ret instruction:

```
ret
```

(11) f pops actual parameters from stack, and uses return value in EAX:

```
addl $4*M, %esp
movl %eax, ???
```