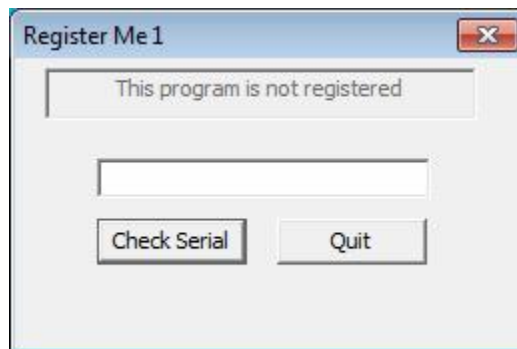


سری آموزش های R4NDOM-فارسی

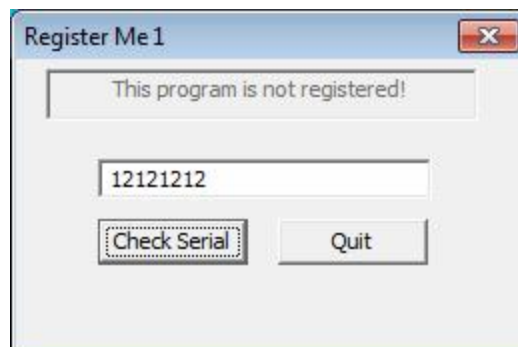
قسمت ۱۲-یک مثال

سلام به قسمت دوازدهم از سری آموزش های R4NDOM خوش آمدید، در این آموزش ما بروی یک برنامه با نام ReverseM1 کار میکنیم (ضمیمه شده) که برای ما جدید خواهد بود و دانش ما را به چالش خواهد کشید. در ضمن ما به پلاگین "Ascii Table" هم احتیاج پیدا خواهیم کرد

طبق معمول برنامه را اجرا میکنیم که یک آگاهی از محدودیت های آن بدست آوریم :



همینطور که میبینید دو نوع محدودیت وجود دارد، یکی متن not registered و دیگری سریال نامبر. بسیار خوب یک سریال دلفواه وارد کنید و کلید Check Serial را بفشارید :



همانطور که انتظار داشتیم پیام خطا ☺ برنامه را درون Olly باز کنید و به دنبال رشته های موجود بگردید:

R Found strings are		
Address	Disassembly	Text string
00401000	PUSH 0	(Initial CPU selection)
004010AC	PUSH Register.00403000	ASCII "Register me 1"
004010B1	PUSH Register.00403023	ASCII "That is correct!"
004010BD	PUSH Register.00403034	ASCII "This program is registered!"
004010D5	PUSH Register.00403000	ASCII "Register me 1"
004010DA	PUSH Register.0040300E	ASCII "That is not correct."
004010E6	PUSH Register.00403050	ASCII "This program is not registered!"
00401101	PUSH Register.00403098	ASCII "12121212"
00401116	MOV EAX, Register.00403098	ASCII "12121212"

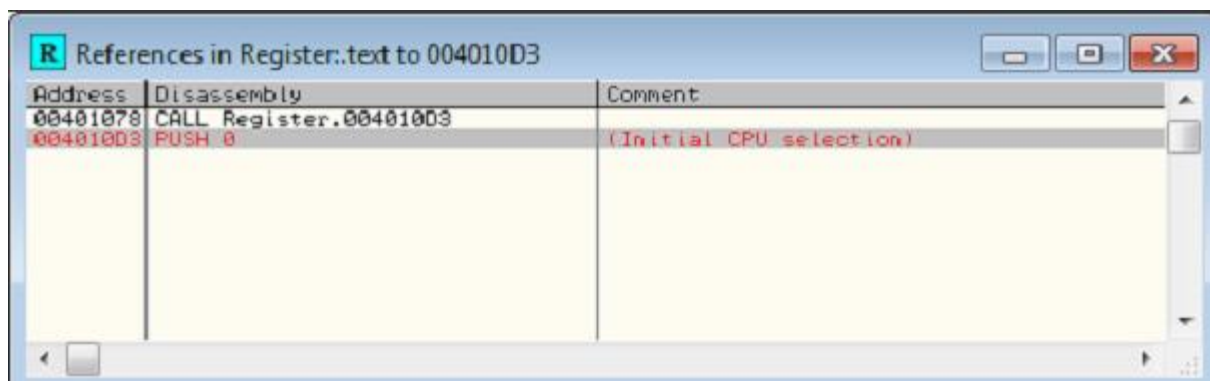
بسیار خوب به روی متن "That is not correct" دبل کلیک کنید :

004010A6	C9	LEAVE	Register.00403000	Style = MB_OK;MB_APPLMODAL
004010A7	C2 1000	RETN 10	Register.00403000	Title = "Register me 1"
004010AA	6A 00	PUSH 0	Register.00403023	Text = "That is correct!"
004010AC	68 00304000	PUSH Register.00403000	Register.00403023	hOwner = NULL
004010B1	68 23304000	PUSH Register.00403023	Register.00403023	MessageBoxA
004010B6	6A 00	PUSH 0	Register.00403034	Text = "This program is registered"
004010B8	E8 C9000000	CALL <JMP.<user32.MessageBoxA>	Register.00403034	ControlID = 3ED (1005.)
004010BD	68 34304000	PUSH Register.00403034	Register.00403034	hWnd = 00030368 ('Register Me 1',class='#32770')
004010C2	68 ED000000	PUSH 3ED	Register.00403034	SetDlgItemTextA
004010C7	FF35 94304000	PUSH DWORD PTR DS:[403094]	Register.00403034	Style = MB_OK;MB_APPLMODAL
004010CD	E8 BA000000	CALL <JMP.<user32.SetDlgItemTextA>	Register.00403034	Title = "Register me 1"
004010D2	C3	RETN 0	Register.00403034	Text = "That is not correct."
004010D3	6A 00	PUSH 0	Register.00403034	hOwner = NULL
004010D5	68 00304000	PUSH Register.00403000	Register.00403034	MessageBoxA
004010DA	68 0E304000	PUSH Register.0040300E	Register.00403034	Text = "This program is not registered"
004010DF	6A 00	PUSH 0	Register.00403050	ControlID = 3ED (1005.)
004010E1	E8 A0000000	CALL <JMP.<user32.MessageBoxA>	Register.00403050	hWnd = 00030368 ('Register Me 1',class='#32770')
004010E6	68 50304000	PUSH Register.00403050	Register.00403050	SetDlgItemTextA
004010EB	68 ED000000	PUSH 3ED	Register.00403050	Count = 200 (512.)
004010F0	FF35 94304000	PUSH DWORD PTR DS:[403094]	Register.00403098	Buffer = Register.00403098
004010F6	E8 91000000	CALL <JMP.<user32.SetDlgItemTextA>	Register.00403098	ControlID = 3E9 (1001.)
004010FB	C3	RETN 0	Register.00403098	hWnd = 00030368 ('Register Me 1',class='#32770')
004010FC	68 00020000	PUSH 200	Register.00403098	SetDlgItemTextA
00401101	68 90304000	PUSH Register.00403098	Register.00403098	ASCII "12121212"
00401106	68 E9000000	PUSH 3E9	Register.00403098	
0040110B	FF35 94304000	PUSH DWORD PTR DS:[403094]	Register.00403098	
00401111	E8 6A000000	CALL <JMP.<user32.SetDlgItemTextA>	Register.00403098	
00401116	B8 90304000	MOV EAX,Register.00403098	Register.00403098	
0040111B	8B00	MOV EAX,DWORD PTR DS:[EAX]	Register.00403098	
00401120	66:3D 3433	CMP AX,3334	Register.00403098	
00401121	7C 07	JNZ SHORT Register.00401120	Register.00403098	

