

# Commenting your code

**GAM150** 





## Everyone needs to contribute code















## Resharper-C++ to automatize it

Educational license





#### Count Lines of Code

- https://github.com/AlDanial/cloc
  - Download the latest version
- Open a terminal (cmd.exe):
  - cloc-1.84 --by-file \*.cpp \*.h --csv
    --quiet -out=LOC.CSV
- Ocheck the file for Lines of Code



#### Guidelines



- Of Good quality code
  - Minimal Hardcoded Data
  - Minimal Cut and Paste

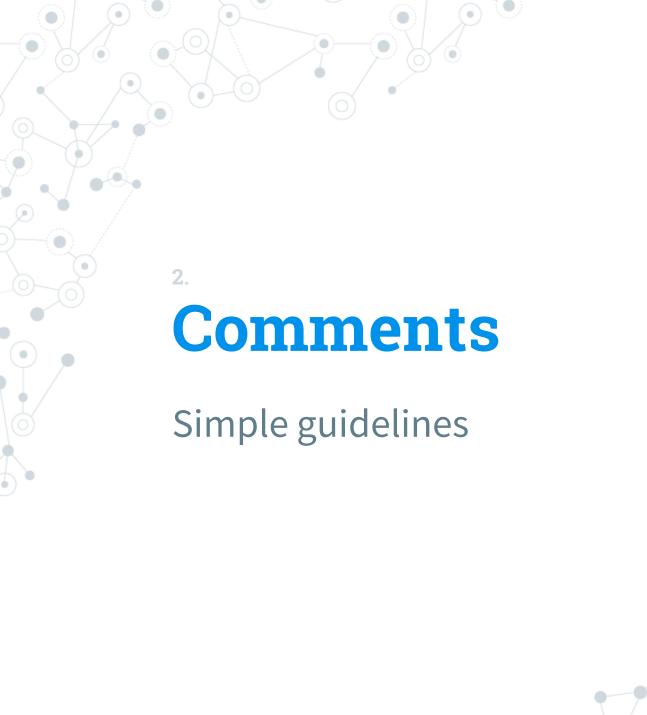


54 □static constexpr UInt32 msTab[] = {	100	0x45df5c75	144	0x12b7e950,
54 □static constexpr UInt32 msTab[] = { 55 □ 0x00000000,	101	0xdcd60dcf	145	0x8bbeb8ea,
56 0x77073096	102	0xabd13d59	146	0xfcb9887c,
57 0xee0e612c		i a	147	0x62dd1ddf,
	103	0x26d930ac,	148	0x15da2d49,
58 0x990951ba, 59 0x076dc419,	104	0x51de003a,	149	0x8cd37cf3,
- T	105	0xc8d75180,	150	0xfbd44c65,
60 0x706af48f, 61 0xe963a535,	106	0xbfd06116,	151	0x4db26158,
62 0x9e6495a3	107	0x21b4f4b5,	152	0x3ab551ce,
63 0x0edb8832,	108	0x56b3c423,	153	0xa3bc0074,
64 0x79dcb8a4	109	0xcfba9599,	154	0xd4bb30e2
65 0xe0d5e91e	110	0xb8bda50f,	155	0x4adfa541
66 0x97d2d988	111	0x2802b89e,	156	0x3dd895d7
67 0x09b64c2b	112	0x5f058808,	157	0xa4d1c46d
68 0x7eb17cbd	113	0xc60cd9b2,	158	0xd3d6f4fb
	114	0xb10be924,	159	0x4369e96a
	115	0x2f6f7c87,	160	0x346ed9fc
70 0x90bf1d91, 71 0x1db71064,	116	0x58684c11,	161	0xad678846
	117	0xc1611dab,	162	0xda60b8d0
72 0x6ab020f2,	118	0xb6662d3d,	163	0x44042d73,
73 0xf3b97148, 74 0x84be41de	119	0x76dc4190,	164	0x33031de5
	120	0x01db7106,	165	0xaa0a4c5f
75 0x1adad47d,	121	0x98d220bc,	166	0xdd0d7cc9.
76 0x6ddde4eb,	122	0xefd5102a,		0x5005713c,
77 0xf4d4b551,	123	0x71b18589,	167	0x270241aa
78 0x83d385c7,	124	0x06b6b51f,	168	0xbe0b1010.
79 0x136c9856,	125	0x9fbfe4a5,	169	
80 0x646ba8c0,	126	0xe8b8d433	170	0xc90c2086,
81 0xfd62f97a,	127	0x7807c9a2	171	0x5768b525
82 0x8a65c9ec,	128	0x0f00f934	172	0x206f85b3,
83 0x14015c4f,	129	0x9609a88e	173	0xb966d409,
84 0x63066cd9,	130	0xe10e9818	174	0xce61e49f,
85 0xfa0f3d63,	131	0x7f6a0dbb	175	0x5edef90e,
86 0x8d080df5,	132	0x086d3d2d	176	0x29d9c998,
87 0x3b6e20c8,	133	0x91646c97	177	0xb0d09822,
88 0x4c69105e,	134	0xe6635c01	178	0xc7d7a8b4,
89 0xd56041e4,	135	0x6b6b51f4	179	0x59b33d17,
90 0xa2677172,	136	0x1c6c6162	180	0x2eb40d81,
91 0x3c03e4d1,	137	0x856530d8	181	0xb7bd5c3b,
92 0x4b04d447,	138	0xf262004e	182	0xc0ba6cad,
93 0xd20d85fd,		0x6c0695ed	183	0xedb88320,
94 0xa50ab56b,	139		184	0x9abfb3b6,
95 0x35b5a8fa,		0x1b01a57b,	185	0x03b6e20c,
96 0x42b2986c,	141	0x8208f4c1,	186	0x74b1d29a,
97 0xdbbbc9d6,	142	0xf50fc457,	187	0xead54739,
98 0xacbcf940,	143	0x65b0d9c6,	188	0x9dd277af,
99 0x32d86ce3,	144	0x12b7e950,	189	0x04db2615,

