Lecture 1

Introduction to Programming Concept

Dr M Kasim A Jalil

Faculty of Mechanical Engineering

UTM

What is PROGRAMMING LANGUAGE?

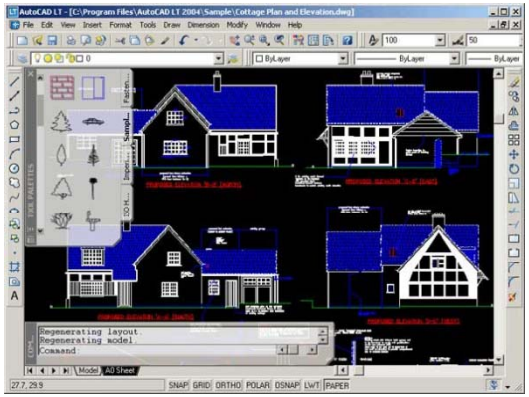
- code of reserved words and symbols used in computer programs, which give instructions to the computer on how to accomplish certain computing tasks
- an artificial language that can be used to control the behaviour of a machine (often a computer)
- syntax, grammar, and symbols or words used to give instructions to a computer

Programming vs Natural Language

- **Elements of language – vocabulary, rules/grammar, structure.**
- **Natural languages**
  - can be ambiguous and make small errors, and still expect their intent to be understood
  - human can guess the 'intended' meaning
- **Programming languages**
  - require a greater degree of precision and completeness
  - have syntactic and semantic rules used to define meaning
  - computers do exactly what they are told to do, and cannot understand the code the programmer "intended" to write
  - are used to facilitate communication about the task of organizing and manipulating information, and to express algorithms precisely

WHY programming?

- **A prominent purpose of programming languages is to provide instructions to a computer.**
- **Basis for developing applications**
  - Softwares – AutoCAD, Yahoo Messenger, Microsoft Words, MATLAB, etc.
  - Computer games – SIMS, Car racing, flight simulators, etc.
- **Developing skills & interests**
  - From USER (of softwares) to DEVELOPER (of applications)
  - MSC dream!



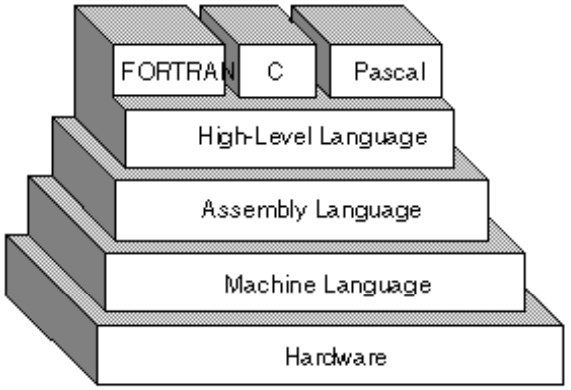
How many programming languages??

- **Definitely more than 50 languages!**
- **Many are short-lived and a few are still in-use:**
  - Some are application-specific
  - Lack of user and developer
- **To name a few:**
  - **Ada** - Comprehensive, Pascal-based language used by the Department of Defense.
  - **ALGOL** - International language for expressing algorithms.
  - **APL** - Used for statistics and mathematical matrices. Requires special keyboard symbols.
  - **BASIC -** Developed as a timesharing language in the 1960s. It has been widely used in microcomputer programming in the past, and various dialects of BASIC have been incorporated into many different applications. Microsoft's Visual Basic is widely used.

Programming languages....