همانطور که میبینید تمامی روتین ها از هم جدا هستند، طبق آموزش های قبل باید (ارجاع یا ارجاعات مربوط به این پیام را بررسی کنیم،برای اینکار بروی دستور فوق راست کلیک کرده :

004010C7	FF35 94304000	PUSH DWORD PTR DS:[403094]	hWnd = 00030368 ('Register Me 1',class='#32770')	ST0 empty 0.0
004010CD	E8 BA000000	CALL <JMP.<user32.SetDlgItemTextA>	SetDlgItemTextA	ST1 empty 0.0
004010D2	C3	RETN 0	SetDlgItemTextA	ST2 empty 0.0
004010D3	6A 00	PUSH 0	SetDlgItemTextA	ST3 empty 0.0
004010D5	68 00304000	PUSH Register.00403000	SetDlgItemTextA	ST4 empty 0.0
004010DA	68 0E304000	PUSH Register.0040300E	SetDlgItemTextA	ST5 empty 0.0
004010DF	6A 00	PUSH 0	SetDlgItemTextA	ST6 empty 0.0
004010E1	E8 A0000000	CALL <JMP.<user32.MessageBoxA>	SetDlgItemTextA	ST7 empty 0.0
004010E6	68 50304000	PUSH Register.00403050	SetDlgItemTextA	FST 4020 Cond 1 0
004010EB	68 ED000000	PUSH 3ED	SetDlgItemTextA	FCW 027F Prec NEAR
004010F0	FF35 94304000	PUSH DWORD PTR DS:[403094]	SetDlgItemTextA	
004010F6	E8 91000000	CALL <JMP.<user32.SetDlgItemTextA>	SetDlgItemTextA	
004010FB	C3	RETN 0	SetDlgItemTextA	
004010FC	68 00020000	PUSH 200	SetDlgItemTextA	
00401101	68 90304000	PUSH Register.00403098	SetDlgItemTextA	
00401106	68 E9000000	PUSH 3E9	SetDlgItemTextA	
0040110B	FF35 94304000	PUSH DWORD PTR DS:[403094]	SetDlgItemTextA	
00401111	E8 6A000000	CALL <JMP.<user32.SetDlgItemTextA>	SetDlgItemTextA	
00401116	B8 90304000	MOV EAX,Register.00403098	SetDlgItemTextA	
0040111B	8B00	MOV EAX,DWORD PTR DS:[EAX]	SetDlgItemTextA	
00401120	66:3D 3433	CMP AX,3334	SetDlgItemTextA	
00401121	7C 07	JNZ SHORT Register.00401120	SetDlgItemTextA	
00401123	B8 00000000	MOV EAX,0	SetDlgItemTextA	
00401128	E8 05	JMP SHORT Register.0040112F	SetDlgItemTextA	
0040112A	B8 01000000	MOV EAX,1	SetDlgItemTextA	
0040112F	C3	RETN 0	SetDlgItemTextA	
00401130	E8 C7FFFFFF	CALL Register.004010FC	SetDlgItemTextA	
00401135	00C0	OR EAX,EAX	SetDlgItemTextA	
00401137	74 33	JE SHORT Register.0040116C	SetDlgItemTextA	
00401139	B9 1F000000	MOV ECX,1F	SetDlgItemTextA	
0040113E	BE 00000000	MOV ESI,0	SetDlgItemTextA	
00401143	3300	XOR EAX,EAX	SetDlgItemTextA	
00401145	8B86 70304000	MOV AL,BYTE PTR DS:[ESI+403070]	SetDlgItemTextA	
0040114E	83F0 2C	XOR EAX,2C	SetDlgItemTextA	
0040114E	8B86 70304000	MOV BYTE PTR DS:[ESI+403070],AL	SetDlgItemTextA	
00401154	46	INC ESI	SetDlgItemTextA	
00401155	E2 EE	LOOPD SHORT Register.00401145	SetDlgItemTextA	
00401157	68 70304000	PUSH Register.00403070	SetDlgItemTextA	
0040115C	68 ED000000	PUSH 3ED	SetDlgItemTextA	
00401161	FF35 94304000	PUSH DWORD PTR DS:[403094]	SetDlgItemTextA	
00401167	E8 20000000	CALL <JMP.<user32.SetDlgItemTextA>	SetDlgItemTextA	
0040116C	C3	RETN 0	SetDlgItemTextA	
0040116D	CC	INT3	SetDlgItemTextA	
0040116E	FF25 24204000	JMP DWORD PTR DS:[<user32.DialogBoxParamA>]	SetDlgItemTextA	
00401174	FF25 18204000	JMP DWORD PTR DS:[<user32.EndDialogBox>]	SetDlgItemTextA	

پنجره ی References :



بروی فراخوانی دبل کلیک کنید تا به محل آن بروید :

