

입력값을 1234567890 으로 넣고.. 대충 프로그램 실행중에 중간에 멈출만한 부분을 bp 걸어보자.

004010AF	> 8305 5C224000	ADD DWORD PTR [40225C],1	
004010B6	. 8305 5D224000	ADD DWORD PTR [40225D],1	
004010BD	. 8305 5E224000	ADD DWORD PTR [40225E],1	
004010C4	. 8305 5F224000	ADD DWORD PTR [40225F],1	
004010CB	. FECA	DEC DL	
004010CD	. 75 E0	JNZ SHORT 07.004010AF	
004010CF	. 68 FD234000	PUSH 07.004023FD	StringToAdd = "L2C-5781"
004010D4	. 68 00204000	PUSH 07.00402000	ConcatString = ""
004010D9	. E8 63000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010DE	. 68 5C224000	PUSH 07.0040225C	StringToAdd = ""
004010E3	. 68 00204000	PUSH 07.00402000	ConcatString = ""
004010E8	. E8 54000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010ED	. 68 24234000	PUSH 07.00402324	String2 = ""
004010F2	. 68 00204000	PUSH 07.00402000	String1 = ""
004010F7	. E8 51000000	CALL <JMP.&KERNEL32.lstrcmpiA>	lstrcmpiA
004010FC	. 83F8 00	CMP EAX,0	
004010FF	. 74 16	JE SHORT 07.00401117	
00401101	. 6A 00	PUSH 0	Style = MB_OKIMB_APPLMODAL
00401103	. 68 34244000	PUSH 07.00402434	Title = "Error!"
00401108	. 68 3B244000	PUSH 07.0040243B	Text = "The serial you entered is not cor
0040110D	. FF75 08	PUSH [ARG.1]	hOwner
00401110	. E8 56000000	CALL <JMP.&USER32.MessageBoxA>	MessageBoxA
00401115	. EB 16	JMP SHORT 07.0040112D	
00401117	> 6A 00	PUSH 0	Style = MB_OKIMB_APPLMODAL
00401119	. 68 06244000	PUSH 07.00402406	Title = "Well Done!"
0040111E	. 68 11244000	PUSH 07.00402411	Text = "Yep, you entered a correct serial
00401123	. FF75 08	PUSH [ARG.1]	hOwner
00401126	. E8 40000000	CALL <JMP.&USER32.MessageBoxA>	MessageBoxA

필자는 0040106C, 004010CF, 00401117 3개 걸었다. (그림에는 2개 bp만 나왔다)

004010A3	. 68 5C224000	PUSH 07.0040225C	ConcatString = "6784-ABEX"
004010A8	. E8 94000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010AD	. B2 02	MOV DL,2	
004010AF	> 8305 5C224000	ADD DWORD PTR [40225C],1	
004010B6	. 8305 5D224000	ADD DWORD PTR [40225D],1	
004010BD	. 8305 5E224000	ADD DWORD PTR [40225E],1	
004010C4	. 8305 5F224000	ADD DWORD PTR [40225F],1	
004010CB	. FECA	DEC DL	
004010CD	. 75 E0	JNZ SHORT 07.004010AF	
004010CF	. 68 FD234000	PUSH 07.004023FD	StringToAdd = "L2C-5781"
004010D4	. 68 00204000	PUSH 07.00402000	ConcatString = "L2C-57816784-ABEX"
004010D9	. E8 63000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010DE	. 68 5C224000	PUSH 07.0040225C	StringToAdd = "6784-ABEX"
004010E3	. 68 00204000	PUSH 07.00402000	ConcatString = "L2C-57816784-ABEX"
004010E8	. E8 54000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010ED	. 68 24234000	PUSH 07.00402324	String2 = "1234567890"
004010F2	. 68 00204000	PUSH 07.00402000	String1 = "L2C-57816784-ABEX"
004010F7	. E8 51000000	CALL <JMP.&KERNEL32.lstrcmpiA>	lstrcmpiA
004010FC	. 83F8 00	CMP EAX,0	
004010FF	. 74 16	JE SHORT 07.00401117	
00401101	. 6A 00	PUSH 0	Style = MB_OKIMB_APPLMODAL
00401103	. 68 34244000	PUSH 07.00402434	Title = "Error!"
00401108	. 68 3B244000	PUSH 07.0040243B	Text = "The serial you entered is not cor
0040110D	. FF75 08	PUSH [ARG.1]	hOwner
00401110	. E8 56000000	CALL <JMP.&USER32.MessageBoxA>	MessageBoxA
00401115	. EB 16	JMP SHORT 07.0040112D	
00401117	> 6A 00	PUSH 0	Style = MB_OKIMB_APPLMODAL
00401119	. 68 06244000	PUSH 07.00402406	Title = "Well Done!"
0040111E	. 68 11244000	PUSH 07.00402411	Text = "Yep, you entered a correct serial
00401123	. FF75 08	PUSH [ARG.1]	hOwner
00401126	. E8 40000000	CALL <JMP.&USER32.MessageBoxA>	MessageBoxA

