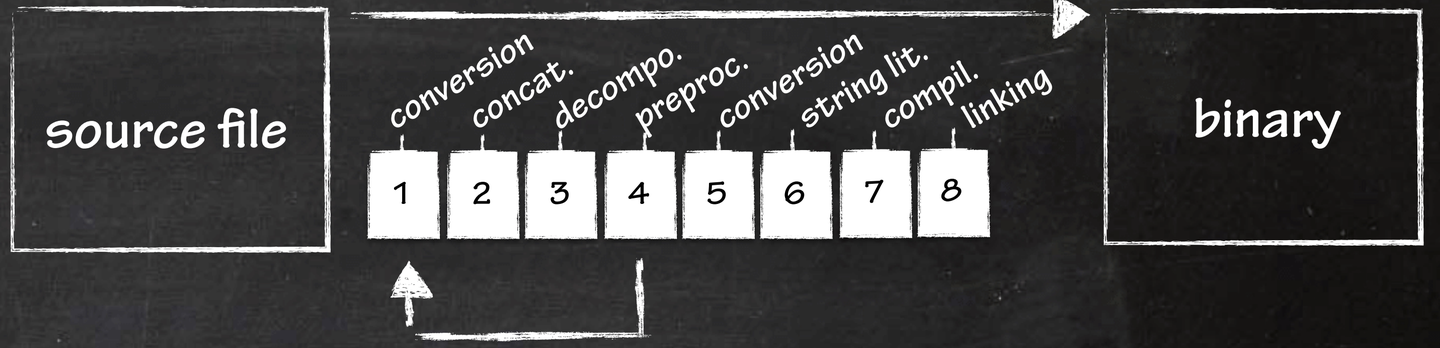
Translation (1)



Phase 1

Source Character Set Conversion. Conversion of the source file text file characters into text containing only characters from the source character set:

- 5 types of whitespace characters
  - space, horizontal tab, vertical tab, form feed, new-line
- 10 digit characters (0 to 9)
- 52 letters (‚a' to ‚z', 'A' to ‚Z')
- 29 punctuation characters (_ { } [ ] # ( ) < > % : ; . ? * + - / ^ & | ~ ! = , \ " ')
- Trigraphs are replaced by single-character representations

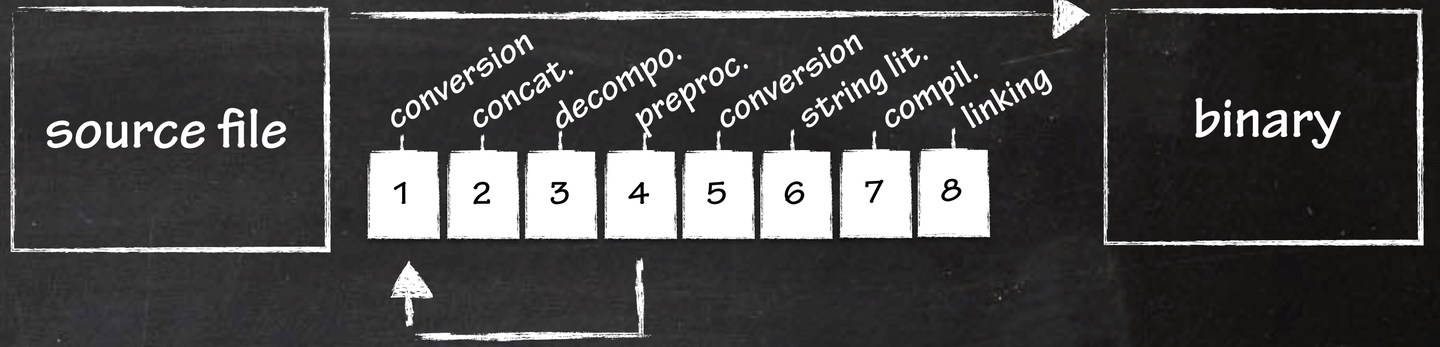
Phase 2

Concatenation. Replacement of new line characters, i.e., concatenate physical lines into one logical line

- does not apply to empty lines
  - long logical lines

based on: http://en.cppreference.com/w/c/language/translation_phases

Translation (2)



