Programmable devices have existed for centuries..  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
Normally the first step in debugging is to attempt to reproduce the problem.  
For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software.  
 Allen Downey, in his book How To Think Like A Computer Scientist, writes:  
 Many computer languages provide a mechanism to call functions provided by shared libraries.  
One approach popular for requirements analysis is Use Case analysis.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Ideally, the programming language best suited for the task at hand will be selected.  
The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
However, readability is more than just programming style.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Techniques like Code refactoring can enhance readability.  
 Different programming languages support different styles of programming (called programming paradigms).  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.