보면 004010AF부터 004010CB까지 ADD와 DEC를 하면서 입력한 문자열값이 바뀌는데.. 결국 최종적으로 L2C-57816784-ABEX 이러한 문자열로 바뀌고 확인해보니 맞았다.

근데.. “컴퓨터 C 드라이브의 이름이 CodeEngn 일경우 시리얼이 생성될때 CodeEngn은 "어떤것"으로 변경되는가" 라고 하는걸 보니 문제가 드라이브 명을 따오는거같다;

0040107D	. 5A 00	PUSH 0	pFileSystemNameSize = NULL
0040107F	. 6A 00	PUSH 0	pFileSystemNameBuffer = NULL
00401081	. 68 C8204000	PUSH 07.004020C8	pFileSystemFlags = 07.004020C8
00401086	. 68 90214000	PUSH 07.00402190	pMaxFilenameLength = 07.00402190
0040108B	. 68 94214000	PUSH 07.00402194	pVolumeSerialNumber = 07.00402194
00401090	. 6A 32	PUSH 32	MaxVolumeNameSize = 32 (50.)
00401092	. 68 5C224000	PUSH 07.0040225C	VolumeNameBuffer = 07.0040225C
00401097	. 6A 00	PUSH 0	RootPathName = NULL
00401099	. E8 B5000000	CALL <JMP.&KERNEL32.GetVolumeInformationA>	GetVolumeInformationA
0040109E	. 68 F3234000	PUSH 07.004023F3	StringToAdd = "4562-ABEX"
004010A3	. 68 5C224000	PUSH 07.0040225C	ConcatString = ""
004010A8	. E8 94000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010AD	B2 02	MOV DL,2	
004010AF	> 8305 5C224000	ADD DWORD PTR [40225C],1	

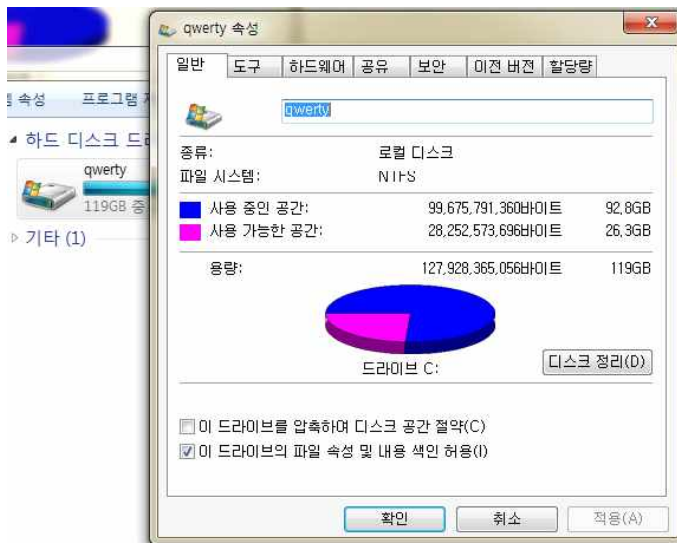
다시 한번 살펴보니, GetVolumeInformationA라는 함수가 있는데 아마 이게 드라이브 정보를 따오는 함수로 추측된다. 여기로 step into로 들어가보자.

