Foundation of Computer Science with C++

Dr Kafi Rahman, PhD
Email: kafi@truman.edu
Text: 605-270-7645
Truman State University

Your Instructor

- Instructor: Dr. Kafi Rahman
- Office: VH2242
- Email: kafi@truman.edu
- Office hours: Monday and Wednesday; 1:20pm to 6:20pm
  - Please email if you want to set an appointment at a different time

Motivation to Learn Programming

- Computer Engineers/programmers get to work to solve real world amazing problems!
- Attractive remuneration packages!
- Less stressful compared to several other high paying jobs
- Computer usage is expanding in most domains. Hence, it is super useful in many areas and it is smart idea to know programming.

Quiz of the Day (1 of 5)

- In this course, we are going to use which of the following compiler?
  - C++11 with clang (llvm)
  - C++11 with gcc
  - C++14 with code::blocks

Quiz of the Day (2 of 5)

- Which of the following is not an example of computer Hardware?
  - computer monitor
  - computer printer
  - keyboard
  - computer chair

Quiz of the Day (3 of 5)

- Which of the following is not a Characteristics of a Truman Student?
  - Ask questions and passionately seek knowledge
  - Welcome and value new and diverse perspectives
  - Take generously money, gold, and financial resources and spend them in computer games.
  - Live emotionally and physically healthy lives

Quiz of the Day (4 of 5)

- Which of the following is something you should always do as a Truman student
  - discuss the mechanics of editing, compiling, and running a program
  - discuss the details of any assignment, in person or electronically with other students
  - discuss the mechanics of using the codeBlocks IDE
  - use code from the instructor or the course book without reference

Quiz of the Day (5 of 5)

- Which of the following a Truman student should always avoid in order to ensure academic integrity
  - look at any portion of another student's code (to solve assignment) or writeup (during the exam)
  - show any portion of your code or writeup to another student
  - discuss the general strategy for completing an assignment
  - copy code from any sources whenever possible including from the instructor and the course book

Ways to Make Learning to Program Efficient

- Computer science is so much more than programming
  - But you cannot be a computer scientist without programming

a. Take time to learn!

- This class is largely about developing programming skills
- Program every day
  - type in, build/run, and experiment with code examples (course book)
  - change the programs and observe and learn from those changes
    - example program shows the summation of two numbers.
    - change that to calculate the summation of 10 numbers.

b. Utilize examples

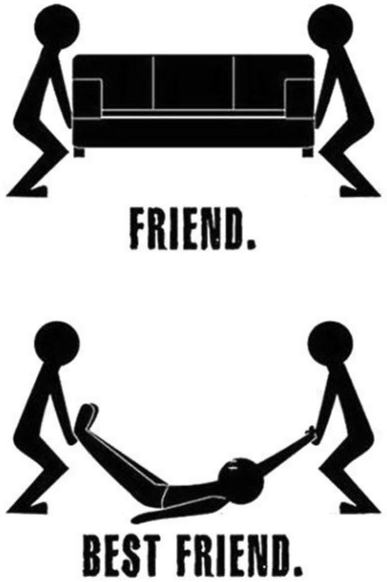
- Instead of writing all the code from scratch, it is good to look at a few examples and learn from them.
- In fact, majority of the learners prefer to look at examples instead of reading a manual.

c. Use online resources

- Do not confine yourself only to the textbook and the lecture notes
  - You should consult other references (including online sites) to improve your understanding of the material.
  - Tutorial points C++
- You should try to solve and understand the problems given in the course book.
- Understand your mistakes of last homework and assignments so that you can avoid them later

d. Learn with a friend

- Research says – The easiest way to get a good grades in this course would be to make friendship with students from this class!
- Learning in a group setting is preferred, if not, try at least to learn with a friend.
- Discuss ideas and help each other when you get stuck.
  - Enables you to work on a team assignment
  - Self-paced learning alone is not for every one.



e. Implement the solution of problems that interests you

- It will boost your confidence level
- The experience can later be useful in solving similar problems.

- Lastly,
  - Take advantage of the office hours
  - Prepare and consult your questions with the instructor.
  - Find a tutor @Truman for free!

Algorithm — history

- Muhammad Ibn Musa al-Khwarizmi was a Persian scholar
  - lived in Baghdad (now Iraq) in the 800's
  - around 820 he wrote a "book" giving precise, unambiguous, mechanical, efficient, and correct instructions for adding and multiplying numbers, and for calculating square roots, using decimal numbers
- The title of his book included the word al-jabr, from which our word algebra comes
- The word algorithm is named after al-Khwarizmi

Algorithm — definition

- Definition
  - A set of precise, unambiguous, mechanical, efficient, and correct instructions for accomplishing a task.

Algorithm — operations

- Categories of operations used to construct algorithms
  - Sequential operations
    - Carry out a single well-defined task; when that task is finished, the algorithm moves on to the next operation
    - Examples:
      - Add 1 cup of butter to the mixture in the bowl
      - Subtract the amount of the check from the current account balance
      - Set the value of MAX to 1

Algorithm — operations (contd.)

- An algorithm is a list that looks like
  - STEP 1: Do something.
  - STEP 2: Do something.
  - STEP 3: Do something.
  - . .
  - . .
  - . .
  - STEP N: Stop. You are finished.

Algorithm — operations (contd.)

- Conditional operations
  - Ask a question and then select the next operation to be executed on the basis of the answer to that question
  - Examples
    - If the mixture is too dry, then add one-half cup of water to the bowl

Algorithm — operations (contd.)

