

Q: The chapter says that a shared array should be declared as follows:

```
extern int a[];
```

Since arrays and pointers are closely related, would it be legal to write

```
extern int *a;
```

instead? [p. 356]

A: No. When used in expressions, arrays “decay” into pointers. (We’ve noticed this behavior when an array name is used as an argument in a function call.) In variable declarations, however, arrays and pointers are distinct types.

Q: Does it hurt if a source file includes headers that it doesn’t really need?

A: Not unless the header has a declaration or definition that conflicts with one in the source file. Otherwise, the worst that can happen is a minor increase in the time it takes to compile the source file.

Q: I needed to call a function in the file `foo.c`, so I included the matching header file, `foo.h`. My program compiled, but it won’t link. Why?

A: Compilation and linking are completely separate in C. Header files exist to provide information to the compiler, not the linker. If you want to call a function in `foo.c`, then you have to make sure that `foo.c` is compiled and that the linker is aware that it must search the object file for `foo.c` to find the function. Usually this means naming `foo.c` in the program’s makefile or project file.

Q: If my program calls a function in `<stdio.h>`, does that mean that all functions in `<stdio.h>` will be linked with the program?

A: No. Including `<stdio.h>` (or any other header) has no effect on linking. In any event, most linkers will link only functions that your program actually needs.

Q: Where can I get the make utility? [p. 367]

A: make is a standard UNIX utility. The GNU version, known as GNU Make, is included in most Linux distributions. It’s also available directly from the Free Software Foundation (www.gnu.org/software/make/).

Exercises

Section 15.1

1. Section 15.1 listed several advantages of dividing a program into multiple source files.
 - (a) Describe several other advantages.
 - (b) Describe some disadvantages.

Section 15.2

- W 2. Which of the following should *not* be put in a header file? Why not?
 - (a) Function prototypes
 - (b) Function definitions