Recipe for Computing

Remember that in the 1950s, computer scientists used punch cards to store programs. These punch cards represented bits that the CPU would read and then perform a series of instructions based on what the program was. The binary code could have looked like this, and the instructions will be translated to this, grab some input data from this location in memory. Using the input data, do some math, then put some output data into this location in memory. But storing programs on punch cards was a long and tedious task. The programs had to be kept on stacks and stacks of punch cards. Computer scientists needed a better way to send instructions to a machine, but how? Eventually a language was invented called assembly language, that allowed computer scientists to use human readable instructions assembled into code that the machines could understand. Instead of generating binary code, computer scientists could program using machine instructions like this. Take integer from register 1, take integer from register 2, add integer from register 1 and register 2 and output to register 4. This example makes it look like a human could read it, but don't be fooled. Let's take an example of saying something simple like, hello world in assembly language. It looks pretty robotic, don't get me wrong that's still an improvement over its binary code cousin. But assembly language was still a thin veil from machine code. It's still didn't let computer programmers use real human words to build a program, and a program that was written for a specific CPU could only be run on that CPU or family of CPUs. A program was needed that could run on many types of CPUs, enter compiled programming languages. A compiled programming language uses human readable instructions and sends them through a compiler. The compiler takes the human instructions and compiles them into machine instructions. Admiral Grace Hopper invented this to help make programming easier. Compilers are a key component to programming and helped pave the road that led us to today's modern computing. Thanks to compilers, we can now use something like this, and it would be the same thing as this. Computer scientists have developed hundreds of programming languages in the past couple of decades to try and abstract the different CPU instructions into simpler commands. Along the way, another type of language emerged that was interpreted rather than compiled, interpreted languages aren't compiled ahead-of-time. A file that has code written in one of these languages is usually called a script. The script is run by an interpreter which interprets the code into CPU instructions just in time to run them. You'll learn how to write code using a scripting language later in this program as an IT support specialist, scripting can help you by harnessing the power of a computer to perform tasks on your behalf, allowing you to solve a problem once and then move on to the next thing. Programming languages are used to create programs that can be run to perform a task or many tasks.

Revision #1
Created 2023-05-29 13:29:30 UTC by naruzkurai
Updated 2023-05-29 14:13:00 UTC by naruzkurai