00401063	E8 94000000	CALL Register.004010FC	
00401068	0BC0	OR EAX,EAX	rport4.75C81B9C
0040106A	75 0C	JNZ SHORT Register.00401078	
0040106C	E8 39000000	CALL Register.004010AA	
00401071	E8 BA000000	CALL Register.00401130	
00401076	EB 2C	JMP SHORT Register.004010A4	
00401078	E8 56000000	CALL Register.004010D3	
0040107D	EB 25	JMP SHORT Register.004010A4	
0040107F	817D 10 EC030000	CMP [ARG.3],3EC	
00401086	75 1C	JNZ SHORT Register.004010A4	
00401088	6A 00	PUSH 0	[Result = 0 hWnd = 0252006C EndDialog
0040108A	FF75 08	PUSH [ARG.1]	
0040108D	E8 E2000000	CALL <JMP.&user32.EndDialog>	
00401092	EB 10	JMP SHORT Register.004010A4	
00401094	837D 0C 10	CMP [ARG.2],10	
00401098	75 0A	JNZ SHORT Register.004010A4	
0040109A	6A 00	PUSH 0	[Result = 0 hWnd = 0252006C EndDialog rport4.75C81B9C
0040109C	FF75 08	PUSH [ARG.1]	
0040109F	E8 D0000000	CALL <JMP.&user32.EndDialog>	
004010A4	33C0	XOR EAX,EAX	
004010A6	C9	LEAVE	
004010A7	C2 1000	RETN 10	
004010AA	6A 00	PUSH 0	[Style = MB_OK MB_APPLMODAL Title = "Register me 1" Text = "That is correct!" hOwner = NULL MessageBoxA Text = "This program is registered!" ControlID = 3ED (1005.) hWnd = 00030368 ('Register Me 1',class='#32770') SetDlgItemTextA
004010AC	68 00304000	PUSH Register.00403000	
004010B1	68 23304000	PUSH Register.00403023	
004010B6	6A 00	PUSH 0	
004010B8	E8 C9000000	CALL <JMP.&user32.MessageBoxA>	
004010BD	68 34304000	PUSH Register.00403034	
004010C2	68 ED030000	PUSH 3ED	
004010C7	FF35 94304000	PUSH DWORD PTR DS:[403094]	
004010CD	E8 BA000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010D2	C3	RETN	
004010D3	6A 00	PUSH 0	[Style = MB_OK MB_APPLMODAL Title = "Register me 1" Text = "That is not correct." hOwner = NULL MessageBoxA Text = "This program is not registered!" ControlID = 3ED (1005.) hWnd = 00030368 ('Register Me 1',class='#32770') SetDlgItemTextA
004010D5	68 00304000	PUSH Register.00403000	
004010DA	68 0E304000	PUSH Register.0040300E	
004010DF	6A 00	PUSH 0	
004010E1	E8 A0000000	CALL <JMP.&user32.MessageBoxA>	
004010E6	68 50304000	PUSH Register.00403050	
004010EB	68 ED030000	PUSH 3ED	
004010F0	FF35 94304000	PUSH DWORD PTR DS:[403094]	
004010F6	E8 91000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010FB	C3	RETN	
004010FC	68 00020000	PUSH 200	[Count = 200 (512.) Buffer = Register.00403098 ControlID = 3E9 (1001.) hWnd = 00030368 ('Register Me 1',class='#32770') GetDlgItemTextA ASCI "12121212"
00401101	68 90304000	PUSH Register.00403098	
00401106	68 E9030000	PUSH 3E9	
0040110B	FF35 94304000	PUSH DWORD PTR DS:[403094]	
00401111	E8 6A000000	CALL <JMP.&user32.SetDlgItemTextA>	
00401116	68 90304000	MOV EAX,Register.00403098	
0040111B	8B00	MOV EAX,DWORD PTR DS:[EAX]	
0040111D	66:3D 3433	CMPSX 3433	
00401121	75 07	JNZ SHORT Register.0040112A	
00401123	68 00000000	MOV EAX,0	

بسیار عالی،همانطور که میبیند تابع Bad Boy در آدرس 401078 است.و بلافاصله در آدرس 40106A پرشی را میبینید که ما را به این تابع ارجاع میدهد:

00401063	E8 94000000	CALL Register.004010FC	
00401068	0BC0	OR EAX,EAX	rport4.75C81B9C
0040106A	75 0C	JNZ SHORT Register.00401078	This JNZ...
00401071	E8 39000000	CALL Register.004010AA	
00401076	E8 BA000000	CALL Register.00401130	
00401078	EB 2C	JMP SHORT Register.004010A4	
0040107D	E8 56000000	CALL Register.004010D3	
0040107F	EB 25	JMP SHORT Register.004010A4	
00401086	817D 10 EC030000	CMP [ARG.3],3EC	
00401088	75 1C	JNZ SHORT Register.004010A4	
0040108A	6A 00	PUSH 0	[Result = 0 hWnd = 0252006C EndDialog
0040108D	FF75 08	PUSH [ARG.1]	
0040108F	E8 E2000000	CALL <JMP.&user32.EndDialog>	
00401092	EB 10	JMP SHORT Register.004010A4	
00401094	837D 0C 10	CMP [ARG.2],10	
00401098	75 0A	JNZ SHORT Register.004010A4	
0040109A	6A 00	PUSH 0	
0040109C	FF75 08	PUSH [ARG.1]	
0040109F	E8 D0000000	CALL <JMP.&user32.EndDialog>	
004010A4	33C0	XOR EAX,EAX	
004010A6	C9	LEAVE	
004010A7	C2 1000	RETN 10	
004010AA	6A 00	PUSH 0	[Style = MB_OK MB_APPLMODAL Title = "Register me 1" Text = "That is correct!" hOwner = NULL MessageBoxA Text = "This program is registered!" ControlID = 3ED (1005.) hWnd = 00030368 ('Register Me 1',class='#32770') SetDlgItemTextA
004010AC	68 00304000	PUSH Register.00403000	
004010B1	68 23304000	PUSH Register.00403023	
004010B6	6A 00	PUSH 0	
004010B8	E8 C9000000	CALL <JMP.&user32.MessageBoxA>	
004010BD	68 34304000	PUSH Register.00403034	
004010C2	68 ED030000	PUSH 3ED	
004010C7	FF35 94304000	PUSH DWORD PTR DS:[403094]	
004010CD	E8 BA000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010D2	C3	RETN	
004010D3	6A 00	PUSH 0	[Style = MB_OK MB_APPLMODAL Title = "Register me 1" Text = "That is not correct." hOwner = NULL MessageBoxA Text = "This program is not registered!" ControlID = 3ED (1005.) hWnd = 00030368 ('Register Me 1',class='#32770') SetDlgItemTextA
004010D5	68 00304000	PUSH Register.00403000	
004010DA	68 0E304000	PUSH Register.0040300E	
004010DF	6A 00	PUSH 0	
004010E1	E8 A0000000	CALL <JMP.&user32.MessageBoxA>	
004010E6	68 50304000	PUSH Register.00403050	
004010EB	68 ED030000	PUSH 3ED	
004010F0	FF35 94304000	PUSH DWORD PTR DS:[403094]	
004010F6	E8 91000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010FB	C3	RETN	
004010FC	68 00020000	PUSH 200	[Count = 200 (512.) Buffer = Register.00403098 ControlID = 3E9 (1001.) hWnd = 00030368 ('Register Me 1',class='#32770') GetDlgItemTextA ASCI "12121212"
00401101	68 90304000	PUSH Register.00403098	
00401106	68 E9030000	PUSH 3E9	
0040110B	FF35 94304000	PUSH DWORD PTR DS:[403094]	
00401111	E8 6A000000	CALL <JMP.&user32.SetDlgItemTextA>	
00401116	68 90304000	MOV EAX,Register.00403098	
0040111B	8B00	MOV EAX,DWORD PTR DS:[EAX]	
0040111D	66:3D 3433	CMPSX 3433	
00401121	75 07	JNZ SHORT Register.0040112A	
00401123	68 00000000	MOV EAX,0	

