

As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Their jobs usually involve: Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers. 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. Following a consistent programming style often helps readability. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. Programs were mostly entered using punched cards or paper tape. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). 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. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Also, specific user environment and usage history can make it difficult to reproduce the problem. Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code. 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. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.