

The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists. Following a consistent programming style often helps readability. Use of a static code analysis tool can help detect some possible problems. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems. Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code. In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams. Code-breaking algorithms have also existed for centuries. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Scripting and breakpointing is also part of this process. Many applications use a mix of several languages in their construction and use. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory. Programmable devices have existed for centuries. 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. Code-breaking algorithms have also existed for centuries. Programmable devices have existed for centuries. There are many approaches to the Software development process. While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se. However, readability is more than just programming style. When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear. Different programming languages support different styles of programming (called programming paradigms).