More on Streams

When writing a function which processes an input or output, the stream parameters must always be **passed by reference**.

Here, for example, is a function that copies input to output.

```
void streamCopy(istream& in, ostream& out)
{
    char ch;
    while (in.get(ch)) { out.put(ch); }
}
```

We could rewrite ***filter*** by **calling this function**, like this:

```
int main()
{
    streamCopy(cin, cout);
}
```

Other I/O Functions

When reading individual characters, you'll sometimes find that you have **read more than you need**. There are several ways to solve this problem in C++.

1. **`in.unget()`** returns the last read character **to the input stream**.
2. **`in.putback(ch)`** allows you to put back a **different character**.
3. **`in.peek()`** looks at the next character in the stream, but doesn't remove it from the stream.

The C++ library guarantees that it you can push back push one character. You are not able to read several characters ahead and then push them all back. Fortunately, being able to push back one character is sufficient in the vast majority of cases.