- **C** - Developed in the 1980s at AT&T. Widely used to develop commercial applications. Unix is written in C. C++-Object-oriented version of C that is popular because it combines object-oriented capability with traditional C programming syntax.
- **COBOL** - Developed in the 1960s. Widely used for mini and mainframe programming.
- **dBASE** - Used to be widely used in business applications, but FoxPro (Microsoft's dBASE) has survived the longest.
- **FORTRAN** - Developed in 1954 by IBM, it was the first major scientific programming language and continues to be widely used. Some commercial applications have been developed in FORTRAN.
- **Java** - The programming language developed by Sun and repositioned for Web use. It is widely used on the server side, although client applications are increasingly used.
- **JavaScript** - A scripting language widely used on the Web. JavaScript is embedded into many HTML pages.
- **LISP** - Developed in 1960. Used for AI applications. Its syntax is very different than other languages.
- **Logo** - Developed in the 1960s, it was noted for its ease of use and "turtle graphics" drawing functions.
- **M** - Originally MUMPS (Massachusetts Utility MultiProgramming System), it includes its own database. It is widely used in medical applications.
- **Modula-2** - Enhanced version of Pascal introduced in 1979
- **Pascal** - Originally an academic language developed in the 1970s. Borland commercialized it with its Turbo Pascal.
- **Perl** - A scripting language widely used on the Web to write CGI scripts.
- **Prolog** - Developed in France in 1973. Used throughout Europe and Japan for AI applications.
- **Python** - A scripting language used for system utilities and Internet scripts. Developed in Amsterdam by Guido van Rossum.
- **Visual Basic** - Version of BASIC for Windows programming from Microsoft that has been widely used.
- **Web Languages** - Languages such as JavaScript, Jscript, Perl and CGI are used to automate Web pages as well as link them to other applications running in servers.

Talking to Computer



- **Machine Code/Language -** The lowest-level programming language understood by a computer's CPU, consisting entirely of numbers, hardly understood by human. Every CPU model has its own machine code.
- **Assembly Language -** An assembly language contains the same instructions as a machine language, but the instructions and variables have names instead of being just numbers.
- **High-Level Language** - A programming language (such as C, FORTRAN, or Pascal) that enables a programmer to write programs that are more or less independent of a particular type of computer. Such languages are considered high-level because they are closer to human languages and further from machine languages.

Low-level Languages (closer to hardware)

Assembler & Compiler

- **Assembler -** Software that translates assembly language into machine language.
- **Compiler -** Software that translates a program written in a high-level programming language (C/C++, COBOL, etc.) into machine language. A compiler usually generates assembly language first and then translates the assembly language into machine language. A utility known as a "linker" then combines all required machine language modules into an executable program that can run in the computer

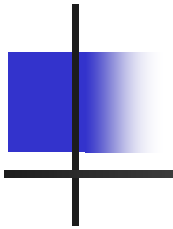
Source Code	Assembly Language	Machine Language
IF COUNT=10	Compare A to B	Compare 3477 2883
GOTO DONE	If equal go to C	If = go to 23883
ELSE	Go to D	Go to 23343
GOTO AGAIN		
ENDIF		

Actual Machine Code (simulated)

```
100101010010100010101000
101010100101010010010100
101001010100010100100100
```

Program vs Application

- A collection of instructions that tell the computer what to do. A program is generically known as "software" and the programs users work with, such as word processors and spreadsheets, are called "applications" or "application programs." Thus, the terms software, application, program and instruction are synonymous in the sense that they all tell the computer what to do.
- A program is written in a programming language, such as C or FORTRAN, and the statements and commands written by the programmer are converted into the computer's machine language by software called "assemblers," "compilers" and "interpreters." The program contains machine instructions, buffers, constants and counters.



Programming Process

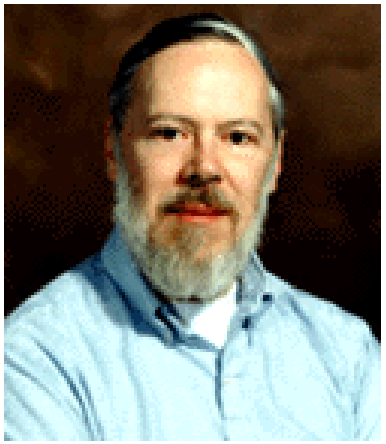
- **Just like learning any natural language:**
  - Learn the vocabulary
  - Learn how to construct the sentence (grammar & structure)
  - Have the expert check your skill
- **In any computer programming language**
  - Knowing the lingo - terms & rules (syntax)
  - Compiling - checking for errors (or called debugging)

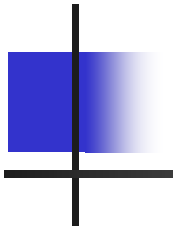
What do we normally program?

- **Repetitive & Duplicative process**
  - Sequence/series/multiple of complex operations
  - Repeated applications
  - Computationally expensive for human to do
- **Examples??**
  - Telephone directory – searching for a contact number
  - Balance of your prepaid phone
  - Bank transaction – withdrawal, balance enquiry, transfer, deposit, etc
  - Advanced Applications – matrix inverse, wave simulation.

