

The New C Standard (Excerpted material)

An Economic and Cultural Commentary

Derek M. Jones

derek@knosof.co.uk

5.1.2.2.3 Program termination

main
return equivalent to

If the return type of the `main` function is a type compatible with `int`, a return from the initial call to the `main` function is equivalent to calling the `exit` function with the value returned by the `main` function as its argument;¹⁰⁾

180

Commentary

The call to `exit` will invoke any functions that have been registered with the `atexit` function. This behavior is not the same as a return from `main` simply returning control to whatever startup function called it, which in turn calls `exit`. After `main` returns, any objects defined in it with automatic storage will no longer have storage reserved for them, rendering any access (i.e., in the functions registered with `atexit`) to them as undefined behavior.

C90

Support for a return type of `main` other than `int` is new in C99.

C++

The C++ wording is essentially the same as C90.

Common Implementations

The value returned is often used as the termination status of the executed program in the host environment. Many host environments provide a mechanism for testing this status immediately after program termination.

Example

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  signed char *gp;
5
6  void clean_up(void)
7  {
8  /*
9   * The lifetime of the object pointed to by gp
10  * has terminated by the time we get here.
11  */
12  printf("Value=%c\n", *gp);
13  }
14
15  int main(void)
16  {
17  signed char lc;
18
19  if (atexit(clean_up) != 0)
20      printf("Ho hum\n");
21
22  gp=&lc;
23
24  return 0;
25  }
```

reaching the `}` that terminates the `main` function returns a value of 0.

181

Commentary

The standard finally having to bow to sloppy existing practices.

C90

This requirement is new in C99.

If the main function executes a return that specifies no value, the termination status returned to the host environment is undefined.

Common Implementations

Many implementations followed this C99 specification in C90, prior to it being explicitly specified in C99.

A translator can choose to special case functions defined with the name `main` or simply provide, where necessary, an implicit return of 0 for all functions returning type `int`.

- 182 If the return type is not compatible with `int`, the termination status returned to the host environment is unspecified.

main
termination sta-
tus unspecified

Commentary

The return type can only be incompatible with `int` if `main` has been defined to return a different, incompatible, type. If `main` has a return type of `void`, an implementation may choose not to return any value. In this case, the standard does not guarantee that a call to the `exit` library function will return the requested status to the host environment.

C90

Support `main` returning a type that is not compatible with `int` is new in C99.

C++

It shall have a return type of `int`, . . .

3.6.1p2

Like C90, C++ does not support `main` having any return type other than `int`.

- 183 **Forward references:** definition of terms (7.1.1), the `exit` function (7.20.4.3).

References