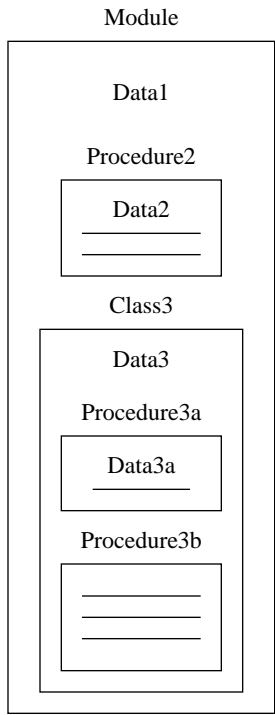
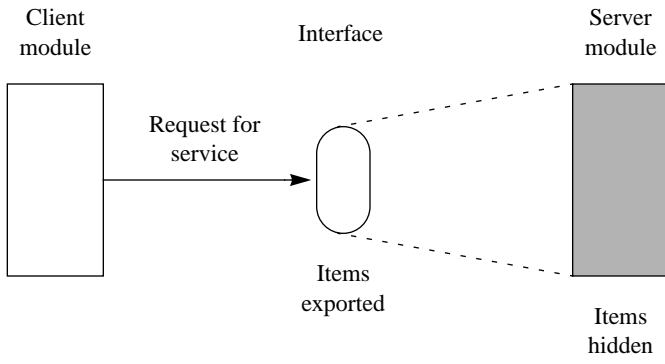


Chapter 3

# *Modules and Interfaces*



**Figure 3.1**  
A module containing a procedure and an object.



**Figure 3.2**  
The interface between a client module and a server module.

**Figure 3.3**

Highlighting a stretch of text to access the interface of a module.

---

DEFINITION StdLog;

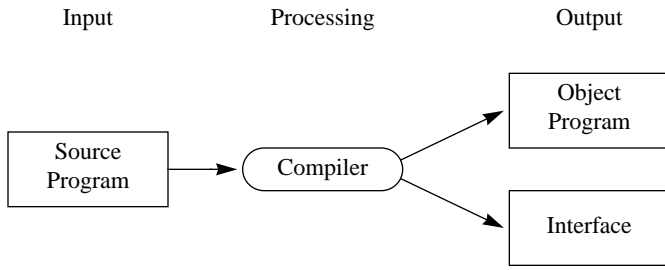
```
PROCEDURE Bool (x: BOOLEAN);  
PROCEDURE Char (ch: CHAR);  
PROCEDURE Int (i: LONGINT);  
PROCEDURE Ln;  
PROCEDURE Real (x: REAL);  
PROCEDURE String (IN str: ARRAY OF CHAR);
```

```
END StdLog.
```

---

**Figure 3.4**

The interface of the StdLog module. Not all exported items are shown.



**Figure 3.5**  
The compiler as a program.

---

```
MODULE Hw99Pr0380;  
(* Stan Warford *)  
(* June 12, 2002 *)
```

```
IMPORT StdLog;
```

```
PROCEDURE PrintAddress;
```

```
BEGIN
```

```
  StdLog.String("Mr. K. Kong"); StdLog.Ln;
```

```
  StdLog.String("Empire State Building"); StdLog.Ln;
```

```
  StdLog.String("350 Fifth Avenue"); StdLog.Ln;
```

```
  StdLog.String("New York, NY 10118-0110"); StdLog.Ln
```

```
END PrintAddress;
```

```
END Hw99Pr0380.
```

---

**Figure 3.6**

Sending output to the Log.

ABSTRACT	ELSIF	LIMITED	RECORD
ARRAY	EMPTY	LOOP	REPEAT
BEGIN	END	MOD	RETURN
BY	EXIT	MODULE	THEN
CASE	EXTENSIBLE	NIL	TO
CLOSE	FOR	OF	TYPE
CONST	IF	OR	UNTIL
DIV	IMPORT	OUT	VAR
DO	IN	POINTER	WHILE
ELSE	IS	PROCEDURE	WITH

*Component Pascal's reserved words*

When you try to execute a program, two types of errors are possible:

- Syntax error—The program does not compile.
- Logical error—The program compiles but produces incorrect results.