C Language

- **The C programming language was designed by Dennis Ritchie at Bell Laboratories in the early 1970s**
- **Influenced by**
  - ALGOL 60 (1960),
  - CPL (Cambridge, 1963),
  - BCPL (Martin Richard, 1967),
  - B (Ken Thompson, 1970)





C Language

- **Traditionally used for systems programming, though this may be changing in favor of C++**
- **Traditional C:**
  - *The C Programming Language*, by Brian Kernighan and Dennis Ritchie, 2nd Edition, Prentice Hall
  - Referred to as *K&R*

Things we need to master….

- **'Programming thinking' – thinking like a programmer.**

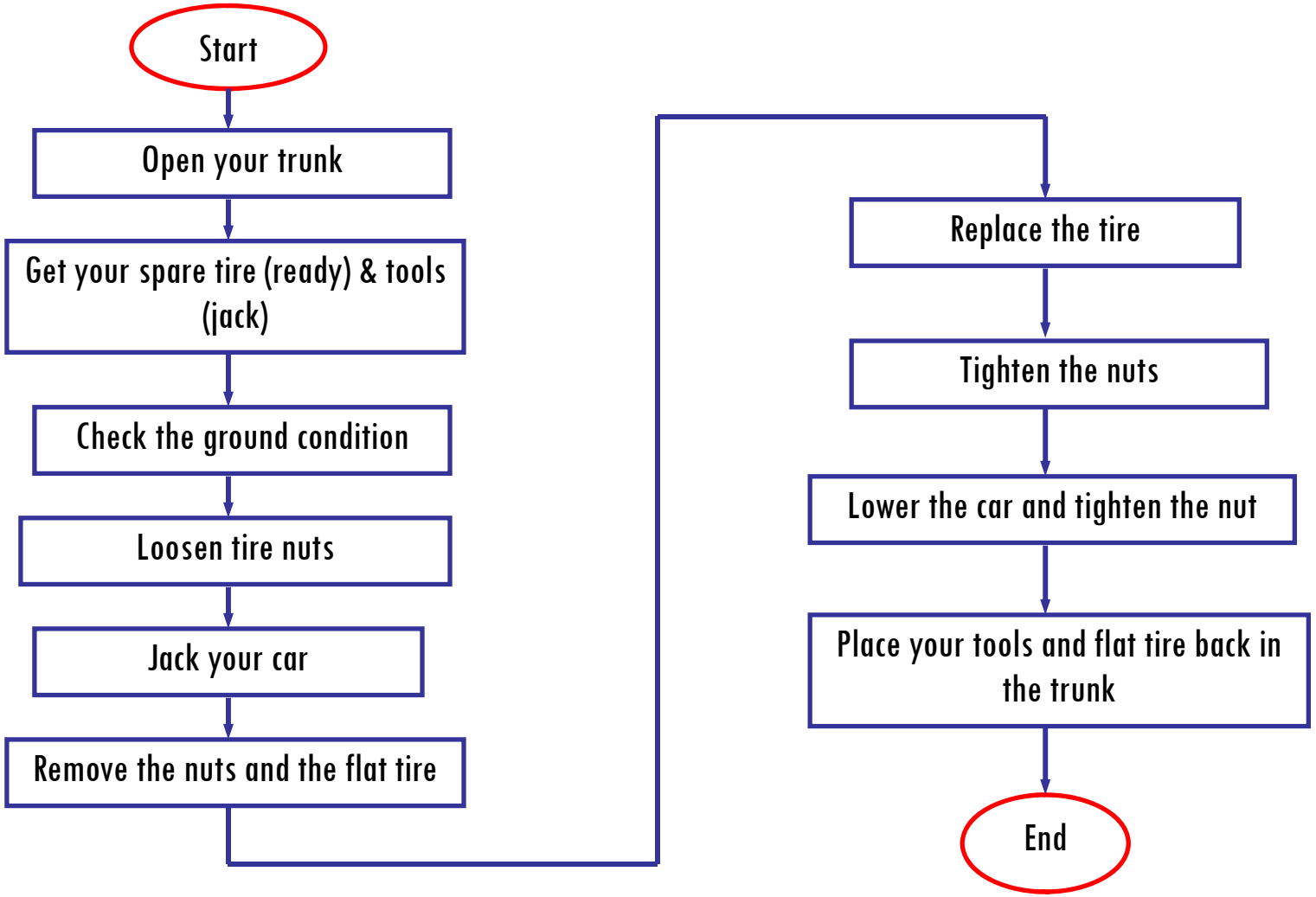
  You can't just simply think in Malay and speak in Mandarin. To be a master, you need to think and dream in Mandarin:

