

COMSC-165 Lecture Topic 1

Console Programming Basics

Reference

Deitel, chapter 2

[Compiling on a Windows system with Visual Studio PDF](#)

[Compiling on a Windows system with Visual Studio video](#)

[Compiling on a Mac PDF](#)

[Compiling in Linux or UNIX PDF](#)

[Compiling online, using only a browser](#)

C++ Editors And Compilers

compilers variations

variable initialization

zero, or unpredictable?

`#include <cstdlib>`

`using std::atoi;`

`#include <string>` and `using std::string;`

ANSI Standard C++ (versions 99 and 11)

2-space indenting -- no TABs

Command Line Compiling

command-line compiling used in lecture

"lowest common denominator"

vendor-, system-, and compiler-independent

on PCs in ATC and L bldgs (*USB drive recommended*)

last-executed `cout` ends with `endl`

The main Function

`int main()`

```
{
}
```

not void, for compatibility across compilers

does not work in g++

`int main()` **does not need to return anything!**

Console Input Pause

```
cout << "Press ENTER to continue...";
cin.get();
```

do NOT use Microsoft-specific

Working With Numbers

using the "stream extraction operator", `>>`

e.g., `int x;` OR `double y;`

Deitel: `std::cin >> x >> y;`

Comsc-165: `char buf[100]; cin >> buf; x = atoi(buf);`

`cin >> buf; y = atof(buf);`

with proper `#includes`

note: `buf` is reuseable

allows `Q` as sentinel instead of `-999`

`if (buf[0] == 'Q' || buf[0] == 'q') break;`

"calculator" arithmetic: `+` `-` `*` `/` `%`, *binary* arithmetic operators

about "promotion" in mixed arithmetic

truncation in integer division

order of operation, and parentheses

`a + b / c + d` does *not*

equal `(a + b) / (c + d)`

Working With Multiple Inputs

read numbers into `ints` like this:

```
#include <iostream>
using std::cin;
using std::cout;
```

```
#include <cstdlib>
```

```
int main()
{
    char buf[100];
    int age;
    cout << "What is your age? ";
    cin >> buf; age = atoi(buf);
    cin.ignore(1000, 10); // clears keyboard buffer
}
```

read text into a C string like this:

```
#include <iostream>
using std::cin;
using std::cout;
```

```
int main()
{
    char name[100];
```

```
system("pause");
```

Basic Formatting Of double Values

formatting numeric output

```
cout.setf(ios::fixed|ios::showpoint);
cout << setprecision(2) ...
```

C and C++ Header Files

A good ref: msdn.microsoft.com

C vs. pre-standard C++ vs. **modern C++**

<cstring> instead of <string.h>

<cstdlib> instead of <stdlib.h>

<ctime> instead of <time.h>

<cctype> instead of <ctype.h>

the std namespace

```
#include <iostream>
```

```
using std::cout;
```

```
using std::endl;
```

```
using std::ios;
```

```
#include <iomanip>
```

```
using std::setw;
```

```
using std::setprecision;
```

```
#include <cmath>
```

```
#include <cstdlib>
```

Console Output Of A Line Of Text

using the "stream insertion operator", <<
and the `iostream` C++ library

Deitel: `std::cout << "Hello\n";`

Comsc-165: `cout << "Hello\n";`

with `using std::cout;`

under `#include <iostream>`

using namespace std; not allowed in
Comsc-165

the std namespace in C++

Working With Line Breaks

\n VS endl for "line break"

either is okay

the "escape" character in C and C++, \

using multiple or no line breaks...

```
cout << "What is your name? ";
cin.getline(name, 100); // clears keyboard buffer
}
```

note: no usings or includes needed for "ignore"

Decision Making With Numbers

simple "branching": if x is equals to zero, do this...

binary equality operators: == and !=

binary relational operators: <, <=, >, and >=

```
if (x == 0)
    cout << "x's value equals zero\n";
```

Working With Absolute Values

use the `cstdlib` function

`abs(x)` for int x

use the `cmath` function

`fabs(x)` for float x or double x

Storing a DOUBLE in an INT

avoid: `int x = fabs(...);`

prefer: `int x = (int)fabs(...);`

Text File I/O

`ifstream fin;` and `ofstream fout;`

`.open({a C string});`

`fin >> and fin.getline({a C string}, {C string
size});`

`fout << ...`

`.close();`

where's my "working folder"?

command line: same as CPP location

IDE: write a test CPP to create an output file, look
for it

Using Visual C++ 2010 For Win32 Console Applications

Application type:	Add common header files for:
<input type="radio"/> Windows application	<input type="checkbox"/> ATL
<input checked="" type="radio"/> Console application	<input type="checkbox"/> MFC
<input type="radio"/> DLL	
<input type="radio"/> Static library	
Additional options:	
<input checked="" type="checkbox"/> Empty project	
<input type="checkbox"/> Export symbols	
<input type="checkbox"/> Precompiled header	
