The `cin` stream input operator is a class which has a number of functions associated with it for special input purposes. Two of these functions can be used for strings and characters.

`cin.getline` is a function which returns a full line of input (ending with a carriage return) from the input stream. The syntax is illustrated below. The function takes two arguments the first of which is a character buffer capable of holding the input line and the second is the maximum number of characters.

```cpp
#include<iostream>
#include<cstring>
#include<cstdio>
using namespace std;

main()
{
    char line[80];
    int i;
    cout << "Enter a line of text... " << endl;
    cin.getline(line, 80);
    while(strlen(line) > 0)
    {
        for(i=0;i<strlen(line);i++)
            if(line[i] != ' ')
                cout << line[i];
        cout << endl;
        cout << "Enter a line of text... " << endl;
        cin.getline(line, 80);
    }

    system("Pause");
}
```
cin.get is a function which inputs one character at a time. Its only argument is a char. The use of this function is illustrated below. Note that when an empty line is entered the function returns the ascii character for line feed (ascii code 10).

```cpp
//SGet.cpp
//This program illustrates the use of the getch function. It counts
//the number of e's in the input sequence. It ends on a line feed
//character.
#include<iostream>
#include<stdio>
#include<string>
using namespace std;
main()
{
    char ch;
    int eCount = 0;
    cin.get(ch);
    while(ch != (char)10)
    {
        if(ch == 'e' || ch == 'E')
            eCount++;
        cin.get(ch);
    }
    cout << "There were " << eCount << " e's entered." << endl;
    system("Pause");
}
```