اگر کمی بالا بروید در آدرس 401063 (روتین اصلی (مقایسه / پرش) را میبینید :

00401059	75 24	JNZ SHORT Register.0040107F	This is the main call..
0040105B	8B45 08	MOV EAX,[ARG.1]	
0040105E	A3 94304000	MOV DWORD PTR DS:[403094],EAX	
00401063	E8 94000000	CALL Register.004010FC	
00401068	0BC0	OR EAX,EAX	
0040106A	75 0C	JNZ SHORT Register.00401078	
0040106C	E8 39000000	CALL Register.004010AA	
00401071	E8 BA000000	CALL Register.00401130	
00401076	EB 2C	JMP SHORT Register.004010A4	
00401078	E8 56000000	CALL Register.004010D3	
0040107D	EB 25	JMP SHORT Register.004010A4	Result = 0 hWnd = 0252006C EndDialog
0040107F	817D 10 EC030000	CMP [ARG.3],3EC	
00401086	75 1C	JNZ SHORT Register.004010A4	
00401088	6A 00	PUSH 0	
0040108A	FF75 08	PUSH [ARG.1]	
0040108D	E8 E2000000	CALL <JMP.&user32.EndDialog>	
00401092	EB 10	JMP SHORT Register.004010A4	
00401094	837D 0C 10	CMP [ARG.2],10	
00401098	75 0A	JNZ SHORT Register.004010A4	
0040109A	6A 00	PUSH 0	
0040109C	FF75 08	PUSH [ARG.1]	
0040109F	E8 D0000000	CALL <JMP.&user32.EndDialog>	
004010A4	33C0	XOR EAX,EAX	
004010A6	C9	LEAVE	
004010A7	C2 1000	RETN 10	
004010AA	6A 00	PUSH 0	
004010AC	68 00304000	PUSH Register.00403000	
004010B1	68 23304000	PUSH Register.00403023	
004010B6	6A 00	PUSH 0	
004010B8	E8 C9000000	CALL <JMP.&user32.MessageBoxA>	Style = MB_OK MB_APPLMODAL Title = "Register Me 1" Text = "That is correct!" hOwner = NULL MessageBoxA Text = "This program is registered" ControlID = 3ED (1005.) hWnd = 00030368 ('Register Me 1',class='SetDlgItemTextA
004010BD	68 34304000	PUSH Register.00403034	
004010C2	68 ED030000	PUSH 3ED	
004010C7	FF35 94304000	PUSH DWORD PTR DS:[403094]	
004010CD	E8 BA000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010D2	C3	RETN	
004010D3	6A 00	PUSH 0	
004010D5	68 00304000	PUSH Register.00403000	
004010D8	68 003030E	PUSH Register.0040300E	
004010DB	6A 00	PUSH 0	
004010DE	68 50304000	PUSH Register.00403050	Style = MB_OK MB_APPLMODAL Title = "Register Me 1" Text = "That is not correct." hOwner = NULL MessageBoxA Text = "This program is not registered!" ControlID = 3ED (1005.) hWnd = 00030368 ('Register Me 1',class='SetDlgItemTextA
004010E3	68 ED030000	PUSH 3ED	
004010F0	FF35 94304000	PUSH DWORD PTR DS:[403094]	
004010F6	E8 91000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010FB	C3	RETN	
004010FC	68 00200000	PUSH 200	
00401101	68 98304000	PUSH Register.00403098	
00401106	68 E9030000	PUSH 3E9	
0040110B	FF35 94304000	PUSH DWORD PTR DS:[403094]	
00401111	E8 6A000000	CALL <JMP.&user32.GetDlgItemTextA>	
00401116	8B00	MOV EAX,Register.00403098	
00401118	MOV EAX,DWORD PTR DS:[EAX]		
00401121	66:3D 3433	CMP AX,3334	
00401123	75 07	JNZ SHORT Register.0040112A	
00401128	8B 05	MOV EAX,0	
0040112A	JMP SHORT Register.0040112F		
0040112F	MOV EAX,1		
00401130	C3	RETN	
00401133	E8 C7FFFFFF	CALL Register.004010FC	
00401135	0BC0	OR EAX,EAX	rport4.75CB1B9C
00401137	74 33	JE SHORT Register.0040116C	

محتوای این روتین در آدرس 4010FC قرار دارد و مقدار بازگشتی درون ثبات EAX ذخیره میشود بدین معنی که اگر مقدار این ثبات برابر با صفر باشد به Good Boy و اگر ۱ باشد به Bad Boy پرش خواهیم کرد، بیایید بروی آدرس 40106A یک BP بگذاریم و برنامه را ریستارت و سریال دلفواه را وارد کنیم (من 12121212 را وارد کرده ام) ، میبینیم که olly به روی آدرس فوق متوقف شده :

