C++ main() function is able to accept command line arguments as follows

```c++
int main(int argc, char* argv[]) {
    // ...
}
```

argc stores the count, i.e., the number of elements in the char* array. argv is the char* array that stores the individual arguments.

argv[0] is always the fullname (/path/program_name) of the program.

The following code prints the command line arguments passed to the program.

```c++
#include <iostream>

using namespace std;

int main(int argc, char* argv[]) {
    cout << "argc = " << argc << endl;
    for (int i=0; i<argc; ++i) {
        cout << "argv[" << i << "] = " << argv[i] << endl;
    }

    return 0;
}
```

You are not forced to use argc and argv as identifiers in main(). These are simply the established conventions.

An alternate way to declare argv is

```c++
int main(int argc, char** argv) {
    // ...
}
```

Command line only returns character array. It’s your responsibility to convert these into appropriate form. E.g., atoi(), atof() and atol can be used to a “valid” string to int, float and double.