---

```
MODULE Hw99Pr0381;  
(* Stan Warford *)  
(* June 12, 2002 *)
```

```
IMPORT StdLog;
```

```
PROCEDURE PrintAddress*
```

```
BEGIN
```

```
  StdLog.String("Mr. K. Kong"); StdLog.Ln;
```

```
  StdLog.String("Empire State Building"); StdLog.Ln;
```

```
  StdLog.String("350 Fifth Avenue"); StdLog.Ln;
```

```
  StdLog.String("New York, NY 10118-0110"); StdLog.Ln
```

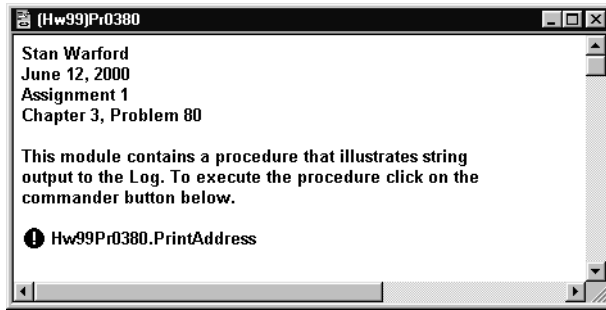
```
END PrintAddress;
```

```
END Hw99Pr0381.
```

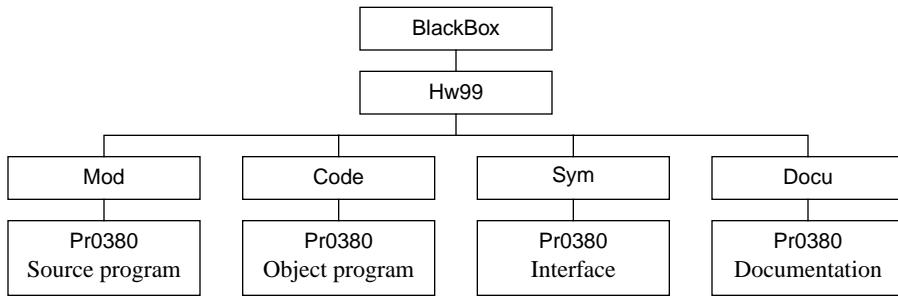
---

**Figure 3.7**

This procedure has a bug.

**Figure 3.8**

Documentation for module Hw99Pr0380, It is stored as file Pr0380 in the Docu folder.



**Figure 3.9**  
The files associated with a BlackBox program.

---

DEFINITION Hw99Pr0380;

    PROCEDURE PrintAddress;

END Hw99Pr0380.

---

**Figure 3.10**

The interface for the program in Figure 3.6.

---

```
MODULE Hw99Pr0382;
(* Stan Warford *)
(* June 18, 2002 *)

IMPORT StdLog;

PROCEDURE PrintAddress *;
BEGIN StdLog.String ("Mr. K. Kong"); StdLog.Ln;
StdLog.String
("Empire State Building"); StdLog.Ln
; StdLog.String ("350 Fifth Avenue"); StdLog.Ln;
StdLog.String
("New York, NY 10118-0110"); StdLog.Ln
END PrintAddress; END Hw99Pr0382.
```

---

**Figure 3.11**  
This module compiles  
without error.

---

```
MODULE Hw99Pr0383;
  IMPORT StdLog;

  PROCEDURE PrintPattern*;
  BEGIN
    StdLog.String("@"); StdLog.Ln;
    StdLog.String("@ @"); StdLog.Ln;
    StdLog.String("@ @ @"); StdLog.Ln;
    StdLog.String("@ @ @ @"); StdLog.Ln;
    StdLog.String("@"); StdLog.Ln;
    StdLog.String("@ @"); StdLog.Ln;
    StdLog.String("@ @ @"); StdLog.Ln;
    StdLog.String("@ @ @ @"); StdLog.Ln;
    StdLog.String("@"); StdLog.Ln;
    StdLog.String("@ @"); StdLog.Ln;
    StdLog.String("@ @ @"); StdLog.Ln;
    StdLog.String("@ @ @ @"); StdLog.Ln
  END PrintPattern;

END Hw99Pr0383.
```

---

**Figure 3.12**

This procedure prints three triangles to the Log.