0040105E	A3 94304000	MOV DWORD PTR DS:[403094],EAX
00401063	E8 94000000	CALL Register.004010FC
00401068	0BC0	OR EAX,EAX
0040106A	75 0C	JNZ SHORT Register.00401078
0040106C	E8 39000000	CALL Register.004010AA
00401071	E8 BA000000	CALL Register.00401130
00401076	EB 2C	JMP SHORT Register.004010A4
00401078	E8 56000000	CALL Register.004010D3
0040107D	EB 25	JMP SHORT Register.004010A4
0040107F	817D 10 EC030000	CMP [ARG.3],3EC
00401086	75 1C	JNZ SHORT Register.004010A4
00401088	6A 00	PUSH 0
0040108A	FF75 08	PUSH [ARG.1]
0040108D	E8 E2000000	CALL <JMP.&user32.EndDialog>
00401092	EB 10	JMP SHORT Register.004010A4
00401094	837D 0C 10	CMP [ARG.2],10
00401098	75 0A	JNZ SHORT Register.004010A4
0040109A	6A 00	PUSH 0

به کمک olly مقدار پرچم Z را تغییر میدهیم :

```

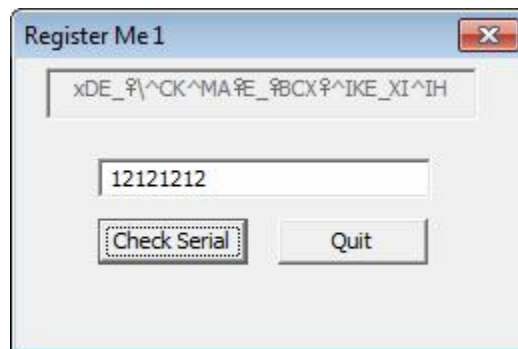
C 0  ES 002
P 0  CS 001
A 0  SS 002
Z 1  DS 002
S 0  FS 003
T 0  GS 000
D 0
O 0  LastEx

```

مال برنامه را کامل با F9 اجرا کنید:



راحت بود نه؟ حالا Ok را کلیک کنید:



!! متن داخل تکس باکس اول تغییر کرده ،یعنی چه؟(مروف عجیب و غریب) یک معنی بیشتر نمیدهد ،شاید به نظر پیام Good Boy را دریافت کرده باشیم اما در کل یک اشتباهی رخ داده ، برنامه را ری استارت کنید و یک سریال به برنامه بدهید Olly باز بروی فضا 40106A متوقف میشود :

00401068	EB 75 0C	JNZ SHORT Register.00401078
0040106A	CALL Register.004010AA	
0040106C	CALL Register.00401130	
00401071	EB 2C	JMP SHORT Register.004010A4
00401076	CALL Register.004010D3	
00401078	JMP SHORT Register.004010A4	
0040107D	817D 10 EC030000	CMP [ARG.3],3EC
0040107F	75 1C	JNZ SHORT Register.004010A4
00401086	6A 00	PUSH 0
00401088	FF75 08	PUSH [ARG.1]
0040108A	CALL <JMP.&user32.EndDialog>	
0040108D	JMP SHORT Register.004010A4	
00401092	837D 0C 10	CMPS [ARG.2],10
00401094	75 0A	JNZ SHORT Register.004010A4
00401098	6A 00	PUSH 0
0040109C	FF75 08	PUSH [ARG.1]
0040109F	CALL <JMP.&user32.EndDialog>	
004010A4	XOR EAX,EAX	
004010A6	LEAVE	
004010A7	C9	RETN 10
004010A9	6A 00	PUSH 0
004010AC	PUSH Register.00403000	
004010B1	PUSH Register.00403023	
004010B6	6A 00	PUSH 0
004010B8	CALL <JMP.&user32.MessageBoxA>	
004010BD	PUSH Register.00403034	
004010C2	PUSH 3ED	
004010C7	PUSH DWORD PTR DS:[403094]	
004010CD	CALL <JMP.&user32.SetDlgItemTextA>	
004010D2	C3	RETN

همان طور که میبینید این پرش به سمت bad boy خواهد رفت ،مال پرچم Z را تغییر دهید و کلید F7 را بفشارید:

00401068	EB 75 0C	JNZ SHORT Register.00401078	
0040106A	CALL Register.004010AA		
0040106C	CALL Register.00401130		
00401071	EB 2C	JMP SHORT Register.004010A4	
00401076	CALL Register.004010D3		
00401078	JMP SHORT Register.004010A4		
0040107D	817D 10 EC030000	CMP [ARG.3],3EC	
0040107F	75 1C	JNZ SHORT Register.004010A4	
00401086	6A 00	PUSH 0	
00401088	FF75 08	PUSH [ARG.1]	
0040108A	CALL <JMP.&user32.EndDialog>		
0040108D	JMP SHORT Register.004010A4		
00401092	837D 0C 10	CMPS [ARG.2],10	
00401094	75 0A	JNZ SHORT Register.004010A4	
00401098	6A 00	PUSH 0	
0040109C	FF75 08	PUSH [ARG.1]	
0040109F	CALL <JMP.&user32.EndDialog>		
004010A4	XOR EAX,EAX		
004010A6	LEAVE		
004010A7	C9	RETN 10	
004010A9	6A 00	PUSH 0	
004010AC	PUSH Register.00403000		
004010B1	PUSH Register.00403023		
004010B6	6A 00	PUSH 0	
004010B8	CALL <JMP.&user32.MessageBoxA>		
004010BD	PUSH Register.00403034		
004010C2	PUSH 3ED		
004010C7	PUSH DWORD PTR DS:[403094]		
004010CD	CALL <JMP.&user32.SetDlgItemTextA>		
004010D2	C3	RETN	

[Result = 0
hWnd = 000503B2 ('Register Me 1',class='#32770')
EndDialog

[Result = 0
hWnd = 000503B2 ('Register Me 1',class='#32770')
EndDialog

[Style = MB_OK|MB_APPLMODAL
Title = "Register me 1"
Text = "That is correct!"
hOwner = NULL
MessageBoxA
Text = "This program is registered!"
ControlID = 3ED (1005.)
hWnd = 000503B2 ('Register Me 1',class='#32770')
SetDlgItemTextA

