There exist a lot of different approaches for each of those tasks.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
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).  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
To produce machine code, the source code must either be compiled or transpiled.  
However, readability is more than just programming style.  
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.  
Ideally, the programming language best suited for the task at hand will be selected.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
There are many approaches to the Software development process.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
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.