75816E18	FF75 E0	PUSH DWORD PTR [EBP-20]	
75816E1B	FF75 C0	PUSH DWORD PTR [EBP-40]	
75816E1E	E8 0A5AFFFF	CALL <JMP.&API-MS-Win-Core-File-L1-1-0.00000000>	
75816E23	8945 E4	MOV [EBP-1C],EAX	
75816E26	3BC6	CMP EAX,ESI	
75816E28	✓ 74 12	JE SHORT kernel32.75816E3C	
75816E2A	3975 0C	CMP [EBP+C],ESI	
75816E2D	✓ 0F85 99D30000	JNZ kernel32.758241CC	
75816E33	3975 20	CMP [EBP+20],ESI	
75816E36	✓ 0F85 7A210000	JNZ kernel32.75818FB6	
75816E3C	C745 FC FEFFFFFF	MOV DWORD PTR [EBP-4],-2	
75816E43	E8 31000000	CALL kernel32.75816E79	
75816E48	8B45 E4	MOV EAX,[EBP-1C]	
75816E4B	E8 B1A7FDFF	CALL kernel32.757F1601	
75816E50	C2 2000	RETN 20	
75816E53	90	NOP	
75816E54	90	NOP	
75816E55	90	NOP	
75816E56	90	NOP	
75816E57	90	NOP	
75816E58	FE	???	Unknow
75816E59	FFFF	???	Unknow
75816E5B	FF00	INC DWORD PTR [EAX]	
75816E5D	0000	ADD [EAX],1	
7580C82D=<JMP.&API-MS-Win-Core-File-L1-1-0.00000000>			

음 뭔가 또 나오긴했다.. ?

76BD6F83	FF15 9410BC7B	CALL <C:\ntdll.RtlSetLastWinSZEError>	ntdll.RtlSetLastWinSZEError
76BD6F89	33C0	XOR EAX,EAX	
76BD6F8B	✓ E9 12010000	JMP KERNELBA.76BD70A2	
76BD6F90	0FB74D EC	MOVZX ECX,WORD PTR [EBP-14]	
76BD6F94	8B45 F0	MOV EAX,[EBP-10]	
76BD6F97	D1E9	SHR ECX,1	
76BD6F99	66:837C48 FE 5C	CMP WORD PTR [EAX+ECX*2-2],5C	
76BD6F9F	8945 FC	MOV [EBP-4],EAX	C:\ 라고 문자열이 보인다
76BD6FA2	✓ 74 20	JE SHORT KERNELBA.76BD6FC4	
76BD6FA4	50	PUSH EAX	
76BD6FA5	64:A1 18000000	MOV EAX,FS:[18]	
76BD6FAB	8B40 30	MOV EAX,[EAX+30]	
76BD6FAD	57	PUSH EAX	
EAX=00257878, (UNICODE "\??\C:\")			
Stack SS:10018F8D41=770CE785 (ntdll.770CE785)			
Address	Value	ASCI Comment	001259A0 00000032

보면 EAX값에 C:\W라고 대충 문자열이 들어감을 알 수 있다.



그런데 리버싱하던중 생각해보니 기본으로 c:\w\으로 들어가는게 맞나 싶어서 확인해보니
아예 이름이 없었따? ㄷ.ㄷ
default값이 잇겠지만 일단 테스트해보기 위해서 임의로 qwerty라고 넣어서 다시해보자!..

```

00401075 | . FF75 08      PUSH [ARG.1]
00401078 | . E8 F4000000  CALL <JMP.&USER32.GetDlgItemTextA>
0040107D | . 6A 00        PUSH 0
0040107F | . 6A 00        PUSH 0
00401081 | . 68 C8204000  PUSH 07.004020C8
00401086 | . 68 90214000  PUSH 07.00402190
0040108B | . 68 94214000  PUSH 07.00402194
00401090 | . 6A 32        PUSH 32
00401092 | . 68 5C224000  PUSH 07.0040225C
00401097 | . 6A 00        PUSH 0
00401099 | . E8 B5000000  CALL <JMP.&KERNEL32.GetVolumeInformationA>
0040109E | . 68 F3234000  PUSH 07.004023F3
004010A3 | . 68 5C224000  PUSH 07.0040225C
004010A8 | . E8 94000000  CALL <JMP.&KERNEL32.lstrcatA>
004010AD | . B2 02        MOV DL,2
004010AF | . 8305 5C224000  ADD DWORD PTR [40225C],1
004010B6 | . 8305 5D224000  ADD DWORD PTR [40225D],1
004010BD | . 8305 5E224000  ADD DWORD PTR [40225E],1
004010C4 | . 8305 5F224000  ADD DWORD PTR [40225F],1

