

```
/*  
* Advanced programming  
* Alireza Akhavan Pour  
* Akhavan@AlirezaWeb.com  
* date: 1394/11/10  
*/
```

```
int main()  
{  
    cout<< "In the name of God";  
    Lecture_1();  
    return 0;  
}
```

```
string Lecture_1()  
{  
    cout << "1.Introduction" <<endl;  
    cout << "2.Review" <<endl;  
    cout << "3.string" <<endl;  
    return ":"  
}
```

این ترم میخوانیم ...

C++ □

- مفاهیم پایه
- شی گرای
- قالب
- ...

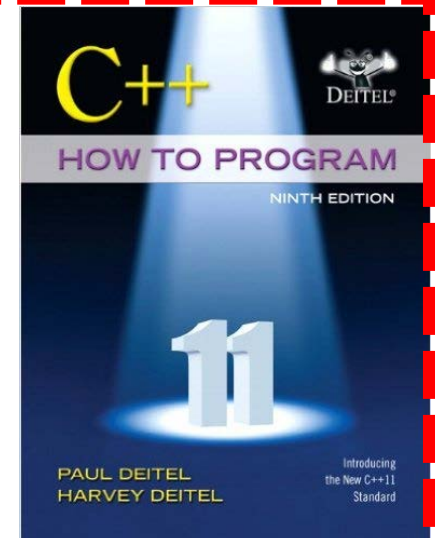
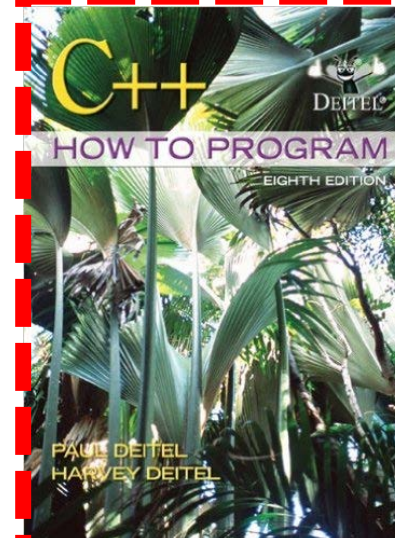
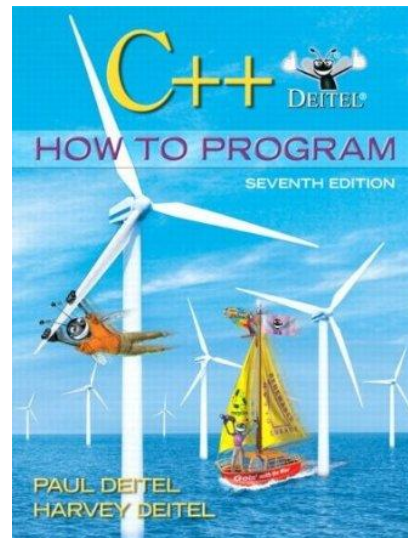
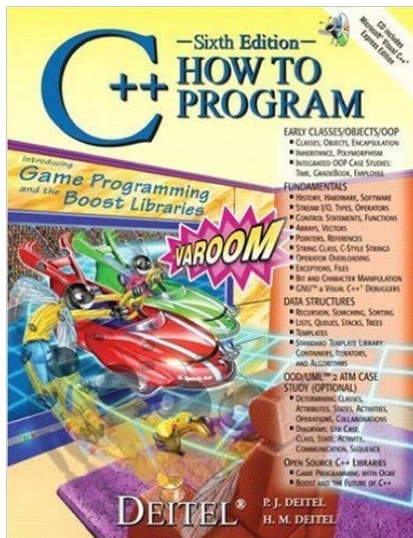
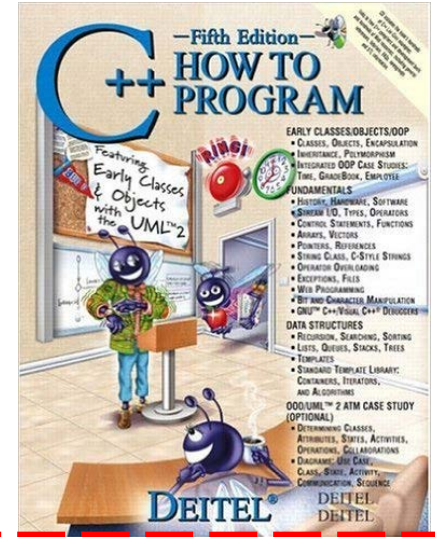
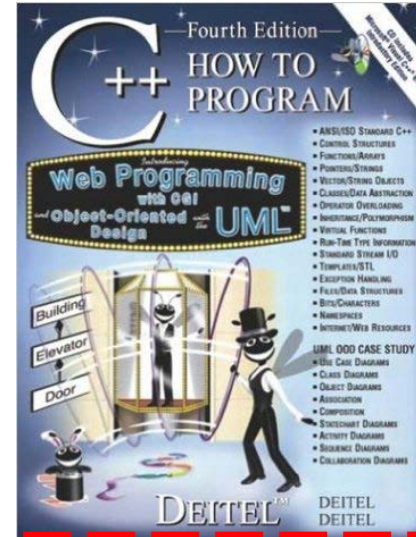
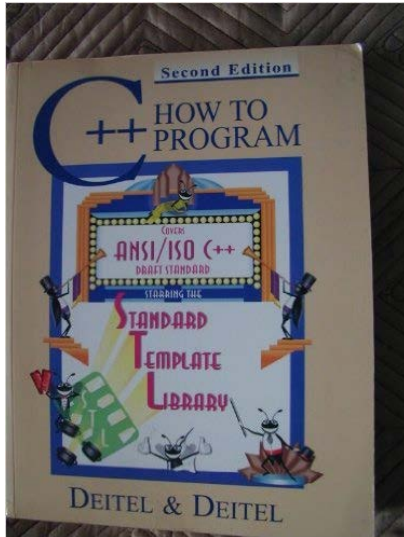
!? **C#** □

