The Software Development Process

Jill Seaman
CS 1428
Spring 2019

The Software Development Process

Software is usually developed in a process involving the following stages:

- Analysis and specification
- Design
- Implementation
- Testing and debugging
- Maintenance

Analysis and Specification

- **Analyze the Problem**: Figure out exactly the problem to be solved. Try to understand it as much as possible.
- **Determine Specifications**: Describe exactly what your program will do.
  - Don't worry about *how* the program will work, but *what* it will do.
  - This includes describing the inputs, outputs, and how they relate to one another.

Design

Create a Design:

- Formulate the overall structure of the program.
- This is where the *how* of the program gets worked out.
- You choose or develop your own algorithm that meets the specifications.

Implementation

Implement the Design:

- Translate the design into a computer language.
- AKA "coding"
- In this course we will use C++.

Testing and Debugging

Test/Debug the Program:

- Try out your program to see if it worked.
- If there are any errors (bugs), they need to be located and fixed. This process is called *debugging*.
- Your goal is to find errors, so try everything that might "break" your program!

Maintenance

Maintain the Program:

- Continue developing the program in response to the needs of your users.
- In the real world, most programs are never completely finished – they evolve over time.

Example Program: Temperature Converter

- Analysis – the temperature is given in Celsius, user wants it expressed in degrees Fahrenheit.
- Specification:
  - Input – temperature in Celsius
  - Output – temperature in Fahrenheit
  - Output = 9/5(input) + 32

Example Program: Temperature Converter

- Design:
  - Input, Process, Output (IPO)
  - Prompt the user for input (Celsius temperature)
  - Process it to convert it to Fahrenheit using F = 9/5(C) + 32
  - Output the result by displaying it on the screen

Example Program: Temperature Converter

- Before we start coding, let's write a rough draft of the program in *pseudocode*
- **Pseudocode** is precise English that describes what a program does, step by step.
- Using pseudocode, we can concentrate on the algorithm rather than the programming language.

Example Program: Temperature Converter

- Pseudocode:
  - Input the temperature in degrees Celsius (call it celsius)
  - Calculate fahrenheit as (9/5)celsius+32
  - Output fahrenheit
- Now we need to convert this to C++!

Example Program: Temperature Converter

```
// convert.cpp
// A program to convert Celsius temps to Fahrenheit

#include <iostream>
using namespace std;

int main() {
    float celsius, fahrenheit;
    cout << "What is the Celsius temperature? ";
    cin >> celsius;
    fahrenheit = (9/5) * celsius + 32;
    cout << "The temperature is " << fahrenheit
         << " degrees Fahrenheit." << endl;
    return 0;
}
```

Example Program: Temperature Converter

- Once we write a program, we should test it!

```
jillseaman$ ./a.out
What is the Celsius temperature? 0
The temperature is 32 degrees Fahrenheit.

jillseaman$ ./a.out
What is the Celsius temperature? 100
The temperature is 132 degrees Fahrenheit.
```

- The first result is correct, but the second case should give 212 degrees Fahrenheit.
  What caused this error?