

vc-4. 文字データと文字コード (Visual Studio C++の機能と操作演習, 全5回)

https://www.kkaneko.jp/cc/vc/index.html

金子邦彦





ASCII 文字コード表



コンピュータで

英数文字データを扱うときの標準

	0	1	2	3	4	5	6	7
0	NULL	DEL	SP	0	@	Р		р
1	SOH	DC1	!	1	Α	Q	а	q
2	STX	DC2	"	2	В	R	b	r
3	ЕТХ	DC3	#	3	С	S	с	s
4	EOT	DC4	\$	4	D	Т	d	t
5	ENQ	NAK	%	5	Е	U	е	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ЕТВ	'	7	G	W	g	w
8	(BS)	CAN	(8	н	Х	h	x
9	(HT)	EM)	9	I	Υ	i	у
А	(LF)	SUB	*	:	J	Z	j	z
В	(VT)	ESC	+	;	К	[k	{
С	(FF)	(FS)	,	<	L	¥	1	
D	(CR)	(GS)	-	=	М]	m	}
Е	SO	(RS)	•	>	Ν	^	n	~
F	SI	(US)	/	?	0	_	0	DEL

太字は特別用途





- Visual Studio 2015 を起動しなさい
- ・Visual Studio 2015 で、Win32 コンソールアプリ ケーション用プロジェクトを新規作成しなさい

プロジェクトの「名前」は何でもよい



・Visual Studio 2015 のエディタを使って、ソース ファイルを編集しなさい





・ビルドしなさい、ビルドのあと「1 正常終了、 0 失敗」の表示を確認しなさい

• Visual Studio 2015 で「char message = "hello";」 の行に、ブレークポイントを設定しなさい



• Visual Studio 2015 で、デバッガーを起動しなさい.





「char message = "hello";」の行で、実行が中断することを確認しなさい

あとで使うので、中断したままにしておくこと





「char message = "hello";」の行で、実行が中断 した状態で、変数の値を表示させなさい.手順は 次の通り.

		_	Ç	マ クイッ	ク起	動 (Ctrl+Q)	- □	×
	デノ	「ッグ(<mark>I</mark>) チーム(M) ツール(T) テスト(S) 分析	(N) ウィン	ドウ(W) ヘレプ(H)	サインイン	
-	Г	ウィンドウ(W)		•	•	ブレークポイント(B)	Ctr	I+Al
١ř		999499A		•	E4	出力(0)		
		続行(C)	F5		<i>71</i> .	GPU スレッド(U)		
	н	すべて中断(K)	Ctrl+/	Alt+Break	۵	タスク(S)	Ctr	l+Sł
	•	デバッグの停止(E)	Shift+	-F5	a	並列スタック(K)	Ctr	l+Sł
	×	すべてデタッチ(D)			•	並列ウォッチ(R)		
		すべて中止(M)				ウォッチ(W)		_
	ð	再起動(R)	Ctrl+9	Shift+F5	F	自動変数(A)	Ctr	I+Al
	o [©]	プロセスにアタッチ(P)			쾨	ローカル(L)	Ctr	I+Al
		例外(X)	Ctrl+/	Alt+E		イミディエイト(I)	Ctr	I+Al
-		パフォーマンスと診断(F)	Alt+F	2	П	JavaScript コンソール	(J) Ctr	- +A -
	ς.	ステップ イン(I)	F11			DOM Explorer(D)	Ctr	1+AI
	G,	ステップ オーバー(0)	F10			[[[[]]][[]]][[]][[]]][[]][[]]][[]][[]]	Ctr	·اــــــــــــــــــــــــــــــــــــ
	¢	ステップ アウト(T)	Shift+	-F11	~		Ctr	4 L AF

ローカル			
名前		値	
Þ 🤗	message	0x004ffc0c "77777	17







• 「 char message = "hello";」の行で、実行が中断 した状態で、メモリの中身を表示させなさい、手 順は次の通り.

