

Many programs need to process values (arguments) that are specified on the command line when the program is invoked. For example, a program might take two integer values on its command-line and return the sum of those numbers:

```
c:\> AddEm 17 -89
```

```
17 + -89 = -72
```

But how can a C program access values specified on the command-line? Actually, very easily . . .

In a C program, main() is a function and can optionally take two arguments:

```
int argc        // the number of command-line arguments
char *argv[]    // an array of pointers to character
                 // strings storing the command-line
                 // arguments
```

The values of `argc` and `argv[]` are automatically stored by the OS when the program is run. The asterisk (*) in the second just indicates that `argv[]` is really an array of **pointers** to character values.

A simple example:

```
#include <iostream.h>
void main(int argc, char *argv[]) {

    char *pname, *firstarg, *secondarg;

    if (argc < 3) {
        cout << "There should be two command-line arguments."
              << endl;
    }
    else {
        pname      = argv[0];      // first command-line argument
        firstarg   = argv[1];      // and second
        secondarg  = argv[2];      // and third
        cout << "The command line was: ";
        cout << pname << " " << firstarg << " " << secondarg <<
        endl;
    }
}
```

When you declare a pointer to character values

```
char *pname;
```

you're essentially creating a pointer to a character string of unspecified length. You can then assign any character pointer to `pname`, regardless of the number of characters that are in that string.

You can even use `pname` just as you would use a string literal.

When the command-line arguments aren't characters then you must carry out a conversion when you access `argv[]`.

Fortunately, the standard C libraries include functions for just that purpose. For example:

- `atof()` converts a character string to a float
- `atoi()` converts a character string to an int

```
#include <iostream.h>
#include <stdlib.h>
void main(int argc, char *argv[]) {
    char *pname;
    int  firstarg, secondarg, sum;
    if (argc < 3) {
        cout << "There should be two command-line arguments."
              << endl;
    }
    else {
        pname      = argv[0];
        firstarg   = atoi(argv[1]);
        secondarg  = atoi(argv[2]);
        sum        = firstarg + secondarg;
        cout << firstarg << " + " << secondarg << " = "
              << sum << endl;
    }
}
```