interface. Look at how to use these choices with \texttt{while} loops.

\section*{Using Unadorned \texttt{cin} for Input}

If a program is going to use a loop to read text input from the keyboard, it has to have some way of knowing when to stop. How can it know when to stop? One way is to choose some special character, sometimes called a \textit{sentinel character}, to act as a stop sign. For example, \texttt{Listing 5.15} stops reading input when the program encounters a \# character. The program counts the number of characters it reads and it echoes them. That is, it redispalyes the characters that have been read. (Pressing a keyboard key doesn't automatically place a character on the screen; programs have to do that drudge work by echoing the input character. Typically, the operating system handles that task. In this case, both the operating system and the test program echo the input.) When finished, it reports the total number of characters processed. \texttt{Listing 5.15} shows the program.

\texttt{Listing 5.15 textin1.cpp}

\begin{verbatim}
// textin1.cpp -- reading chars with a while loop
#include <iostream>
using namespace std;
int main()
{
    char ch;
    int count = 0;    // use basic input

    cin >> ch;       // get a character
    while (ch != '#') // test the character
    {
        cout << ch;    // echo the character
        count++;       // count the character
        cin >> ch;     // get the next character
    }
    cout << "\n" << count << " characters read\n";
    return 0;
}
\end{verbatim}