```

```

hWnd
GetDlgItemTextA
pFileSystemNameSize = NULL
pFileSystemNameBuffer = NULL
pFileSystemFlags = 07.004020C8
pMaxFilenameLength = 07.00402190
pVolumeSerialNumber = 07.00402194
MaxVolumeNameSize = 32 (50)
VolumeNameBuffer = 07.0040225C
RootPathName = NULL
GetVolumeInformationA
StringToAdd = "4562-ABEX"
ConcatString = "qwerty"
lstrcatA

```

아 혹시나 Concat에 문자열붙어서 넣을까 싶어서 일단 분석안해보고 f8로 step over해보니
 qwerty로 그냥 문자열 붙여서 들어간다 ..—; 그래도 공부할겸 분석 한번 해보자!
 보면 일단 0040225C에 문자열을 붙이는걸 알수있다.

CPU - main thread, module kernel32			
75816D9B	6A 34	PUSH 34	
75816D9D	68 586E8175	PUSH kernel32.75816E58	
75816DA2	E8 15A8FDFF	CALL kernel32.757F15BC	
75816DA7	33F6	XOR ESI,ESI	
75816DA9	3975 08	CMP [EBP+8],ESI	
75816DAC	0F84 CBD30000	JE kernel32.7582417D	
75816DB2	FF75 08	PUSH DWORD PTR [EBP+8]	
75816DB5	8D45 BC	LEA EAX,[EBP-44]	
75816DB8	50	PUSH EAX	
75816DB9	E8 74DBFDFF	CALL kernel32.Basep8BitStringToDynamicUr	
75816DBE	85C0	TEST EAX,EAX	
75816DC0	0F84 85000000	JE kernel32.75816E4B	
75816DC6	8975 E0	MOV [EBP-20],ESI	
75816DC9	8975 D8	MOV [EBP-28],ESI	
75816DCC	33C0	XOR EAX,EAX	
75816DCE	66:8945 DE	MOV [EBP-22],AX	
75816DD2	66:8945 D6	MOV [EBP-2A],AX	
75816DD6	8B4D 0C	MOV ECX,[EBP+C]	
75816DD9	894D C8	MOV [EBP-38],ECX	
75816DDC	8B7D 10	MOV EDI,[EBP+10]	
75816DDF	8D47 01	LEA EAX,[EDI+1]	
75816DE2	66:8945 C6	MOV [EBP-3A],AX	
75816DE6	8B55 20	MOV EDX,[EBP+20]	
75816DE9	8955 D0	MOV [EBP-30],EDX	
75816DEC	8B5D 24	MOV EBX,[EBP+24]	
75816DEF	8D53 01	LEA EDX,[EBX+1]	
75816DF2	66:8955 CE	MOV [EBP-32],DX	
75816DF6	8975 FC	MOV [EBP-4],ESI	
75816DF9	3BCF	CMP ECX,ESI	
75816DFB	0F85 88D30000	JNZ kernel32.75824189	
75816E01	3975 20	CMP [EBP+20],ESI	
75816E04	0F85 6D210000	JNZ kernel32.75818F77	

Address	Value	ASCII	Comment
00402230	00000000	
00402234	00000000	
00402238	00000000	
0040223C	00000000	
00402240	00000000	
00402244	00000000	
00402248	00000000	
0040224C	00000000	
00402250	00000000	
00402254	00000000	
00402258	00000000	
0040225C	00000000	
00402260	00000000	
00402264	00000000	
00402268	00000000	
0040226C	00000000	

일단 0040225C 기준으로 값이 들어가고 나오는걸 체크하게 메모리 값을 확인하고
 위에 75816D98은 GetVolumnInformation을 타서 들어가자마자 찍은 주소다.