Etc...

```
static constexpr UInt32 msTab[] = {
 0x000000000, 0x77073096, 0xee0e612c, 0x990951ba, 0x076dc419, 0x706af48f, 0xe963a535, 0x9e6495a3, 0x0edb8832, 0x79dcb8a4, 0xe0d5e91e, 0x97d2d988,
 0x09b64c2b, 0x7eb17cbd, 0xe7b82d07, 0x90bf1d91, 0x1db71064, 0x6ab020f2, 0xf3b97148, 0x84be41de, 0x1adad47d, 0x6ddde4eb, 0xf4d4b551, 0x83d385c7,
 0x136c9856, 0x646ba8c0, 0xfd62f97a, 0x8a65c9ec, 0x14015c4f, 0x63066cd9, 0xfa0f3d63, 0x8d080df5, 0x3b6e20c8, 0x4c69105e, 0xd56041e4, 0xa2677172,
 0x3c03e4d1. 0x4b04d447. 0xd20d85fd. 0xa50ab56b. 0x35b5a8fa. 0x42b2986c. 0xdbbbc9d6. 0xacbcf940. 0x32d86ce3. 0x45df5c75. 0xdcd60dcf. 0xabd13d59.
 0x26d930ac, 0x51de003a, 0xc8d75180, 0xbfd06116, 0x21b4f4b5, 0x56b3c423, 0xcfba9599, 0xb8bda50f, 0x2802b89e, 0x5f058808, 0xc60cd9b2, 0xb10be924,
 0x2f6f7c87, 0x58684c11, 0xc1611dab, 0xb6662d3d, 0x76dc4190, 0x01db7106, 0x98d220bc, 0xefd5102a, 0x71b18589, 0x06b6b51f, 0x9fbfe4a5, 0xe8b8d433,
 0x7807c9a2, 0x0f00f934, 0x9609a88e, 0xe10e9818, 0x7f6a0dbb, 0x086d3d2d, 0x91646c97, 0xe6635c01, 0x6b6b51f4, 0x1c6c6162, 0x856530d8, 0xf262004e,
 0x6c0695ed, 0x1b01a57b, 0x8208f4c1, 0xf50fc457, 0x65b0d9c6, 0x12b7e950, 0x8bbeb8ea, 0xfcb9887c, 0x62dd1ddf, 0x15da2d49, 0x8cd37cf3, 0xfbd44c65,
 0x4db26158, 0x3ab551ce, 0xa3bc0074, 0xd4bb30e2, 0x4adfa541, 0x3dd895d7, 0xa4d1c46d, 0xd3d6f4fb, 0x4369e96a, 0x346ed9fc, 0xad678846, 0xda60b8d0,
 0x44042d73, 0x33031de5, 0xaa0a4c5f, 0xdd0d7cc9, 0x5005713c, 0x270241aa, 0xbe0b1010, 0xc90c2086, 0x5768b525, 0x206f85b3, 0xb966d409, 0xce61e49f,
 0x5edef90e, 0x29d9c998, 0xb0d09822, 0xc7d7a8b4, 0x59b33d17, 0x2eb40d81, 0xb7bd5c3b, 0xc0ba6cad, 0xedb88320, 0x9abfb3b6, 0x03b6e20c, 0x74b1d29a,
