Raúl Cátedra Martínez

Basic Programming concepts

Raúl Cátedra Martínez

What is an algorithm?

Is a finite sequence of well-defined, computer-implementable instructions, typically to solve a class of problems or to perform a computation.Algorithms are always unambiguous and are used as specifications for performing calculations, data processing, automated reasoning, and other tasks.

What is a flow chart?

Is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

What is pseudocode?

Is an informal way of programming description that does not require any strict programming language syntax or underlying technology considerations. It is used for creating an outline or a rough draft of a program. Pseudocode summarizes a program’s flow, but excludes underlying details. System designers write pseudocode to ensure that programmers understand a software project's requirements and align code accordingly.

What is a programming language?

Is a formal language comprising a set of instructions that produce various kinds of output. Programming languages are used in computer programming to implement algorithms.

What is a web programming language?

Web programming refers to the writing, markup and coding involved in Web development, which includes Web content, Web client and server scripting and network security. The most common languages used for Web programming are XML, HTML, JavaScript, Perl 5 and PHP. Web programming is different from just programming, which requires interdisciplinary knowledge on the application area, client and server scripting, and database technology.

Raúl Cátedra Martínez

What differences exist between machine language , assembly language and high level languages?

Machine language, or machine code, is the only language that is directly understood by the computer, and it does not need to be translated. All instructions use binary notation and are written as a string of 1s and 0s.

An **assembly language** consists of a set of symbols and letters. A translator is required to translate the assembly language to machine language. This translator program is called the 'assembler.' It can be called the second generation language since it no longer uses 1s and 0s to write instructions

A **high-level language** is a programming language that uses English and mathematical symbols, like +, -, % and many others, in its instructions. When using the term 'programming languages,' most people are actually referring to high-level languages. High-level languages are the languages most often used by programmers to write programs.

What are variables and what are they used for?

Is a storage location paired with an associated symbolic name, which contains some known or unknown quantity of information referred to as a value. The variable name is the usual way to reference the stored value, in addition to referring to the variable itself, depending on the context. This separation of name and content allows the name to be used independently of the exact information it represents. The identifier in computer source code can be bound to a value during run time, and the value of the variable may thus change during the course of program execution.

Are used to store information to be referenced and manipulated in a computer program. They also provide a way of labeling data with a descriptive name, so our programs can be understood more clearly by the reader and ourselves. It is helpful to think of variables as containers that hold information. Their sole purpose is to label and store data in memory. This data can then be used throughout your program.

What types of variables exist and what each type is used for?

The most extended clasification of variable types is by the data type that contain. Based in this clasification, we can define the next types:

- Bool: it´s a fundamental variable type that can take only two values: 1 and 0, which correspond to true and false.
- Int: It's used to define numeric variable holding whole numbers. Only negarive and positive whole numbers can be stored in int variables.
- Float, double and decimal: these three types of variable habdle whole numbers, numbers with decimals and fractios. The difference between the three lies in the range of values. For example, double is twice the size of float, and it accommodates more digits.
- Char: consits of Unicode characters (the letters that represent most of the written languages).
- String: it's a characters ordered sequence.

Raúl Cátedra Martínez

Do all programming languages make a distinction in the type of variables?

Yes, but not in all languages you have to define the type, like in JavaScript

What is a global variable?

Are those that are in scope for the duration of the programs execution. They can be accessed by any part of the program, and are readwrite for all statements that access them.

What is a local variable?

Are those that are in scope within a specific part of the program (function, procedure, method, or subroutine, depending on the programming language employed).

What is an environment variable?

Is a variable whose value is set outside the program, typically through functionality built into the operating system or microservice. An environment variable is made up of a name/value pair, and any number may be created and available for reference at a point in time.

Raúl Cátedra Martínez

What are operators and what are they used for?

Is a symbol that tells the compiler or interpreter to perform specific mathematical, relational or logical operation and produce final result.

a. Know the assignment operators

Is the operator used to assign a new value to a variable.

- When using the "=" operator for an assignment with the left operand as the property or indexer access, the property or indexer must have a set accessor.
- Overloading a binary operator implicitly overloads its corresponding assignment operator (if any).
- The different assignment operators are based on the type of operation performed between two operands such as addition (+=), subtraction, (-=), etc. The meaning of the operator symbol used depends on the type of the operands.
- Assignment operators are right-associative, which means they are grouped from right to left.
- Although assignment using assignment operator (a += b) achieves the same result as that without (=a +b), the difference between the two ways is that unlike in the latter example, "a" is evaluated only once.
- The assignment operator usually returns a reference to the object so as to be used in multiple assignments made in a single statement such as "a=b=c", where a, b and c are operands.
- The assignment operator expects the type of both the left- and right-hand side to be the same for successful assignment.

b. Know the increment and decrement operators

Increment and decrement operators are unary operators that *add* or *subtract* one, to or from their operand, respectively. In languages syntactically derived from B (including C and its various derivatives), the increment operator is written as ++ and the decrement operator is written as --.

c. Know the logical operators

Is a symbol or word used to connect two or more expressions such that the value of the compound expression produced depends only on that of the original expressions and on the meaning of the operator. Common logical operators include AND, OR, and NOT. The most common way to write this operators are &&, || and ! respectively.

d. Know the mathematical operators

Are the operators to manipulate numbers, addingm subtractig, multiplying and dividing two numbers or two numbers represented by variables. The most common way to writte this operators are + for addition, - for subtraction, / for division, \ for integer division, mod for modulo, * for multiplication and ^ for exponentiation.

e. Know the relational operators

Is a programming language construct or operator that tests or defines some kind of relation between two entities. These include numerical equality and inequalities. The most common are: equal to (==), not equal to (!=), greater than (>), less than (<), greater than or equal to (>=) and less than or equal to (<=).

