Programming Abstractions

CS106B

Cynthia Lee
Today’s Topics

Introducing C++

- Hamilton example
  - In QT Creator (the IDE for our class)
  - Function prototypes
  - C++ strings and streams
  - `<iostream>` and `cin/cout`
  - "simpio.h" and `getInteger()`

- TODO for Friday:
  - Install QT Creator on your laptop
    - (note: if you will use library/lab/dorm computers, it’s already there for you!)
  - Attend the QT Creator help hours if needed: Thurs 7:30-9:30PM in Tressider Food Court
  - (optional) Try coding “Hello, world!” by logging in to codestepbystep.com > Practice > C++ > “Hello, World!”
C++ variables and types (1.5-1.8)

C++

```cpp
int x = 42 + 7 * -5;
double pi = 3.14159;
char letter = 'Q';
bool done = true;
```

JavaScript

```javascript
var x = 42 + 7 * -5;
var pi = 3.14159;
var letter = 'Q';
var done = true;
```

Python

```python
x = 42 + 7 * -5
pi = 3.14159
letter = 'Q'
done = True
```
First C++ program (1.1)

/*
 * hello.cpp
 * This program prints a welcome message
 * to the user.
 */
#include <iostream>
using namespace std;

int main() {
    cout << "Hello, world!" << endl;
    return 0;
}
More C++ syntax examples (1.5-1.8)

```cpp
for (int i = 0; i < 10; i++) {
    if (i % 2 == 0) {
        x += i;
    }
    /* two comment styles */
}

while (letter != 'Q' && !done) {
    x = x / 2; if (x == 42) { return 0; } // works but bad style
}

binky(pi, 17); // function call
```
# C++ math functions (2.1)

#include <cmath>

<table>
<thead>
<tr>
<th>Function name</th>
<th>Description (returns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs(value)</td>
<td>absolute value</td>
</tr>
<tr>
<td>ceil(value)</td>
<td>rounds up</td>
</tr>
<tr>
<td>floor(value)</td>
<td>rounds down</td>
</tr>
<tr>
<td>log10(value)</td>
<td>logarithm, base 10</td>
</tr>
<tr>
<td>max(value1, value2)</td>
<td>larger of two values</td>
</tr>
<tr>
<td>min(value1, value2)</td>
<td>smaller of two values</td>
</tr>
<tr>
<td>pow(base, exp)</td>
<td>base to the exp power</td>
</tr>
<tr>
<td>round(value)</td>
<td>nearest whole number</td>
</tr>
<tr>
<td>sqrt(value)</td>
<td>square root</td>
</tr>
<tr>
<td>sin(value)</td>
<td>sine/cosine/tangent of an angle in radians</td>
</tr>
<tr>
<td>cos(value)</td>
<td></td>
</tr>
<tr>
<td>tan(value)</td>
<td></td>
</tr>
</tbody>
</table>

- see Stanford "gmath.h" library for additional math functionality
Some C++ logistical details (2.2)

```cpp
#include <libraryname>  // standard C++ library
#include "libraryname.h"  // local project library
```

- Attaches a library for use in your program
- Note the differences (common bugs):
  - `<>` vs " "
  - `.h` vs no `.h`

```cpp
using namespace name;
```

- *Mostly, just don’t worry about what this actually does/means! Copy & paste the std line below to the top of your programs.*
- Brings a group of features into global scope so your program can directly refer to them
- Many C++ standard library features are in namespace std:
  - using namespace std;
Printing output in C++

cout << expression << expression ...

endl
  ▪ A variable that means "end of line"
  ▪ Same as "\n", but more compatible with all operating systems

    // good style
    cout << "You are " << age << " years old!" << endl;

    // bad style
    cout << "You are " << age << " years old!\n";
C++ Basics: Making functions

DEMONSTRATION IN QT CREATORS
A simple C++ program (Error)

#include <iostream>
#include "console.h"
using namespace std;

int main()
{
    int crowdEnthusiasmLevel = 5;
    generateLyrics(crowdEnthusiasmLevel);
    return 0;
}

void generateLyrics(int daAmount)
{
    for (int i = 0; i < daAmount; i++)
    {
        cout << "Da ";
    }
    cout << endl;
}
A simple C++ program (Fixed v.1)

kinggeorge.cpp

#include <iostream>
#include "console.h"
using namespace std;

void generateLyrics(int daAmount) {
    for (int i = 0; i < daAmount; i++) {
        cout << "Da ";
    }
    cout << endl;
}

int main() {
    int crowdEnthusiasmLevel = 5;
    generateLyrics(crowdEnthusiasmLevel);
    return 0;
}
A simple C++ program
(Fixed v.2)

#include <iostream>
#include "console.h"
using namespace std;

void generateLyrics(int daAmount);

int main() {
    int crowdEnthusiasmLevel = 5;
    generateLyrics(crowdEnthusiasmLevel);
    return 0;
}

void generateLyrics(int daAmount) {
    for (int i = 0; i < daAmount; i++) {
        cout << "Da ";
    }
    cout << endl;
}
#include "simpio.h"

- string fullName = getline("Student name? ");
- int age = getInteger("How old are you? ");
- double gpa = getReal("What's your GPA so far? ");
- if (getYesOrNo("Destroy the universe?")) { ... }
Hamilton Code Demo: What essential skills did we just see?

- You must use function prototypes for your helper functions (if you want to keep `main` at the top, which is good style)
- You can write input/output with:
  - `cout` (```<iostream>```
  - `cout` uses the `<<` operator
    - Remember: the arrows point in the way the data is “flowing”
    - These aren’t like HTML tags `<b></b>` or C++ parentheses () or curly braces {} in that they don’t need to “match”
- Good style: `static const int` to make int constants
  - (in demo, not previous slides)
  - No “magic numbers”!
  - Works for other types too (`static const double`)
Streams in C++

Extending the Hamilton Example

iostream (C++ Standard)
simpio (Stanford)
Extended Hamilton Code Demo:
What essential skills did we just see?

- You can read input with:
  - `cin(<iostream>)`
  - `getInteger()`, `getLine()`, etc print a message before waiting for input ("simpio.h")

- `cin` and `cout` use the `>>` and `<<` operators, respectively
  - Remember: the arrows point in the way the data is “flowing”
  - These aren’t like HTML tags `<b></b>` or C++ parentheses `()` or curly braces `{}` in that they don’t need to “match”
Conclusion: Today’s Topics

Introducing C++

- Hamilton example
  - In QT Creator (the IDE for our class)
  - Function prototypes
  - C++ strings and streams
  - `<iostream>` and cin/cout
  - "simpio.h" and getInteger()

- TODO for Friday:
  - Install QT Creator on your laptop
    - (note: if you will use library/lab/dorm computers, it’s already there for you!)
  - Attend the QT Creator help hours if needed: Thurs 7:30-9:30PM in Tressider Food Court
  - (optional) Try coding “Hello, world!” by logging in to codestepbystep.com > Practice > C++ > “Hello, World!”