

Computer instructions, aka programs

[Technology](#), [Computer](#)



Computer Instructions, aka Programs Using the information in the Lectures (Computer Instructions, aka Programs) presentation, answer the following: - Name and describe the only language that computers understand and explain how the instructions that people write for computers get into that form. Computers only understand one language and that is the machine language (binary code). This type of language is represented in a two digit sequence zero (0) or one (1). Computers use this machine language to represent every task that they perform by means of electrical current that go through switches (capacitors). If a switch has current it means that the switch is "ON" in machine language this will be represented as the number "one". If there is no current going through the switch then it is considered to be "OFF" and represented in binary code as the number "zero". Now, the way people communicate with computers is not directly by the language that the computer understands (machine language), but instead a set of instructions have to be written in a different language that specializes in writing programs such as C++ or Visual Basic. These types of languages at the same time have to be translated into machine language either through compilers or interpreters. Compilers translate all the source codes (C++, Visual Basic, etc. language) in a program to machine language before it's executed. Then after that the compiled program (in this case machine language) is executed by the Central Process Unit (CPU). At this time the machine language can be used every time the program is run unless the source code changes, in which then the machine language should be recompiled. Like compilers, interpreters also translate the source code into machine language, but they only translate each line of the code as it is

executed. Difference is also that every time the program is run each line must be translated again. Which in comparison to compilers, interpreters are much slower.