

Handling errors in C

Programming languages such as C language do not provide direct support for error handling but because they are system program languages, it provides the lowest level of return values. Most functions of C and functions in Unix return values 1 or null in any error case and set an `errno` error code for global variables and instructions for errors that occur during the function call.

Programming languages such as C language do not provide direct support for error handling but because they are system program languages, it provides the lowest level of return values. Most functions of C and functions in Unix return values 1 or null in any error case and set an **errno** error code for global variables and instructions for errors that occur during the function call. You can find many different error codes in the Header file named.

So a C programmer can check the return value and perform the correct action based on the return value. In fact, the programmer should set the value `errno` to 0 at the time of program initialization. A value of 0 indicates that there is no error in the program.

The function `perror ()` and `strerror ()` and inform `errno` error in C

The C program language provided with **`perror ()`** and **`strerror ()`** functions can be used to display **errno** error messages.

The **`perror ()`** function displays the string you pass to it, followed by a colon, a white space, and then the text that describes the current error value.

The **`strerror ()`** function returns the pointer to the text that represents the error value.

Try to simulate an error condition and try opening a non-existent file. Here I use both functions to show how to use, but you can use one or more ways to print your error value. The second important point to keep in mind is that you should use **`stderr`** to generate all errors.

```
#include #include /* header file de su dung cac ham va hang can thiet*/ #include
```

Compile and execute the above C program to see the results:

Error divided by number 0 in C

This is one of the very common errors during the split process, any programmer who does not check the condition of the number divided by zero may encounter this error during execution.

The code below fixes this error by checking the condition if the dividend is zero before dividing:

```
#include #include main () { int sochia = 15 ; int sobichia = 0 ; int thuoi
```

Compiling and executing the above C program will produce the following results:

```
Ban dang thuc hien phep chia cho so 0!!! Ket thuc chuong trinh ...
```

Exit status of program in C

In fact to exit the program with EXIT_SUCCESS value in case the program exits after a successful operation. Here EXIT_SUCCESS is a macro defined as value 0.

If you have error conditions in your program, you should exit with a return status of EXIT_FAILURE defined as a value of -1. Now write the above program as follows:

```
#include #include main () { int sochia = 36 ; int sobichia = 6 ; int thuoi
```

According to Tutorialspoint

Previous lesson: Pressing type in C

Next lesson: Recursive in C

You finished reading the article "**Handling errors in C**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.