 0xead54739, 0x9dd277af, 0x04db2615, 0x73dc1683, 0xe3630b12, 0x94643b84, 0x0d6d6a3e, 0x7a6a5aa8, 0xe40ecf0b, 0x9309ff9d, 0x0a00ae27, 0x7d079eb1,
 0xf00f9344, 0x8708a3d2, 0x1e01f268, 0x6906c2fe, 0xf762575d, 0x806567cb, 0x196c3671, 0x6e6b06e7, 0xfed41b76, 0x89d32be0, 0x10da7a5a, 0x67dd4acc,
 0xf9b9df6f, 0x8ebeeff9, 0x17b7be43, 0x60b08ed5, 0xd6d6a3e8, 0xa1d1937e, 0x38d8c2c4, 0x4fdff252, 0xd1bb67f1, 0xa6bc5767, 0x3fb506dd, 0x48b2364b,
 0xd80d2bda, 0xaf0a1b4c, 0x36034af6, 0x41047a60, 0xdf60efc3, 0xa867df55, 0x316e8eef, 0x4669be79, 0xcb61b38c, 0xbc66831a, 0x256fd2a0, 0x5268e236,
 0xcc0c7795, 0xbb0b4703, 0x220216b9, 0x5505262f, 0xc5ba3bbe, 0xb2bd0b28, 0x2bb45a92, 0x5cb36a04, 0xc2d7ffa7, 0xb5d0cf31, 0x2cd99e8b, 0x5bdeae1d,
 0x9b64c2b0, 0xec63f226, 0x756aa39c, 0x026d930a, 0x9c0906a9, 0xeb0e363f, 0x72076785, 0x05005713, 0x95bf4a82, 0xe2b87a14, 0x7bb12bae, 0x0cb61b38,
 0x92d28e9b, 0xe5d5be0d, 0x7cdcefb7, 0x0bdbdf21, 0x86d3d2d4, 0xf1d4e242, 0x68ddb3f8, 0x1fda836e, 0x81be16cd, 0xf6b9265b, 0x6fb077e1, 0x18b74777,
 0x88085ae6, 0xff0f6a70, 0x66063bca, 0x11010b5c, 0x8f659eff, 0xf862ae69, 0x616bffd3, 0x166ccf45, 0xa00ae278, 0xd70dd2ee, 0x4e048354, 0x3903b3c2,
 0xa7672661, 0xd06016f7, 0x4969474d, 0x3e6e77db, 0xaed16a4a, 0xd9d65adc, 0x40df0b66, 0x37d83bf0, 0xa9bcae53, 0xdebb9ec5, 0x47b2cf7f, 0x30b5ffe9,
 0xbdbdf21c, 0xcabac28a, 0x53b39330, 0x24b4a3a6, 0xbad03605, 0xcdd70693, 0x54de5729, 0x23d967bf, 0xb3667a2e, 0xc4614ab8, 0x5d681b02, 0x2a6f2b94,
 0xb40bbe37, 0xc30c8ea1, 0x5a05df1b, 0x2d02ef8d
```