What is a constant and understand its characteristics

A constant is a value that cannot be altered by the program during normal execution. Constants are used in two ways. They are:

- Literal constant: A literal constant is a value you type into your program wherever it is needed.
- Defined constant: its the constant you defined in the program.

What are flow control structures and what are they used for?

A control structure is like a block of programming that analyses variables and chooses a direction in which to go based on given parameters. The term *flow control* details the direction the program takes.

Raúl Cátedra Martínez

What are the main flow control structures

There are three principal flow control structures:

- IF and IF-ELSE: is a simple control that tests whether a condition is true or false and if the condition is true, then an action occurs. If the condition is false, nothing is done. It can be write like a if-else statement, who the control can "look both ways" so to speak, and take a secondary course of action. If the condition is true, then an action occurs. If the condition is false, take an alternate action.
- WHILE and DO-WHILE: is a process in which a loop is initiated until a condition has been met. This structure is useful when performing iterative instructions to satisfy a certain parameter. A DO-WHILE loop is nearly the exact opposite to a WHILE loop. A WHILE loop initially checks to see if the parameters have been satisfied before executing an instruction. A DO-WHILE loop executes the instruction before checking the parameters.
- FOR: Is an extension of a while loop. A for loop usually has three commands. The first is used to set a starting point . The second is the end condition and is run every round. The third is also run every round and is usually used to modify a value used in the condition block.

What are the most used programming languages?

It´s a dificult question, because it´s depend on the things you want to program, for example if you want to programming software you will use C, but if you want to program webs you will use HTML, CSS and JavaScrpit. In general terms of programming languages nowadays Python is the most used programming language followed by JavaScript, Java and C#.

What is a function and what is it used for?

A function is a block of organized, reusable code that is used to perform a single, related action. Functions provide better modularity for your application and a high degree of code reusing.

What is a function parameter and what is it used for?

Parameters identify values that are passed into a function. For example, a function to add three numbers might have three parameters. A function has a name, and it can be called from other points of a program. When that happens, the information passed is called an argument. Modern programming languages typically allow functions to have several parameters.

What is the output data of a function?

It's the data that a function give back before the execution

Raúl Cátedra Martínez

What is meant by object-oriented programming?

Is a programming paradigm based on the concept of "objects", which can contain data and code: data in the form of fields (often known as *attributes* or *properties)*, and code, in the form of procedures (often known as methods).

What is meant by "compiler"?

It´s a program that converts instructions into a machine-code or lower-level form so that they can be read and executed by a computer.

What is meant by IDE?

An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of at least a source code editor, build automation tools and a debugger.

What is Pair Programming and what advantages does it have?

Pair programming consists of two programmers sharing a single workstation (one screen, keyboard and mouse among the pair). The programmer at the keyboard is usually called the “driver”, the other, also actively involved in the programming task but focusing more on overall direction is the “navigator”; it is expected that the programmers swap roles every few minutes or so. The advantages are:

- Increased code quality: “programming out loud” leads to clearer articulation of the complexities and hidden details in coding tasks, reducing the risk of error or going down blind alleys.
- Better diffusion of knowledge among the team, in particular when a developer unfamiliar with a component is pairing with one who knows it much better.
- Better transfer of skills, as junior developers pick up micro-techniques or broader skills from more experienced team members.
- Large reduction in coordination efforts, since there are N/2 pairs to coordinate instead of N individual developers.
- Improved resiliency of a pair to interruptions, compared to an individual developer: when one member of the pair must attend to an external prompt, the other can remains focused on the task and can assist in regaining focus afterwards.

Raúl Cátedra Martínez

What is a repository?

A repository is like a folder for your project. Your project's repository contains all of your project's files and stores each file's revision history. You can also discuss and manage your project's work within the repository.

What do we mean when we talk about implementation?

Implementation is often used in the tech world to describe the interactions of elements in programming languages. In Java, where the word is frequently used, to implement is to recognize and use an element of code or a programming resource that is written into the program.

What a debugger is and what it is used for?

It´s a software whith it´s main use of a debugger is to run the target program under controlled conditions that permit the programmer to track its operations in progress and monitor changes in computer resources (most often memory areas used by the target program or the computer's operating system) that may indicate malfunctioning code.

What are comments in the code and what are they used for?

In computer programming, a comment is a programmer-readable explanation or annotation in the source code of a computer program. They are added with the purpose of making the source code easier for humans to understand, and are generally ignored by compilers and interpreters.