Ideally, the programming language best suited for the task at hand will be selected.  
This is interpreted into machine code.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
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.  
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).  
Also, those involved with software development may at times engage in reverse engineering, which is the practice of seeking to understand an existing program so as to re-implement its function in some way.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
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.  
 Tasks accompanying and related to programming include testing, debugging, source code maintenance, implementation of build systems, and management of derived artifacts, such as the machine code of computer programs.