

# BASICs of C/C++ Programming

By Leland Robotics



# General Program Flow

- Title (mostly for other programming languages)
  - name of the program/function
- Declaration section
  - not very clear cut in C/C++
  - where you declare your variables
  - know what your working with
- Executable section
  - the calculations, processes, etc.
- For the purposes of programming on the team, we are not necessarily programming the whole program, rather, defining the processes needed in order to operate the robot.

# General Programming Practices

- Identify what you are going to need and what the purpose of the program is.
- Map out how your general program is going to work.
  - Flowchart (diagrams flow of program)
  - Structure Plan (a little more general on flow, has basic calculations)
- Figure out what you're going to be using.
  - Functions
    - Function Libraries/Declaring a new function
    - Constructions (Classes)
  - Calculations

# If You EVER Get Stuck

- Write out what you are thinking to solve the problem
- DIAGRAMS!
- Ask others for input.
  - It is a group effort.

# Flowchart vs. Structure Plan

## Flowchart

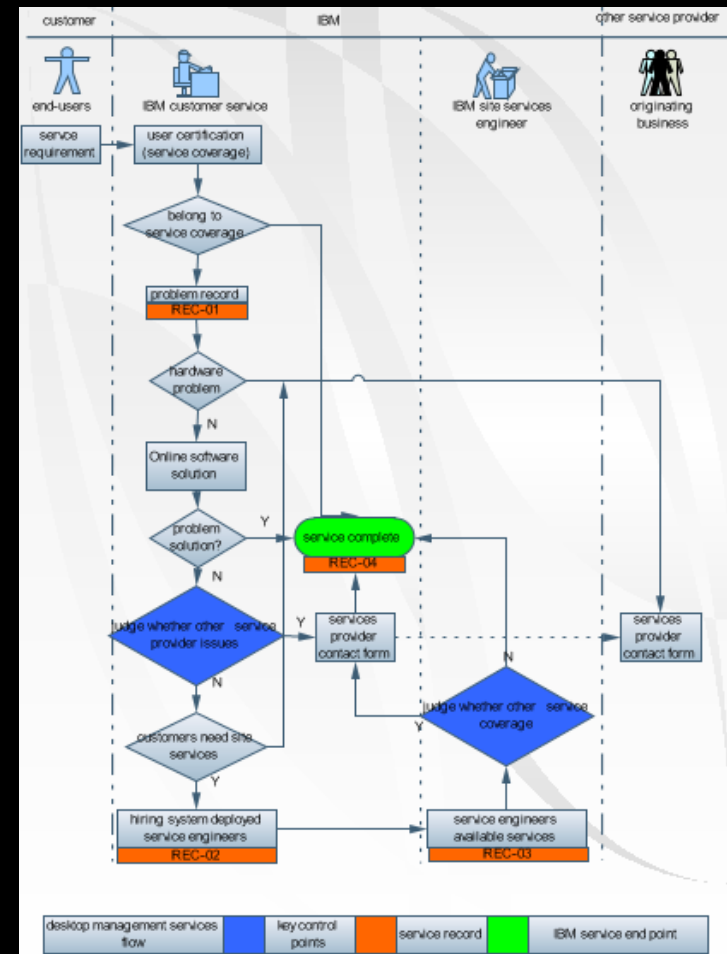
- Gives a simple visual representation of how the program should run.
  - Uses special figures to denote certain phases

## Structure Plan

- Bullet point plan
  - Almost a summary of what your program is going to look like.

# Flowchart (cont.)

- Shows the flow of the program
- As your programs get larger, your flow chart becomes more confusing.
- One drawback is that you don't lay down what your going to do in each step.
  - You only get a general idea



# Structure Plan (cont.)

- Summarizes your program into sections
- Does not clearly show the flow of the program
- Shows everything you might need for the program.

## Program (insert name)

- Declaration
  - A & C are integers
  - B is a string
- Calculation
  - $C=A$
  - get user input for B
- Display
  - Show C, A
  - Show B

# Variables

- Place holders you want to put stuff in
  - very much like variables in math
- Stores information
  - examples
    - `x="a"`
      - x now holds the value of "a"
      - display x results in a being displayed
    - `x=1`
      - x now holds value of 1
      - display x results in 1 being displayed



# Basic Data Types of variables in C/C++

- integer
  - short int (short) (memory allocation: 2 bytes)
    - integer able to hold around 5 digits
  - int (memory allocation: 4 bytes)
    - integer able to hold around 10 digits
  - long int (long)
    - integer able to hold around 10 digits (basically the same as int)
- char
  - holds one character (a, b, c, etc.) or a small integer
- bool
  - boolean values (i.e. true or false)
  - C technically doesn't have it innately, although in C++ it manifests itself as 1 or 0

# Basic Data Types of C/C++ (cont.)

- Float(floating point number)
  - can hold decimal digits
  - around 7 digits in total
- double (double precision float)
  - can hold decimal digits (around 15 digits)
- long double (long double precision float)
  - can hold decimal digits (same as double)
- string
  - can hold several characters

# Logic Operators

- In order of precedence
  - Not (only true when condition is false)
  - And (only true when both conditions are true)
  - Or (only true when one or more of the conditions is true)
  - Xor (exclusive or)(only true when either is true, and the other is false)
- And/Or/Xor have the same precedence

# Functions

FUNCTIONS

- Just think of them like math
  - the actual inner workings of a function are much more complicated
    - more on that later

# Control Structures

- "Artificially" controls the flow of a program
  - Generally two types
    - Conditionals
      - If structures
      - switch-case structures (conditions with multiple "inputs", different outputs for each)
    - Loops
      - Fall into two types as well
      - Conditional loops
      - While-loops
      - Set repetition loops
      - For-loops