بعد از فشردن کلید F7 اکنون در آدرس 4010AA هستیم با نمایش پیام "That is correct" و در پایین این پیام، متن تکس باکس اول با مضمون "This program is registered!" را میبینیم، و در خط 4010CD تابع SetDlgItemTextA را میبینیم، این تابع قادر است متن تکس باکس ما را جایگزین/ویرایش /مذف کند، و در انتها به آدرس 401071 ارجاع داده میشود :

00401068	75 0C	JNZ SHORT Register.00401078	
0040106C	E8 39000000	CALL Register.004010AA	
00401071	E8 BA000000	CALL Register.00401130	
00401076	EB 2C	JMP SHORT Register.004010A4	
00401078	E8 56000000	CALL Register.004010D3	
0040107D	EB 25	JMP SHORT Register.004010A4	
0040107F	017D 10 EC030000	CMP [ARG_3], 3EC	
00401086	75 1C	JNZ SHORT Register.004010A4	
00401088	6A 00	PUSH 0	
0040108A	FF75 08	PUSH [ARG_1]	
0040108D	E8 E2000000	CALL <JMP.&user32.EndDialog>	Result = 0 hWnd = 000503B2 ('Register Me 1',class='#32770') EndDialog
00401092	EB 10	JMP SHORT Register.004010A4	
00401094	837D 0C 10	CMP [ARG_2], 10	
00401098	75 0A	JNZ SHORT Register.004010A4	
0040109A	6A 00	PUSH 0	
0040109C	FF75 08	PUSH [ARG_1]	
0040109F	E8 D0000000	CALL <JMP.&user32.EndDialog>	Result = 0 hWnd = 000503B2 ('Register Me 1',class='#32770') EndDialog
004010A4	33C0	XOR EAX, EAX	
004010A6	C9	LEAVE	
004010A7	C2 1000	RETN 10	
004010AA	6A 00	PUSH 0	
004010AC	68 00304000	PUSH Register.00403000	
004010B1	68 23304000	PUSH Register.00403023	
004010B6	6A 00	PUSH 0	
004010B8	E8 C9000000	CALL <JMP.&user32.MessageBox>	Style = MB_OK MB_APPLMODAL Title = "Register me 1" Text = "That is correct!" hOwner = NULL MessageBoxA
004010BD	68 34304000	PUSH Register.00403034	
004010C2	68 ED030000	PUSH 3ED	
004010C7	FF35 94304000	PUSH DWORD PTR DS:[403094]	Text = "This program is registered!" ControlID = 3ED (1005.) hWnd = 000503B2 ('Register Me 1',class='#32770') SetDlgItemTextA
004010CD	E8 BA000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010D2	C3	RETN	
004010D3	6A 00	PUSH 0	
004010D5	68 00304000	PUSH Register.00403000	Style = MB_OK MB_APPLMODAL Title = "Register me 1"

حال با فشردن کلید F7 به درون تابع در آدرس 401130 میرویم :

0040112F	C3	RETN	
00401130	E8 C7FFFFFF	CALL Register.004010FC	Call to serial check
00401135	0BC0	OR EAX, EAX	
00401137	74 33	JE SHORT Register.0040116C	If EAX != 0...
00401139	B9 1F000000	MOV ECX, 1F	
0040113E	BE 00000000	MOV ESI, 0	
00401143	33C0	XOR EAX, EAX	
00401145	8A86 70304000	MOV AL, BYTE PTR DS:[ESI+403070]	
00401148	83F0 2C	XOR EAX, 2C	
0040114E	8886 70304000	MOV BYTE PTR DS:[ESI+403070], AL	
00401154	46	INC ESI	
00401155	E2 EE	LOOPE SHORT Register.00401145	
00401157	68 70304000	PUSH Register.00403070	
00401161	FF35 94304000	PUSH DWORD PTR DS:[403094]	
00401166	E8 20000000	CALL <JMP.&user32.SetDlgItemTextA>	Text = "xDE\x0C\CK\MA\x0CE\x0CBC\x0C\KE_XI^IHr" ControlID = 3ED (1005.) hWnd = 000503B2 ('Register Me 1',class='#32770') SetDlgItemTextA
0040116C	C3	RETN	

در خط اول دوباره روتین چک سریال در آدرس 4010FC انجام میگیرد ، و در خط بعد مقدار ثبات EAX چک میشود که آیا برابر صفر است یا نه ، اگر برابر صفر بود پرش صورت میگیرد و اگر مقدارش 1 بود فیر (این مقدار برگشتی تابع 4010FC است)، در خط بعد مقدار 1F درون ثبات ECX (معادل 31) قرار میگیرد، در دو خط بعدی به ترتیب ابتدا ESI=0 و بعد EAX=0 میشود، حال در خط 401145 وارد یک حلقه میشویم :

00401137	B9 1F000000	MOV ECX, 1F	
0040113E	BE 00000000	MOV ESI, 0	
00401143	33C0	XOR EAX, EAX	
00401145	8A86 70304000	MOV AL, BYTE PTR DS:[ESI+403070]	
00401148	83F0 2C	XOR EAX, 2C	
0040114E	8886 70304000	MOV BYTE PTR DS:[ESI+403070], AL	
00401154	46	INC ESI	
00401155	E2 EE	LOOPE SHORT Register.00401145	
00401157	68 70304000	PUSH Register.00403070	

در خط اول حلقه در آدرس 401145 ابتدا یک بایت از آدرس ESI + 403070 درون AL قرار میگیرد، ما از قبل میدانیم مقدار ESI برابر با صفر شده (در دو خط بالا)، پس یک بایت از آدرس 403070 درون AL قرار میگیرد، بیایید ببینیم چه چیزی در AL ذخیره میشود برای اینکار بروی خط جاری راست کلیک کنید و سپس : Follow in dump ->Memory Addres

004011B9	00	DB 00
DS:[00403070]=54 ('T')		
AL=00		
Jump from 00401155		
Address	Hex dump	ASCII
00403070	54 68 69 73 20 70 72 6F	This pro
00403078	67 72 61 6D 20 69 73 20	gram is
00403080	6E 6F 74 20 72 65 67 69	not regi
00403088	73 74 65 72 65 64 21 00	stered!.
00403090	00 00 40 00 60 03 05 00	..0.'...
00403098	31 32 31 32 31 32 31 32	12121212
004030A0	00 00 00 00 00 00 00 00	
004030A8	00 00 00 00 00 00 00 00	
004030B0	00 00 00 00 00 00 00 00	
004030B8	00 00 00 00 00 00 00 00	
004030C0	00 00 00 00 00 00 00 00	
004030C8	00 00 00 00 00 00 00 00	
004030D0	00 00 00 00 00 00 00 00	
004030D8	00 00 00 00 00 00 00 00	