- ویندوز اپلیکیشن

چرا **C++** **!?**

C++ how to program Deitel & Deitel

منابع!؟



▲ First, make an `ifstream`:

342

```
#include <fstream>
std::ifstream infile("thefile.txt");
```

▼

The two standard methods are:



1. Assume that every line consists of two numbers and read token by token:

```
int a, b;
while (infile >> a >> b)
{
    // process pair (a,b)
}
```

2. Line-based parsing, using string streams:

```
#include <sstream>
#include <string>

std::string line;
while (std::getline(infile, line))
{
    std::istringstream iss(line);
    int a, b;
    if (!(iss >> a >> b)) { break; } // error

    // process pair (a,b)
}
```

You shouldn't mix (1) and (2), since the token-based parsing doesn't gobble up newlines, so you may end up with spurious empty lines if you use `getline()` after token-based extraction got you to the end of a line already.

share improve this answer

edited Mar 28 '12 at 6:57



Michael Mrozek

61k ● 10 ● 109 ● 130

answered Oct 23 '11 at 20:34



Kerrek SB

254k ● 32 ● 448 ● 678

2 Will solution #1 work with commas as the token? – stackptr Oct 18 '14 at 14:18

2 the OP used a space to delimit the two integers. I wanted to know if while (infile >> a >> b) would work if the

ارزیابی

- کوئیزها: ۲ نمره
- تمارین: ۳ نمره
- پروژه: ۳ نمره
- امتحان میانترم: ۴ نمره
- امتحان پایان ترم: ۶ نمره
- فعالیت کلاسی: ۳ نمره



- تحقیقات داوطلبانه: ۲ نمره

IDE & compiler?!

Linux

- ✓ g++

- ✓ ...

Window

- ✓ Code blocks

- ✓ Visual studio

- ✓ ...


```
1 // my first program in C++
2 #include <iostream>
3 using namespace std;
4
5 int main ()
6 {
7     cout << "Hello World! ";
8     cout << "I'm a C++ program";
9 }
```

Hello World! I'm a C++ program

session1\hello_world

```
1 // i/o example
2
3 #include <iostream>
4 using namespace std;
5
6 int main ()
7 {
8     int i;
9     cout << "Please enter an integer value: ";
10    cin >> i;
11    cout << "The value you entered is " << i;
12    cout << " and its double is " << i*2 << ".\n";
13    return 0;
14 }
```

Please enter an integer value: 702
The value you entered is 702 and its double is 1404.

session1\2_io

نوع جدیدی به نام string

```
1 // my first string
2 #include <iostream>
3 #include <string>
4 using namespace std;
5
6 int main ()
7 {
8     string mystring;
9     mystring = "This is a string";
10    cout << mystring;
11    return 0;
12 }
```

This is a string

session1\3_string

هر یک از سه حال زیر را می‌توانیم برای تعریف یک رشته در C++ استفاده کنیم.

```
1 string mystring = "This is a string";
2 string mystring ("This is a string");
3 string mystring {"This is a string"};
```

آیا میتوان ورودی فایل exe برنامه را از فایل متنی لود کرد؟ خروجی برنامه را میتوان در فایل متنی ذخیره کرد؟



`endl` appends `'\n'` to the stream **and** calls `flush()` on the stream. So

22

```
cout << x << endl;
```



is equivalent to



```
cout << x << '\n';  
cout.flush();
```

A stream may use an internal buffer which gets actually streamed when the stream is flushed. In case of `cout` you may not notice the difference since it's somehow synchronized (**tied**) with `cin`, but for an arbitrary stream, such as file stream, you'll notice a difference in a multithreaded program, for example.

[Here's](#) an interesting discussion on why flushing may be necessary.

```
int main()
{
    string previous = " ";    // previous word; initialized to "not a word"
    string current;          // current word
    while (cin>>current) {   // read a stream of words
        if (previous == current) // check if the word is the same as last
            cout << "repeated word: " << current << "\n";
        previous = current;
    }
}
```

session1\4_repeated_word

- ✓ برنامه‌ی بالا چه میکند؟
- ✓ چگونه حلقه‌ی بالا به پایان میرسد؟