- Conditional operations examples (continued):
  - If the amount of the check is less than or equal to the current account balance, then cash the check; otherwise, tell the person that the account is overdrawn
    - If x is not equal to 0, then set y equal to 1/x; otherwise, print an error message that says we cannot divide by 0

Algorithm — operations (contd.)

- Iterative operations
  - Tell us to go back and repeat the execution of a previous block of instructions
  - Examples
    - Repeat the previous two operations until the mixture has thickened
    - While there are still more checks to be processed, continue processing checks
    - Repeat steps 1, 2, and 3 until the value of sum is less to 100

Algorithm (contd.)

- If we can specify an algorithm to solve a problem then we can solve the problem by using any programming languages.

- Hence, often times, algorithm is platform and programming language independent.

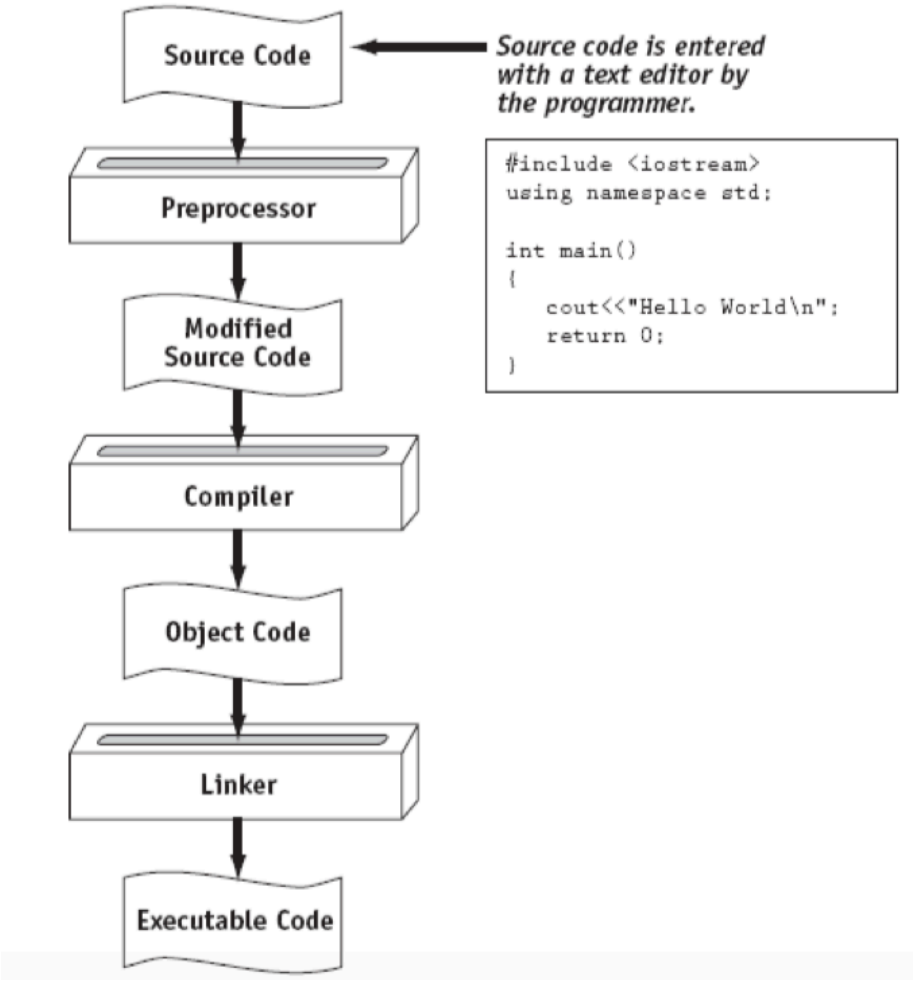
Algorithm — an example

- Lets design an algorithm for calculating area of a square in the following
  - 1. Display, "Enter the length of the square: "
  - 2. Get the value from the user and Store it in a memory location named LENGTH
  - 3. Display, "Enter the width of the square: "
  - 4. Get the value from the user and Store it in a memory location named WIDTH
  - 5. Now, calculate LENGTH x WIDTH and store the result in a memory location named AREA
  - 6. Display AREA along with an explanatory message

Source to Executable

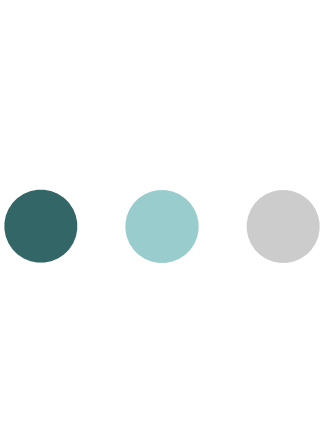
- C++ to Executable
  - 1. create C++ source code file using a text editor
  - 2. preprocessor: convert source file directives to source code program statements
  - 3. compiler: convert source code statements into machine instructions
  - 4. linker: connect hardware-specific code to machine instructions, producing a file of machine language statements
- Steps 2 – 4 are often performed by a single command or button click
- An error at any step will prevent the following steps

Source to Executable (contd.)



More Information

- http://www.codeblocks.org/
  - Information about the open source, cross platform, free C++ IDE
- https://en.wikipedia.org/wiki/Clang
  - Information about clang
- https://en.wikipedia.org/wiki/LLVM
  - Information bout llvm
- http://www.landofcode.com/programming-intro/source-code.php
  - Source code to programs



Questions?