بله ، ابتدا ک(راکتر T (از متن داخل تکس باکس) را درون AL انتقال داده میشود و سپس در فضا بعد کد اسکی ک(راکتر 2C) با مقدار EAX (هاوی ک(راکتر T) به صورت XOR میشود :

AL=78 ('x')
DS:[00403070]=54 ('T')

که نتیجه ک(راکتر (X) میشود و در فضا بعد در ESI+403070 نوشته میشود (جای قبلی) و در فضا بعد یک واحد به ESI اضافه میشود تا ذخیره شود و عملیات XOR تا آخرین ک(راکتر ادامه یابد در فضا بعد یک واحد از مقدار ECX کاسته میشود و این عمل تا موقعی که ECX=0 باشد انجام میشود :

```

00401155  E2 EE      LOOP SHORT Register.00401145
00401157  68 70304000 PUSH Register.00403070
0040115C  68 ED030000 PUSH 3ED
00401161  FF35 94304000 PUSH DWORD PTR DS:[403094]
00401167  E8 20000000 CALL <JMP.>user32.SetDlgItemTextA
0040116C  C3        RETN
0040116D  CC        INT3
0040116E  FF25 24204000 JMP DWORD PTR DS:[<user32.DialogBoxParamA>]
00401174  FF25 18204000 JMP DWORD PTR DS:[<user32.EndDialog>]
0040117A  FF25 10204000 JMP DWORD PTR DS:[<user32.GetDlgItem>]
00401180  FF25 20204000 JMP DWORD PTR DS:[<user32.GetDlgItemTextA>]
00401186  FF25 0C204000 JMP DWORD PTR DS:[<user32.MessageBoxA>]
0040118C  FF25 1C204000 JMP DWORD PTR DS:[<user32.SetDlgItemTextA>]
00401192  FF25 14204000 JMP DWORD PTR DS:[<user32.SetFocus>]
00401198  FF25 00204000 JMP DWORD PTR DS:[<kernel32.ExitProcess>]
0040119E  FF25 04204000 JMP DWORD PTR DS:[<kernel32.GetModuleHandleA>]
004011A4  00        DB 00
004011A5  00        DB 00
004011A6  00        DB 00
004011A7  00        DB 00
004011A8  00        DB 00
004011A9  00        DB 00
004011AA  00        DB 00
004011AB  00        DB 00
004011AC  00        DB 00

```

```

JE SHORT Register.0040116C
MOV ECX,1F
MOV ESI,0
XOR EAX,EAX
MOV AL,BYTE PTR DS:[ESI+403070]
XOR EAX,2C
MOV BYTE PTR DS:[ESI+403070],AL
INC ESI

```

```

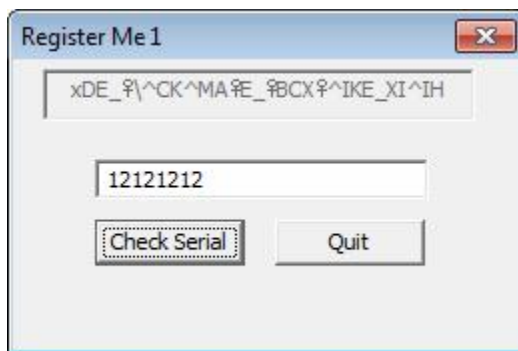
Text = "This program is BCK\K0C^IKE_XI^IH_r"
MessageBox = 3ED (2003)
hWnd = 000503B2 ("Register Me 1",class="#32770")
SetDlgItemTextA

```

Loop is taken
ECX=00000010 (decimal 16.)
00401145=Register.00401145

Address	Hex dump	ASCII
00403070	54 68 69 73 20 70 72 6F	This program is
00403080	67 72 61 6D 20 69 73 20	not registered!
00403090	00 00 40 00 60 03 05 00	..0.'...
004030A0	31 32 31 32 31 32 31 32	12121212
004030B0	00 00 00 00 00 00 00 00	
004030C0	00 00 00 00 00 00 00 00	

میکرد:

[illegible]

حال بیا بید به روتین یک سریال نگاه کنید در ابتدا تابع GetDlgItemTextA فراخوانی شده که مسئول دریافت سریال است :

The screenshot shows a debugger window with assembly code on the left and a comment box on the right. The assembly code is as follows:

```

004010FC 68 00020000 PUSH 200
00401101 68 98040000 PUSH Register.00400308
00401106 68 E9030000 PUSH 3E9
0040110B FF35 94304000 PUSH DWORD PTR DS:[403094]
00401111 E8 6A000000 CALL JUMP.User32.GetDlgItemTextA
00401116 B8 98040000 MOV EAX,Register.00400308
0040111B 8B00 MOV EAX,DWORD PTR DS:[EAX]
00401120 66:3D 3433 CMF AX,3334
00401121 7D 07 JNZ SHORT Register.0040112A
00401123 B8 00000000 MOV EAX,0
00401128 EB 05 JMP SHORT Register.0040112F
0040112D C9 01000000 MOV EAX,1
0040112F RETN

```

The comment box on the right contains the following text:

```

Count = 200 (512.)
Buffer = Register.00400308
ControlID = 3E9 (1001.)
hwnd = 003F0410 ('Register Me 1',class='#32770')
GetDlgItemTextA

```

A red arrow points from the comment box to a blue box labeled "Get entered serial".