758241C4	FF15 E0057F75	CALL [<ntdll.RtlSetLastWin32Error>]	ntdll.RtlSetLastWin32Error
758241CA	EB 2E	JMP SHORT kernel32.758241FA	
758241CC	FF75 E0	PUSH DWORD PTR [EBP-20]	
758241CF	8D45 DC	LEA EAX,[EBP-24]	
758241D2	50	PUSH EAX	
758241D3	FF15 94057F75	CALL [<ntdll.RtlInitUnicodeString>]	ntdll.RtlInitUnicodeString
758241D9	E8 ECD5FCFF	CALL <JMP.&KERNELBASE.KernelBaseGetGlobalFlag>	
758241DE	8B40 1C	MOV EAX,[EAX+1C]	
758241E1	56	PUSH ESI	
758241E2	8D4D DC	LEA ECX,[EBP-24]	
758241E5	51	PUSH ECX	
758241E6	8D4D C4	LEA ECX,[EBP-3C]	
758241E9	51	PUSH ECX	
758241EA	FFD0	CALL EAX	
758241EC	3BC6	CMP EAX,ESI	
758241EE	0F8D 3F2CFFFF	JGE kernel32.75816E33	
758241F4	50	PUSH EAX	
758241F5	E8 3AD4FCFF	CALL kernel32.BaseSetLastNTErr	
758241FA	8975 E4	MOV [EBP-1C],ESI	
758241FD	E9 3A2CFFFF	JMP kernel32.75816E3C	
75824202	33F6	XOR ESI,ESI	
75824204	E9 702CFFFF	JMP kernel32.75816E79	
75824209	64:A1 18000000	MOV EAX,FS:[18]	
7582420F	FF75 E0	PUSH DWORD PTR [EBP-20]	
75824212	56	PUSH ESI	
75824213	8B40 30	MOV EAX,[EAX+30]	
75824216	FF70 18	PUSH DWORD PTR [EAX+18]	
75824219	FF15 00067F75	CALL [<ntdll.RtlFreeHeap>]	ntdll.RtlFreeHeap
7582421F	E9 5E2CFFFF	JMP kernel32.75816E82	
75824224	68 170000C0	PUSH C0000017	
75824229	E8 06D4FCFF	CALL kernel32.BaseSetLastNTErr	
7582422F	22C0	POP EAX,EAX	
Stack SS:[0018F934]=002D9200, (UNICODE "qwerty")			

보면 EBP기준으로 20byte만큼 빼서 qwerty라는 값을 넣는것을 확인할 수 있다.

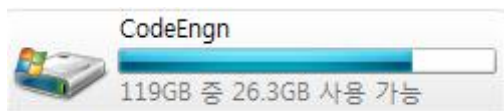
이곳을 찾게된 경로는 757816E1E라고 MS-Win-Core ??? <GetVolumeInformation>에서 몇 번 넘어가다보면 보인다.

770D6B67	0FB70B	MOVZX EAX,WORD PTR [ESI]	
770D6B6A	50	PUSH EAX	
770D6B6B	FF76 04	PUSH DWORD PTR [ESI+4]	
770D6B6E	E8 EBFDFFFF	CALL ntdll.RtlUnicodeToMultiByteN	
770D6B73	8BF8	MOV EDI,EAX	
770D6B75	897D E0	MOV [EBP-20],EDI	
770D6B78	3BF8	CMP EDI,EBX	
770D6B7A	7C 09	JL SHORT ntdll.770D6B85	
770D6B7C	8B46 04	MOV EAX,[ESI+4]	
770D6B7F	8B4D 0C	MOV ECX,[EBP+C]	
770D6B82	881C01	MOV [ECX+EAX],BL	
770D6B85	C745 FC FFFFFFFF	MOV DWORD PTR [EBP-4],-2	
770D6B8C	C745 DC 00000000	MOV DWORD PTR [EBP-24],0	
770D6B93	E8 16000000	CALL ntdll.770D6BAE	
770D6B98	3BF8	CMP EDI,EBX	
770D6B9A	7C 03	JL SHORT ntdll.770D6B9F	
770D6B9C	8B7D E4	MOV EDI,[EBP-1C]	
770D6B9F	8BC7	MOV EAX,EDI	
770D6BA1	E8 8373FFFF	CALL ntdll.770CDF29	
770D6BA6	C2 0C00	RETN 0C	
770D6BA9	90	NOP	
770D6BAA	90	NOP	
770D6BAB	90	NOP	
770D6BAC	90	NOP	
770D6BAD	90	NOP	
770D695E=ntdll.RtlUnicodeToMultiByteN			

