SOME FUNDAMENTAL CONCEPTS OF PROGRAMMING

CSE 130: Introduction to Programming in C
Stony Brook University
“COMPUTER SCIENCE IS NO MORE ABOUT COMPUTERS THAN ASTRONOMY IS ABOUT TELESCOPES.”
— E. W. DIJKSTRA
THE NATURE OF COMPUTATION

➤ *Computation* — the solution of a complex problem by repeated systematic execution of a series of simple operations

➤ The problem must be defined exactly and unambiguously

➤ Computer programming is simply one way to automate (or mechanize) this process
HOW DO WE DESCRIBE A COMPUTATION IN SUFFICIENT DETAIL THAT THE STEPS CAN BE CARRIED OUT BY A MACHINE?
CHARACTERISTICS OF AN ALGORITHM

➤ A precise statement of the starting conditions
➤ A specification of the final state (a termination condition)
➤ A detailed description of the (simple) individual steps that will help move the algorithm forward toward the final state
  ➤ These steps are symbol manipulations
ALGORITHM EXAMPLES

➤ Grandma’s recipe for chocolate chip cookies
➤ Instructions for assembling a piece of furniture
➤ Driving directions
➤ Putting together a class schedule
➤ Euclid’s process for finding the Greatest Common Divisor of two numbers
EXAMPLE: COOKIE RECIPE

2 cups butter
4 cups flour
2 tsp. baking soda
2 cups granulated sugar
2 cups brown sugar
5 cups blended oatmeal (measure oatmeal and blend in blender to a fine powder)
24 oz. chocolate chips
1 tsp. salt
1 8 oz. Hershey bar (grated)
4 eggs
2 tsp. baking powder
3 cups chopped nuts (your choice)
2 tsp. vanilla

Source: http://urbanlegends.about.com/od/foodrink/a/cookie_recipe.htm
➤ Cream the butter and both sugars.
➤ Add eggs and vanilla; mix together with flour, oatmeal, salt, baking powder, and soda.
➤ Add chocolate chips, Hershey bar and nuts.
➤ Roll into balls and place two inches apart on a cookie sheet.
➤ Bake for 10 minutes at 375 degrees.
Once we have an algorithm, we need to express it in a form that the computer can understand.

Computers are designed to understand a specific set of instructions (operations).
MACHINE LANGUAGE

➤ Set of instructions designed into the CPU
  ➤ A CPU is basically a (very) complex system of logic gates (transistors and semiconductors)
  ➤ Internally, each instruction is represented as a sequence of bits (1s and 0s)

➤ Here's an example of a simple machine language program:

```
1100 0000 0000 0000 0100 0001
0111 0000 0000 0000 0010 0011
1110 0001 0000 0000 0000 0101
```
THE NEED FOR TRANSLATION

- Computers (CPUs) only speak binary (1s and 0s)
- People don’t speak binary well; we prefer higher-level languages like C
- High(er)-level languages are much more human-friendly
  - A single high-level instruction often translates to a sequence of multiple machine instructions
- A compiler is a special computer program that translates high-level languages into machine language (binary)
  - On the way to compilers, we developed a special instruction format known as assembly language
THE COMPILATION PROCESS

➤ A text editor is used to enter the C program into a file
  ➤ By convention, C source code files end with .c

➤ The compiler checks for errors and translates the C code into assembly language

➤ The assembler translates the assembly code into binary object code

➤ The linker joins together multiple pieces of object code into a single executable object