بروی آدرس 401101 راست کلیک کنید (DUMP):

[illegible]

همانطور که میبینید سریال وارد شده ی ما در بافر ذخیره شده ،سپس آدرس شروع بافر در ثبات EAX قرار میگیرد درواقع ۴ بایت از سریال ما درون EAX قرار میگیرد و این بایت ها با 3334 مقایسه میشوند اگر نتیجه ی مقایسه صحیح نبود مقدار EAX=1 میشود (بد) و اگر نتیجه مقایسه صحیح بود EAX=0 میشود (خوب):

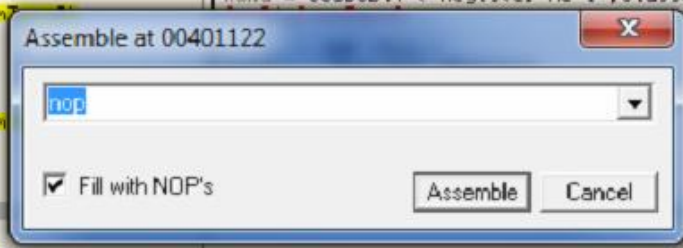
00401111	EB 6A000000	CALL <JMP.&user32.GetDlgItemTextA>	GetDlgItemTextA
00401116	B8 98304000	MOV EAX,Register.00403098	
00401118	B800	MOV EAX,DWORD PTR DS:[EAX]	
0040111D	66:3D 3433	CMP AX,3334	
00401121	75 07	JNZ SHORT Register.0040112A	EAX == 0
00401123	B8 00000000	MOV EAX,0	
00401128	EB 05	JMP SHORT Register.0040112F	EAX == 1
0040112A	B8 01000000	MOV EAX,1	
0040112F	C3	RET	
00401130	EB C2FFFFFF	CALL Register.004010FC	

به آدرس 401121 نگاه کنید یک دستورالعمل ارجاعی (JNZ) :

00401118	B800	MOV EAX,Register.00403098
0040111D	66:3D 3433	MOV EAX,DWORD PTR DS:[EAX] CMP AX,3334
00401121	75 07	JNZ SHORT Register.0040112A
00401123	B8 00000000	MOV EAX,0
00401128	EB 05	JMP SHORT Register.0040112F
0040112A	B8 01000000	MOV EAX,1
0040112F	C3	RET

این دستور مشخص کننده ی مقدار EAX است که میتواند یا صفر باشد یا یک ،اگر ما کاری کنیم که این مقدار همیشه صفر را برگرداند ،کار تمام است:

004010EB	68 ED030000	PUSH 3ED	ControlID = 3ED (1005.)
004010F0	FF35 94304000	PUSH DWORD PTR DS:[403094]	hwnd = 00380D44 ('Register Me 1',class='#32770')
004010F6	EB 91000000	CALL <JMP.&user32.SetDlgItemTextA>	
004010FB	C3	RET	
004010FC	68 00020000	PUSH 200	
00401101	68 98304000	PUSH Register.00403098	
00401106	68 E9030000	PUSH 3E9	
0040110B	FF35 94304000	PUSH DWORD PTR DS:[403094]	
00401111	E8 6A000000	CALL <JMP.&user32.GetDlgItemTextA>	
00401116	B8 98304000	MOV EAX,Register.00403098	
00401118	B800	MOV EAX,DWORD PTR DS:[EAX]	
0040111D	66:3D 3433	CMP AX,3334	
00401121	90	NOP	
00401122	90	NOP	
00401123	B8 00000000	MOV EAX,0	
00401128	EB 05	JMP SHORT Register.0040112F	
0040112A	B8 01000000	MOV EAX,1	
0040112F	C3	RET	
00401130	EB C7FFFFFF	CALL Register.004010FC	
00401135	0BC0	OR EAX,EAX	



بله درست مدس زدید ،تبدیل این دستور به NOP با اینکار همیشه مقدار EAX=0 خواهد بود و در خط بعد با یک پرش به روتین اصلی برمیگردیم :

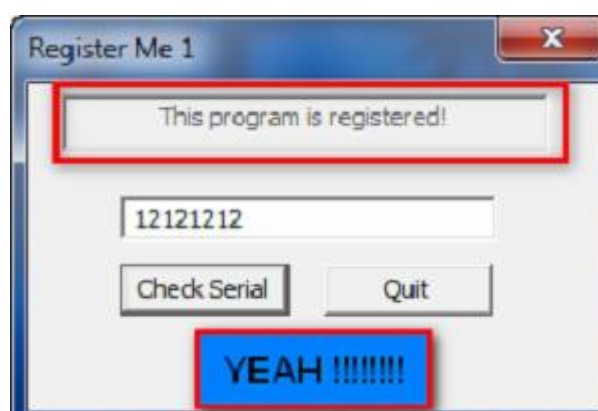


به نظر میرسد همه چیز درست شده است در فضا 401137 هم از حلقه عبور کردیم ☺ :

```

0040112F  C3          RETN
00401130  E8 C7FFFFFF CALL Register.004010FC
00401135  0BC0       OR EAX,EAX
00401137  74 33      JE SHORT Register.0040116C
00401139  B9 1F000000 MOV ECX,1F
0040113E  BE 00000000 MOV ESI,0
00401143  33C0       XOR EAX,EAX
00401145  8A86 70304000 MOV AL,BYTE PTR DS:[ESI+403070]
00401148  83F0 2C     XOR EAX,2C
0040114E  8B86 70304000 MOV BYTE PTR DS:[ESI+403070],AL
00401154  46         INC ESI
00401155  E2 EE      LOOPD SHORT Register.00401145
00401157  68 70304000 PUSH Register.00403070
0040115C  68 ED030000 PUSH 3ED
00401161  FF35 94304000 PUSH DWORD PTR DS:[403094]
00401167  E8 20000000 CALL <JMP.&user32.SetDlgItemTextA>
0040116C  C3          RETN
0040116D  CC          INT3
  
```

نکته: دستور قبل از JE مقدار EAX را تست میکند اگر برابر با ۱ بود پرش صورت نمیگیرد و وارد حلقه میشویم در غیر این صورت پرش انجام میشود :



بسیار عالی، ما فقط با اعمال یک پچ توانستیم دو محدودیت را برداریم یکی سریال و دیگری متن داخل تکس باکس در پنجره ی اصلی

• ASCII Table Plugin

Hex	Dec	Char		
3D	061	075	00111101	=
3E	062	076	00111110	>
3F	063	077	00111111	?
40	064	100	01000000	@
41	065	101	01000001	A
42	066	102	01000010	B
43	067	103	01000011	C
44	068	104	01000100	D
45	069	105	01000101	E
46	070	106	01000110	F
47	071	107	01000111	G
48	072	110	01001000	H
49	073	111	01001001	I
4A	074	112	01001010	J
4B	075	113	01001011	K
4C	076	114	01001100	L
4D	077	115	01001101	M
4E	078	116	01001110	N
4F	079	117	01001111	O
50	080	120	01010000	P
51	081	121	01010001	Q
52	082	122	01010010	R
53	083	123	01010011	S
54	084	124	01010100	T
55	085	125	01010101	U
56	086	126	01010110	V