Address	Value	ASCII	Comment
00402228	00000000	
0040222C	00000000	
00402230	00000000	
00402234	00000000	
00402238	00000000	
0040223C	00000000	
00402240	00000000	
00402244	00000000	
00402248	00000000	
0040224C	00000000	
00402250	00000000	
00402254	00000000	
00402258	00000000	
0040225C	72657771	qwer	
00402260	00007974	ty..	
00402264	00000000	

음 CALL인지 어느 주소에서 정확히 0040225C에 qwerty라 저장되는지 모르겠는데 여튼

이 부근이다. 대충 분석했으니 넘어가자;



드라이브명을 CodeEngn으로 바꾸고 해보면..

004010CB	. FEA	DEC DL	
004010CD	. ^75 E0	JNZ SHORT 07.004010AF	
004010CE	. 68 FD234000	PUSH 07.004023FD	StringToAdd = "L2C-5781"
004010D4	. 68 00204000	PUSH 07.00402000	ConcatString = "L2C-5781EqfgEngn4562-ABEX"
004010D9	. E8 63000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010DE	. 68 5C224000	PUSH 07.0040225C	StringToAdd = "EqfgEngn4562-ABEX"
004010E3	. 68 00204000	PUSH 07.00402000	ConcatString = "L2C-5781EqfgEngn4562-ABEX"
004010E8	. E8 54000000	CALL <JMP.&KERNEL32.lstrcatA>	lstrcatA
004010ED	. 68 24234000	PUSH 07.00402324	String2 = "01234567890"
004010F2	. 68 00204000	PUSH 07.00402000	String1 = "L2C-5781EqfgEngn4562-ABEX"
004010F7	. E8 51000000	CALL <JMP.&KERNEL32.lstrcmpiA>	lstrcmpiA
004010FC	. 83F8 00	CMP EAX,0	
004010FF	. 74 16	JE SHORT 07.00401117	
00401101	. 6A 00	PUSH 0	Style = MB_OKIMB_APPLMODAL
00401103	. 68 34244000	PUSH 07.00402434	Title = "Error!"
00401108	. 68 3B244000	PUSH 07.0040243B	Text = "The serial you entered is not cor
0040110D	. FF75 08	PUSH [ARG.1]	hOwner
00401110	. E8 56000000	CALL <JMP.&USER32.MessageBoxA>	MessageBoxA
00401115	. EB 16	JMP SHORT 07.0040112D	
00401117	. 6A 00	PUSH 0	Style = MB_OKIMB_APPLMODAL
00401119	. 68 06244000	PUSH 07.00402406	Title = "Well Done!"
0040111E	. 68 11244000	PUSH 07.00402411	Text = "Yep, you entered a correct serial
00401123	. FF75 08	PUSH [ARG.1]	hOwner
00401126	. E8 40000000	CALL <JMP.&USER32.MessageBoxA>	MessageBoxA
0040112B	. EB 00	JMP SHORT 07.0040112D	
0040112D	. 6A 00	PUSH 0	Result = 0
0040112F	. FF75 08	PUSH [ARG.1]	hWnd
00401132	. E8 22000000	CALL <JMP.&USER32.EndDialog>	EndDialog
00401137	. C9	LEAVE	
00401138	. C2 1000	RETN 10	
0040113B	. FF25 6C304000	JMP [<&KERNEL32.GetModuleHandleA>]	kernel32.GetModuleHandleA
0040113D	. FF25 70304000	JMP [<&KERNEL32.lstrcatA>]	kernel32.lstrcatA
EAX=00000001			

보면 CMP에서 부면 EAX값하고 0 비교하는거보니 틀렸다고 인증될텐데,

위에 004010F2주소에 string1을 보면 L2C-5781EqfgEngn4562-ABEX이걸 입력해야 정
답인것을 알수 있으니 그냥 넘어가자 ㅋ.

인증해보니 틀렸다고 해서 머지했는데..

컴퓨터 C 드라이브의 이름이 CodeEngn 일경우 시리얼이 생성될때 CodeEngn은 "어떤것"
으로 변경되는가

문제좀 잘읽자 zz..

EgfgEngn 이거겠네.