

Running C Programs

Objectives:

Having read this section you should be able to:

1. Edit, link and run your C programs

This section is primarily aimed at the beginner who has no or little experience of using compiled languages. We cover the various stages of program development. The basic principles of this section will apply to whatever C compiler you choose to use, the stages are nearly always the same

The Edit-Compile-Link-Execute Process:

Developing a program in a compiled language such as C requires at least four steps:

1. editing (or writing) the program
2. compiling it
3. linking it
4. executing it

We will now cover each step separately.

Editing:

You write a computer program with words and symbols that are understandable to human beings. This is the edit part of the development cycle. You type the program directly into a window on the screen and save the resulting text as a separate file. This is often referred to as the source file (you can read it with the TYPE command in DOS or the cat command in unix).

The custom is that the text of a C program is stored in a file with the extension .c for C programming language

Compiling:

You cannot directly execute the source file. To run on any computer system, the source file must be translated into binary numbers understandable to the computer's Central Processing Unit (for example, the 80*87 microprocessor). This process produces an intermediate object file - with the extension .obj, the .obj stands for Object.

Linking:

The first question that comes to most people's minds is Why is linking necessary? The main reason is that many compiled languages come with library routines which can be added to your program. These routines are written by the manufacturer of the compiler to perform a variety of tasks, from input/output to complicated mathematical functions. In the case of C the standard input and output functions are contained in a library (stdio.h) so even the most basic program will require a library function. After linking the file extension is .exe which are executable files.

Executable files:

Thus the text editor produces .c source files, which go to the compiler, which produces .obj object files, which go to the linker, which produces .exe executable file. You can then run .exe files as you can other applications, simply by typing their names at the DOS prompt or run using windows menu.

Using Microsoft C:

Edit stage:

Type program in using one of the Microsoft Windows editing packages.

Compile and link:

Select Building from Make menu. Building option allows you to both compile and link in the same option.

Execute:

Use the Run menu and select Go option.

Errors:

First error highlighted. Use Next Error from Search menu for further errors if applicable.

If you get an error message, or you find that the program doesn't work when you finally run it (at least not in the way you anticipated) you will have to go back to the source file - the .c file - to make changes and go through the whole development process again!

Unix systems:

The University's central irix Service is a Silicon Graphics Inc. Challenge XL system which runs a Unix-like operating system called IRIX. The basic information to run a C program on this system is covered in document HT.SI.05 - How To... Run C Programs On The irix Service. Although this document refers to the IRIX operating system many of the command options will be common to all Unix systems.

On all Unix systems further help on the C compiler can be obtained from the on-line manual. Type

```
man cc
```

on your local Unix system for more information.

Please note that Unix is a case sensitive operating system and files named firstprog.c and FIRSTPROG.c are treated as two separate files on these system. By default the Unix system compiles and links a program in one step, as follows:

```
cc firstprog.c
```

This command creates an executable file called a.out that overwrites any existing file called a.out. Executable files on Unix are run by typing their name. In this case the program is run as follows:

```
a.out
```

To change the name of the executable file type:

```
cc -o firstprog.c
```

This produces an executable file called firstprog which is run as follows:

```
firstprog
```