  - Program concept
  - Algorithm Development
  - Program Coding
  - Debugging

Example – Changing your flat tire

- Open your trunk
- Get your spare tire (ready) & tools (jack)
- Check the ground condition
- Loosen tire nuts
- Jack your car
- Remove the nuts and the flat tire
- Replace the tire
- Tighten the nuts
- Lower the car and tighten the nut
- Place your tools and flat tire back in the trunk
- End

Process Flow Chart



Making chicken curry (from suvir.com)

- **Tools –** knife, pan, stirrer, etc.
- **Ingredients**
  - 4-pound chicken, cut into 8 to 10 pieces and skinned
  - 3/4 teaspoon turmeric
  - 3/4 teaspoon cayenne pepper
  - Salt
  - 1 1/2 medium onions, roughly diced
  - 5 garlic cloves
  - 2 inches fresh ginger, peeled and cut in half crosswise
  - 3 tablespoons canola oil
  - 2 inches cinnamon stick
  - 12 green cardamom pods
  - 9 whole cloves
  - 9 black peppercorns
  - 2 large tomatoes, chopped
  - 2 tablespoons tomato paste
  - 1/4 cup yoghurt, whisked until smooth
  - 1 cup water
  - 1/2 cup chopped fresh cilantro
  - Juice of 1 lemon

Step-by-step instructions…

1. Mix the chicken, 1/2 teaspoon of the turmeric, 1/2 teaspoon of the cayenne and 1/4 teaspoon salt in a bowl and stir to coat the chicken with the spices. Let stand while you make the sauce.
2. Finely mince the onion, garlic and ginger in a food processor and set aside.
3. Mix 2 tablespoons of the oil, the cinnamon, cardamom, cloves and black peppercorns in a large casserole over medium-high heat. Cook & stir, until the cinnamon unfurls, for 1 to 2 minutes.
4. Add the minced onion mixture and 1 teaspoon salt and cook and stir, until the onion turns brownish around the edges, for 10 to 15 minutes.
5. Remove and discard the cinnamon and mix with the remaining 1/4 teaspoon turmeric and 1/4 teaspoon cayenne. Add the tomatoes and tomato paste, cook and stir for 5 minutes. Transfer to a food processor or blender and puree until smooth; set aside.
6. Heat the remaining 1 tablespoon oil in the same pan over medium-high heat. Add the chicken, cook and stir for 2 minutes.
7. Add the yoghurt, 1 tablespoon at a time and stir well after each addition. Cook and stir for 2 minutes to evaporate some of the moisture.
8. Add the pureed tomato mixture and bring to a boil. Pour in the water. Return to a boil, then reduce the heat and simmer, partially covered, until the chicken is cooked through, about 30 minutes. Stir and scrape the bottom of the pan every 5 to 8 minutes to keep the sauce from sticking. Then uncover and cook for 5 more minutes to reduce and thicken the sauce. Stir in the cilantro and lemon juice. Taste for salt and serve hot.

Flow Chart – please complete….

Mix the chicken, 1/2 teaspoon of the turmeric, 1/2 teaspoon of the cayenne and 1/4 teaspoon salt in a bowl and stir to coat the chicken with the spices. Let stand while you make the sauce.

Finely mince the onion, garlic and ginger in a food processor and set aside.

Mix 2 tablespoons of the oil, the cinnamon, cardamom, cloves and black peppercorns in a large casserole over medium-high heat. Cook and , until the cinnamon unfurls, for 1 to 2 minutes.

Add the minced onion mixture and 1 teaspoon salt and cook and stir, until the onion turns brownish around the edges, for 10 to 15 minutes.