---

```
MODULE Hw99Pr0384;
  IMPORT StdLog;

  PROCEDURE PrintTriangle;
  BEGIN
    StdLog.String("@"); StdLog.Ln;
    StdLog.String("@ @"); StdLog.Ln;
    StdLog.String("@ @ @"); StdLog.Ln;
    StdLog.String("@ @ @ @"); StdLog.Ln
  END PrintTriangle;

  PROCEDURE PrintPattern*;
  BEGIN
    PrintTriangle;
    PrintTriangle;
    PrintTriangle
  END PrintPattern;

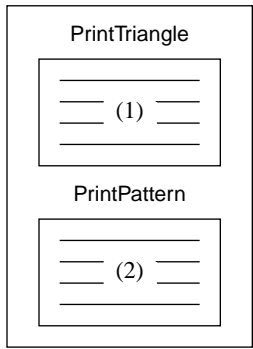
END Hw99Pr0384.
```

---

**Figure 3.13**

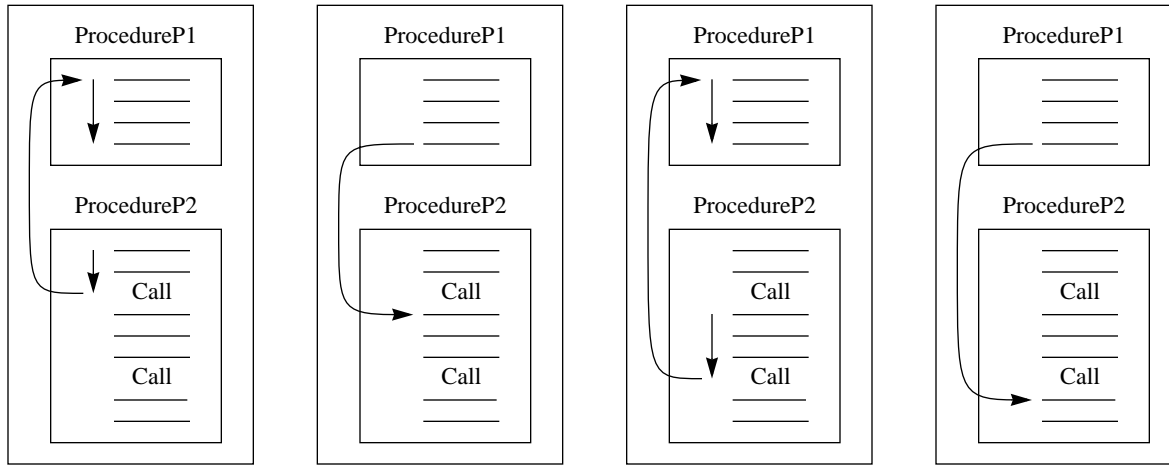
This procedure prints three triangles to the Log using a procedure.

Hw99Pr0384



**Figure 3.14**  
Procedures `PrintPattern` and `PrintTriangle` nested in module `Hw99Pr0384`.





(a) The first call transfers control to the procedure. The procedure executes.

(b) The procedure returns control to the statement following the first call.

(c) The second call transfers control to the procedure. The procedure executes again.

(d) The procedure returns control to the statement following the second call.

**Figure 3.15**  
The order of execution when a procedure has two procedure calls.

---

DEFINITION Hw99Pr0384;

    PROCEDURE PrintPattern;

END Hw99Pr0384.

---

**Figure 3.16**

The interface for the module  
in Listing 3.13.