Relatedly, software engineering combines engineering techniques and principles with software development.  
Many applications use a mix of several languages in their construction and use.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Ideally, the programming language best suited for the task at hand will be selected.  
Programming languages are essential for software development.  
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.  
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.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
To produce machine code, the source code must either be compiled or transpiled.  
Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).  
Some of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.