Phase 3

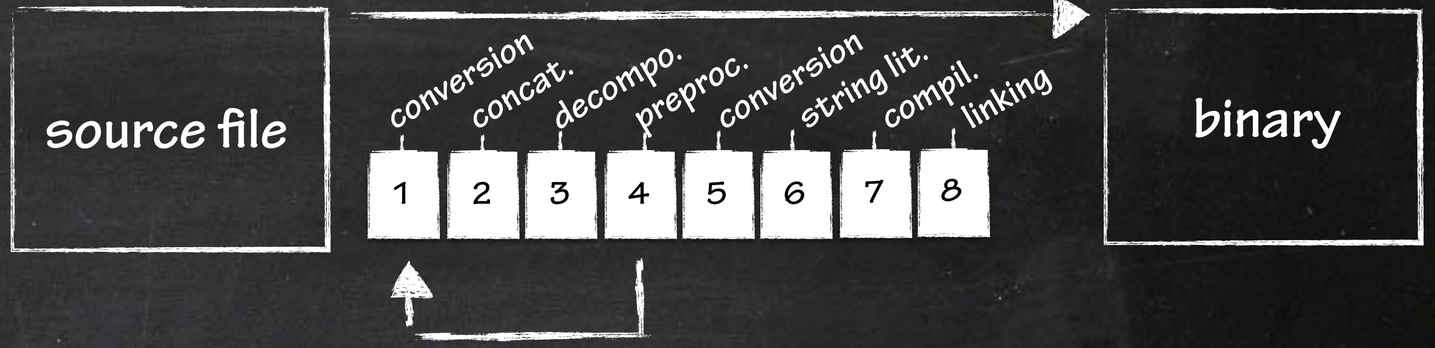
Decomposition into comments, whitespace sequences, and preprocessing tokens:

- header names: <stdio.h> or "file.h"
- identifiers and numbers
- character constants and string literals
- operators
  - ! % ^ & * - + = ~ | . < > / ? : , [ ] ( ) #
  - ++ -- -> << >> <= >= == != *= /=
  - %= += -= <<= >>= &= ^= |= ## && ||
- punctuators
  - < > header name
  - [ ] array delimiter
  - { } initializer list, or function body, or compound statement
  - ( ) function parameter list delimiter
  - * pointer declaration
  - , list separator
  - : statement label
  - = declaration initializer
  - ; end statement
  - ... variable-length argument list
  - # preprocessor directive
  - ' ' character constant
  - " " string literal or header name

based on: http://en.cppreference.com/w/c/language/translation_phases

based on: https://www.cs.auckland.ac.nz/references/unix/digital/AQTLTBTE/DOCU_008.HTM

Translation (3)



Phase 3

- replacement of comments by one whitespace character
- newlines are untouched

Phase 4

Preprocessing

- sources are preprocessed by the C preprocessor (see later)
- recursively apply phase 1 to 4 for all includes files (#include directive)
  - ends up in complete preprocessed file

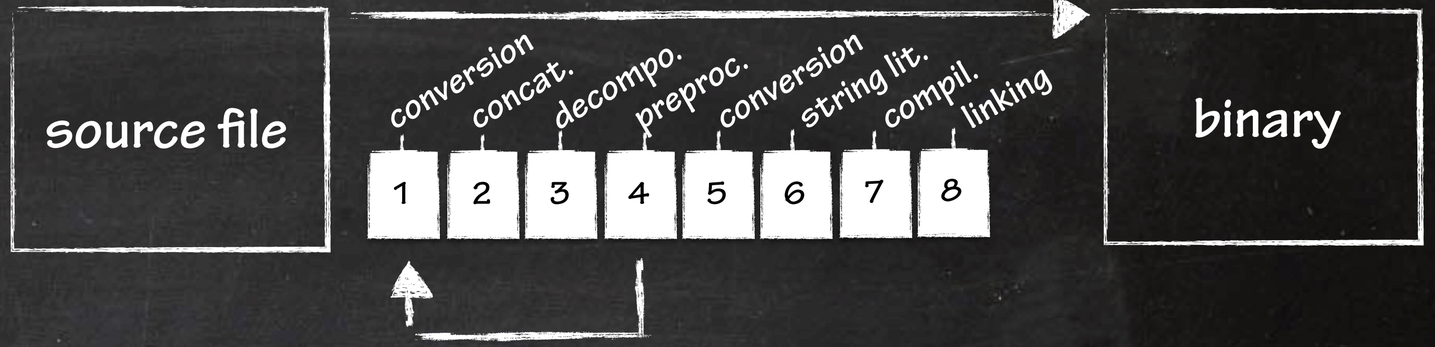
Phase 5

Conversion to Execution Character Set

- Conversion of characters & escape sequences in strings to execution character set
  - often multibyte character set such as UTF-8

based on: http://en.cppreference.com/w/c/language/translation_phases

Translation (4)



Phase 6

Phase 7

Phase 8

binary/executable

Concatenation of String Literals

"Hello" "World" becomes "Hello World"

Complication

- analyzing of tokens (syntactically and semantically)
- translation into translation unit

Linking

- translation units and libraries are bound to external references

based on: http://en.cppreference.com/w/c/language/translation_phases

Program Behavior

keep this in mind!

C language standard specifies observable behavior of programs…
… with the following exceptions in behavior:

- Undefined: it is not defined how the program behaves by illegal memory accesses out of bound, signed integer overflow, null pointer dereference,…
- Unspecified: more than one behavior is possible at the same time, e.g., order of evaluation; result in valid results but may be different when repeated
- Implementation-defined: implementation documents which behavior for unspecified behavior is chosen, e.g., bit number of byte
- Locale-specific: implementation-defined behavior that depends on locale, e.g., whether a given letter other than the 26 latin letters is uppercase or not

taken on: http://en.cppreference.com/w/c/language/behavior