#### Prefer this, always



### **Comment Style**



#### Style

```
/**
  * \file
  * \brief
  * \author
  * \copyright
  */
```

#### Automatic documentation

- Doxygen
- Visual Studio
- Resharper
- O ...



#### Useful tags

```
\file
\namespace
\struct
\class

\var variable, typedef or enum value.
\enum enumeration type.

\def for a #define.
\typedef to document a type definition.

\param Parameter description
```



```
\file
                GameObject.h
 * \brief
                Declares the Game Object Class
  \author
                Yannick Gerber, 100% Code Contribution
    The GameObject class is at the center of the GamePlay Architectures.
    All the systems will access it and call its functions ( Update, render...)
   \copyright
                Copyright (c) 2020 DigiPen Institute of Technology. Reproduction
                 written consent of DigiPen Institute of Technology is prohibited.
#pragma once
#include "Core.h"
∃namespace BasicEngine
     * \struct GameObject
     * \brief Base structure for all the GameObjects.
     * \note Inherit from this if you want to have an Object automatically
            handled by the engine
    struct GameObject : public IGameObject
```

#### With ReSharper

```
virtual void
                    DebugDisplay(char fontID, float& nextX, float& nextY, float glyphHeight) = 0;
 virtual UInt32 GetID()
                                 (public method) virtual void DebugDisplay(char_fontlD, float&_nextX, float&_nextY, float_glyphHeight) = 0
                                   in struct IGameObject
 virtual ~IGameObject()
                                 Display a debug message on screen at the given coordinates
                                 Parameters:
                                             ID of the Font returned by AEGfxSetFont(...)
                                 _fontID:
                                 nextX:
                                             X coordinate
ss Core
                                             Y coordinate
                                 nextY:
                                 _GlyphHeight: Maximum Height of the glyphs in the Font
 Core()
```





```
∃void CreateLevel1()
     // Spawn a Player
     new Player();
     // Get a Random Number as Enemy Count
     const int EnemyCount = Random::GetRange(2, 25);
     // Spawn enemies up until to the chosen Count
     for(int i = 0; i< EnemyCount; ++i)</pre>
         new Enemy();
     // If i spawned more than 20 enemies, I spawn a boss
     if (EnemyCount >= 20)
         new Boss();
```

#### Useless, redundant comments

```
invoid CreateLevel1()

{
    new Player();

    const int EnemyCount = Random::GetRange(2, 25);

    for(int i = 0; i < EnemyCount; ++i)
        new Enemy();

    if (EnemyCount >= 20)
        new Boss();
}
```

Totally understandable, If names have
 been chosen properly

```
Create the first Level of the game.
  \brief
            1 Player and a Random Count of up to 25 enemies.
            Also spawn a boss if there is more than 20 enemies
void CreateLevel1()
    new Player();
    const int EnemyCount = Random::GetRange(2, 25);
    for(int i = 0; i< EnemyCount; ++i)</pre>
        new Enemy();
    if (EnemyCount >= 20)
        new Boss():
```

If you have to comment it, combine it all,and explain the function

```
CreateLevel1();

// Game L
while (co
Random Count of up to 25 enemies. Also spawn a boss if
there is more than 20 enemies
```

#### So Users can get info





### Thanks!

Any questions?

