Should I learn Java C or Haskell?

I will first outline these 3 different languages, showing you what each of them do and how they are often used. I will then explain their advantages and disadvantages. Finally, I will tell you what I think the best language to learn is based on what it is you want to do.

**C**

C is a fairly old language that is sort of the next step up after assembly language (assembly is the actual 1s and 0s applied to the processor for different processor commands). C is a very low level language which makes it possible to do exactly what you want. The problem with being able to do exactly what you want is that you have to do a lot of coding, unless you can find someone else’s open source code online that will work for you, or a library with a function that does what you want. The problem with this is that there isn’t a real standard way of doing some things, so it may take a while to understand what a function does, and how to use it.

C is a great language when it comes to optimizing for very quick programs (remember, less code doesn’t necessarily mean a faster program). Optimization can take time, but you have more control over things when it comes to C which makes it easier to optimize. You must also understand how computers function (how bits are read, etc.) to be truly good at optimizing. C is a procedural language, but if you require object oriented, you can move into C++ which uses C as a base. You must also be careful what libraries you use because they may not be available on other systems, like some windows libraries won’t work on linux. You must also compile your code separately for different operating systems because compiling on linux will not produce a program that will execute in windows.

**Java**

Java is a high level object oriented language. This means your code is built into objects, so if you are trying to create a person, one class (object) may be the head, and another class may be the eye ball. In this situation you could have the head class use the eyeball class. This method of programming is great for graphical user interfaces because graphical user interfaces have many components that may or may not work with each other directly, and may or may not need to be used at the same time. Something that makes java very easy to use is the editor you use it in. Many people like to use vim or other command line editors, but other programs like eclipse or netbeans will work much better when it comes to java because they keep you from having to memorize every single function name. You do still need to know what libraries you want though. The last thing I will mention about java is that it is cross platform. This means that you need only compile on a computer that has java installed be it mac, linux, or windows, and the executable will work on any computer that has java installed. Of course this does mean that the computer must have java installed to use the program, but most computers do.

**Haskell**

Haskell is a language that is probably unlike any programming language you have ever seen. If you have ever done any programming before, then java and c should not be too difficult to learn. Haskell however is a different beast altogether. Haskell is a functional programming language. This means that it treats everything as functions. Haskell is similar to prolog, but much better. Actually it’s kind of like a programming language with both c functionality, and prolog functionality. Something nice about Haskell is that you can nest functions, so one function can be an input for another function, plus these functions are evaluated using lazy evaluation. This means that variables are not evaluated until they have to be. Something else that is nice about Haskell is that it is very easy to work with lists in it.

**Conclusion**

Currently, java and c are much more heavily used in industry which means that’s more along the lines of what employers will be looking for. Java and C are also older and have more support, so if you often get stuck and need help from others, then the support may not be there for you in the Haskell world. If you need optimization, then C is the only real choice since Java and Haskell are both high level languages. Currently graphical user interfaces would be the easiest to create using Java, however there is also visual C++ for windows. If you are looking to program for the use in math fields, then Haskell is probably going to simplify your work significantly.

Personally, I use C for low level coding like coding on a microchip. I will use Java for programs that need graphical user interfaces, and anything I don’t have to use C for like controlling a serial port. Finally, I use Haskell for programming mathematic programs, like fourier series, or sorting data. Haskell is by far my favourite language, but because it is hard to use it with GUIs and the support is not a strong yet, I will defer to Java whenever required.