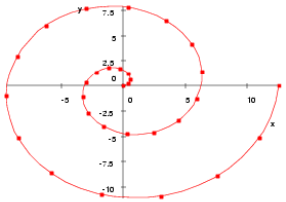
Remove and discard the cinnamon and mix with the remaining 1/4 teaspoon turmeric and 1/4 teaspoon cayenne. Add the tomatoes and tomato paste, cook and stir for 5 minutes. Transfer to a food processor or blender and puree until smooth; set aside.

Heat the remaining 1 tablespoon oil in the same pan over medium-high heat. Add the chicken, cook and stir for 2 minutes.

Add the yoghurt, 1 tablespoon at a time and stirring well after each addition. cook and stir for 2 minutes to evaporate some of the moisture.

Add the pureed tomato mixture and bring to a boil. Pour in the water. Return to a boil, then reduce the heat and simmer, partially covered, until the chicken is cooked through, about 30 minutes. Stir and scrape the bottom of the pan every 5 to 8 minutes to keep the sauce from sticking. Then uncover and cook for 5 more minutes to reduce and thicken the sauce. Stir in the cilantro and lemon juice. Taste for salt and serve hot.

Plotting a Curve



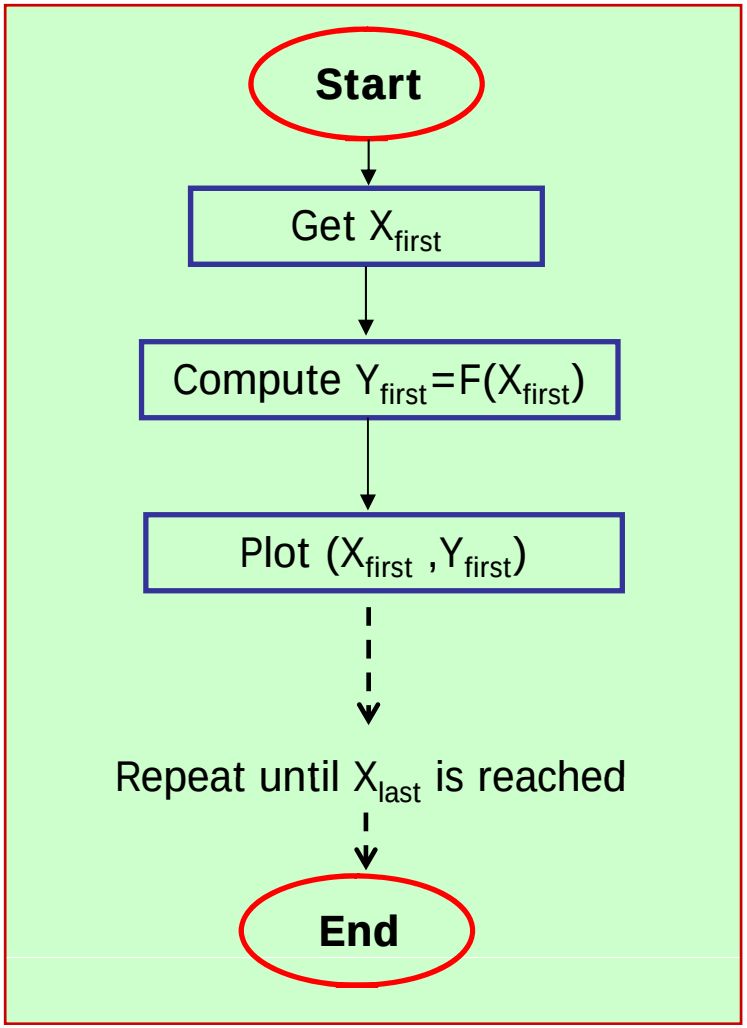
Manual

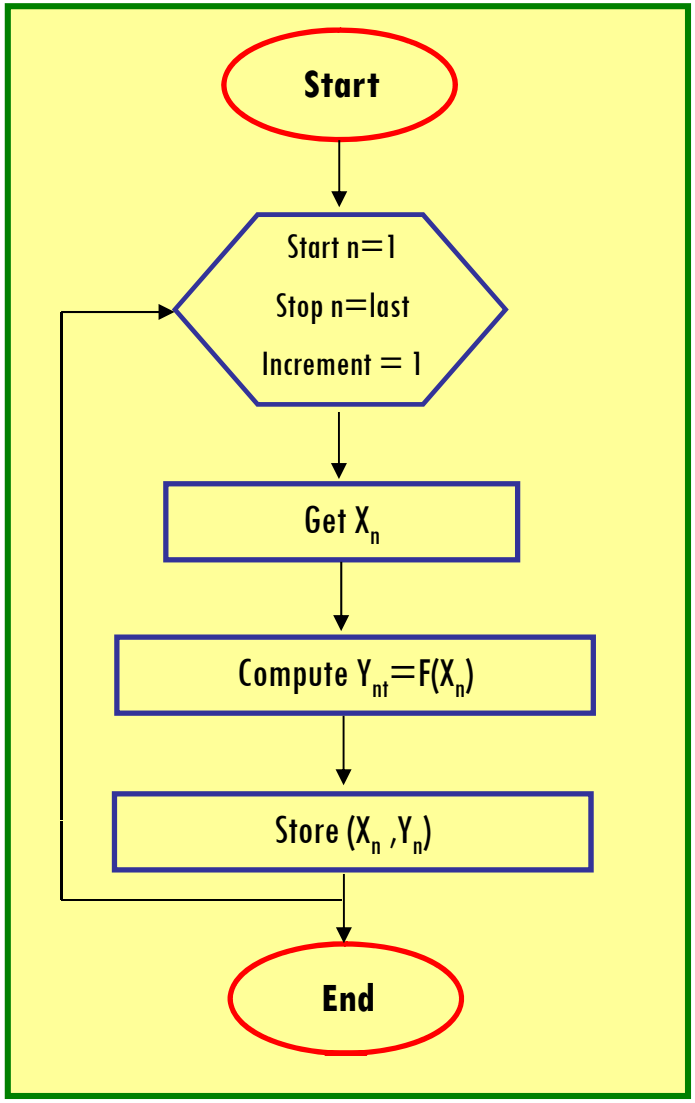
- **Draw the X & Y Axis**
- **Plot the Curve**
  - Get the governing equations: $y=mx+C$, $y=A\cos x$, $y=ax+bx^2$
  - Mark the X & Y axis
  - For each X value, calculate the Y and place the (X,Y) point on the graph paper
  - Connect the points

Process Flow

- Get the input value (X)
- Get the output value (Y)
- Repeat until maximum X is reached
- STOP

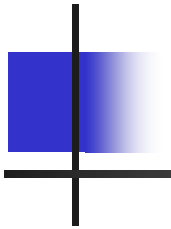
Flow Chart


Start
Get X_{first}
Compute $Y_{first}=F(X_{first})$
