

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. Normally the first step in debugging is to attempt to reproduce the problem. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Use of a static code analysis tool can help detect some possible problems. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). 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. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. There are many approaches to the Software development process. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. 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. Programs were mostly entered using punched cards or paper tape. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. 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. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.