HTML Code Count™ Counting Standard

University of Southern California

Center for Systems and Software Engineering

December , 2016

Revision Sheet

Date	Version	Revision Description	Author
6/10/2016	1.0	Original Release	Derek Lengenfelder
10/20/2008	Version No.	Click here to Description.	Name

Table of Contents

1. Definitions

1.1. **SLOC –** Source Lines of Code is a unit used to measure the size of software program. SLOC counts the program source code based on a certain set of rules. SLOC is a key input for estimating project effort and is also used to calculate productivity and other measurements.

1.2. **Physical SLOC –** One physical SLOC is corresponding to one line starting with the first character and ending by a carriage return or an end-of-file marker of the same line, and which excludes the blank and comment line.

1.3. **Logical SLOC –** Lines of code intended to measure "statements", which normally terminate by a semicolon (C/C++, Java, C#) or a carriage return (VB, Assembly), etc. Logical SLOC are not sensitive to format and style conventions, but they are language-dependent. For HTML, logical statements are based on tags.

1.4. **Data declaration line or data line** - A line containing declaration of data and used by a compiler or assembler to interpret other program elements.

1.5. **Compiler directive** - A statement that tells the compiler how to compile a program, but not what to compile. CSS does not have any compiler directives.

1.6. **Blank Line –** A physical line of code, which contains any number of white space characters (spaces, tabs, form feed, carriage return, line feed, or their derivatives).

1.7. **Comment Line –** A comment is defined as a string of zero or more characters that follow language-specific comment delimiter. The HTML comment delimiter is "<!--". A whole comment line may span one line and does not contain any compilable source code. An embedded comment can co-exist with compilable source code on the same physical line. Banners and empty comments are treated as types of comments.

1.8. **Executable Line of code –** A line that contains software instruction executed during runtime and on which a breakpoint can be set in a debugging tool. An instruction can be stated in a simple or compound form.

An executable line of code may not contain the following statements:

- Data declaration (data) lines
- Whole line comments, including empty comments and banners
- Blank lines

2. Checklist for source statement counts

PHYSICAL SLOC COUNTING RULES			
MEASUREMENT UNIT	ORDER OF PRECEDENCE	PHYSICAL SLOC	COMMENTS
Executable lines	1	One per line	Defined in 1.8
Non-executable lines			
Declaration (Data) lines	2	One per line	Defined in 1.4
Compiler directives	3	One per line	Defined in 1.5
Comments			Defined in 1.7
On their own lines	4	Not included (NI)	
Embedded	5	NI	
Banners	6	NI	
Empty comments	7	NI	
Blank lines	8	NI	Defined in 1.6

LOGICAL SLOC COUNTING RULES				
NO.	STRUCTURE	ORDER OF PRECEDENCE	LOGICAL SLOC RULES	COMMENTS
R01	• Declarations (document type, attribute, entity, text, notation, etc.) • Processing instruction • CDATA section, and • Conditional section	1	Count once per occurrence	
R02	A pair of start-tag and end-tag	2	Count once	
R03	Empty element tag	3	Count once per occurrence	

3. Examples

DECLARATION LINES

D01 – Processing instruction

GENERAL EXAMPLE	SPECIFIC EXAMPLE	SLOC COUNT
'<?' PITarget (S (Char* - (Char* '?>' Char*)))? '?>'	<?dbhtml background-color="color"?>	1 0

D02 – Document type

GENERAL EXAMPLE	SPECIFIC EXAMPLE	SLOC COUNT
'<!DOCTYPE' S Name (S ExternalID)? S? ('[' (markupdecl | DeclSep)* ']' S?)? '>'	<!DOCTYPE html>	1

ELEMENT LINES

E01 - Start-tag & end-tagsEmpty element

GENERAL EXAMPLE	SPECIFIC EXAMPLE	SLOC COUNT
'<' Name (S Attribute)* S? '>'	<HTML> <BODY>	1 1
'<' Name (S Attribute)* S? '/>'	<P> to break lines in a paragraph, use the br tag. </P> </BODY> </HTML>	1 2 2 0 0 0

E02 - Comment Delimiter

GENERAL EXAMPLE	SPECIFIC EXAMPLE	SLOC COUNT
<!-- this is a comment -->	<!DOCTYPE html> <!--This is a comment --> <html> <body> To break lines . 	1 0 1 1 1 3 0

	</body> </html>	0 0

E03 - String delimiter(s); literals; escape characters; nesting, etc.

GENERAL EXAMPLE	SPECIFIC EXAMPLE	SLOC COUNT
Escape Character - Both Entity and character references may be used to escape the delimiters. HTML- < > & ' " (non-exhaustive)	<!DOCTYPE html> <html> <body> < DataSet> <Order> <Customer >439 </Customer> </Order> </DataSet> </body> </html>	1 1 1 0 0
HTML 4 DTD explicitly declares 252 character entities For E.g. ¢ &curren Numeric character - references can also be used in XML; they are expanded immediately when recognized and must be treated as character data.	<!DOCTYPE html> <html> <body>The content of the body element is displayed in your browser.</body> <tt> C H E E S E </tt> </html>	1 1 1 2 0 0

E04 - Tags with Attributes

GENERAL EXAMPLE	SPECIFIC EXAMPLE	SLOC COUNT
HTML can have attributes in the start tag. Attributes provide additional information about the element.	<note date="12/11/2002"> <to>Tove</to> <from>Jani</from> <heading>Reminder</heading> <body>Don't forget me this weekend!</body> </note>	1 1 1 1 1 0 0

4. Complexity

Complexity measures the occurrences of different keywords in code baseline. HTML does not have complexity. Therefore, its complexity is always listed as zero (0).

5. Cyclomatic Complexity

Cyclomatic complexity measures the number of linearly independent paths through a program. It is measured for each function, procedure, or method according to each specific program language. This metric indicates the risk of program complexity and also determines the number of independent test required to verify program coverage.

The cyclomatic complexity is computed by counting the number of decisions plus one for the linear path. Decisions are determined by the number of conditional statements in a function. A function without any decisions would have a cyclomatic complexity of one. Each decision such as an if condition or a for loop adds one to the cyclomatic complexity.

The cyclomatic complexity metric v (G) was defined by Thomas McCabe. Several variations are commonly used but are not included in the UCC. The modified cyclomatic complexity counts select blocks as a single decision rather than counting each case. The strict or extended cyclomatic complexity includes boolean operators within conditional statements as additional decisions.

HTML does not have cyclomatic complexity. Therefore, its cyclomatic complexity is always listed as zero (0).