Plot (X_{first} ,Y_{first})
Repeat until X_{last} is reached
End


Start
Start n=1
Stop n=last
Increment = 1
Get X_n
Compute $Y_{nt}=F(X_n)$
Store (X_n ,Y_n)
End

Algorithm

- A formula or set of steps for solving a particular problem. To be an algorithm, a set of rules must be unambiguous and have a clear stopping point. Algorithms can be expressed in any language, from natural languages like English or French to programming languages like C.
- We use algorithms every day. For example, a recipe for baking a cake is an algorithm. Most programs, with the exception of some artificial intelligence applications, consist of algorithms. Inventing elegant algorithms -- algorithms that are simple and require the fewest steps possible -- is one of the principal challenges in programming.



Now Try!

- Revisit the Chicken Curry problem
- Think of a better way to represent the process in a systematic flow chart

An example of Algorithm

Recipe
CHOCOLATE CAKE

4 oz. chocolate	3 eggs
1 cup butter	1 tsp. vanilla
2 cups sugar	1 cup flour

Melt chocolate and butter. Stir sugar into melted chocolate. Stir in eggs and vanilla. Mix in flour. Spread mix in greased pan. Bake at 350_ for 40 minutes or until inserted fork comes out almost clean. Cool in pan before eating.

Program Code

Declare variables:
chocolate eggs mix
butter vanilla
sugar flour

```
mix = melted ((4*chocolate) + butter)
mix = stir (mix + (2*sugar))
mix = stir (mix + (3*eggs) + vanilla)
mix = mix + flour
spread (mix)
While not clean (fork)
bake (mix, 350)
```