8)	デバッグ(D) ーム(M) ツール(T) テスト(S) ご	アーキテクチャ(C) 分析	(N) ウィンドウ(W) ヘルプ(H)					
₸(C)	ウィンドウ(W)	•	ブレークポイント(B)	Ctrl+Alt+B	→ 6. 6, 6 월 □	ドマップ 🛛 褐	メモリ 1 こううううう	
フサイ	· · · · · · · · · · · · · · · · · · ·	•	【 出力(O)				7547.000	1171700
	▶ 続行(C)	F5	🚧 GPU スレッド(U)				0.01171700	
23-	■ すべて中断(K)	Ctrl+Alt+Break		Ctrl+Shift+D, K			UXUI171790	
_	■ テバックの停止(E)	Shift+F5	並列スタック(K)	Ctrl+Shift+D, S			UXUII/I/BE	. 40 TU 66 60 UG 34 60 17 UI 66 63 40 T4 60 40 TU 3U 66 36 . ff 59 50 5f 50 5h 9h 4d fo 22 od 59 Ch fo ff ff 91 od d4
	▼ ∮へ(ナダツナ(D)		並列ウォッチ(R)	•			0,01171760	ff ff ng nn nn nn ng 10 17 n1 ga g5 79 79 g1 g7 g5 nn an
1	9 正記動(R)	Ctrl+Shift+F5	ウォッチ(W)	•			0x01171014	
	* プロセスにアタッチ(P)	carrometro	局 自動変数(A)	Ctrl+Alt+V, A			0×01171876	00 b8 cc cc cc cc f3 ab e8 44 f9 ff ff 8d 45 0c 89 45 ec
″ł	その他のデバッグターゲット	•		Ctrl+Alt+V, L			0x011718A4	f8 ff ff 50 e8 de f8 ff ff 83 c4 10 89 45 f8 c7 45 ec 00
as	例外(X)	Ctrl+Alt+E	127111F(I)	Ctrl+Alt+I	-		0x011718D2	
~c	パフォーマンスと診断(F)	Alt+F2	DOM Explorer	Ctrl+Alt+V, C	.		0x01171900	1b 8b 55 08 85 d2 74 14 57 b0 cc 8b fe 8b cb f3 aa 8b 02
	ステップ イン(I)	F11		Ctrl+Alt+C			0x0117192E	cc cc 55 8b ec 51 53 8b da 89 4d fc 56 57 85 db 74 4d 33
	🤇 ステップ オーバー(O)	F10	22 スレッド(H)	Ctrl+Alt+H				
	C ステップ アウト(T)	Shift+F11	■ モジュール(O)	Ctrl+Alt+U				
1	□-ドマップに呼び出し履歴を表示(K)	Ctrl+Shift+`	_# T⊓+r7(₽)	Ctrl+Alt+Z				⑤ 「 ノ エ 凵 1 凵 の 両 📗
	↔ クイック ウォッチ(Q)	Shift+F9	メモリ(M)			+Alt+M,		② 入てり I 」 の回
	図 Windows Phone アプリケーション分析の開始	Alt+F1		Ctrl+Alt+D		- Alto- Wi, 2		
	ブレークポイントの設定/解除(G)	F9	■ レジスタ(G)	Ctrl+Alt+G	🛱 メモリ 3(3) Ctrl-	+Alt+M, 3		■
	ブレークポイントの作成(B)	•			第 メモリ 4(4) Ctrl-	+Alt+M, 4		
	すべてのブレークポイントの削除(D) すべてのブレークポイントを無効にする(N)	Ctrl+Shift+F9						
			1					





「メモリ1」の画面の「アドレス」に 「&message」と入れて Enter キーを押しなさい

メモリ 1				አ£በ 1
アドレス: &m	essage			アドレス: 0x004FFD84
0x010FF9C8 0x010FF9F6 0x010FFA24 0x010FFA52 0x010FFA52 0x010FFA80 0x010FFAAE 0x010FFAAC 0x010FFAAC	77 77 77 77 77 77 77 77 77 77 77 77 77 77	?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ??	?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ?? ??	A 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

画面が変化するので確認する



・ステップオーバーの操作を1回行いなさい

ſ	デノ	(ッグ(D)	チーム(M)	ツール(T)	テスト(S)	. 分析(N
٣		ワイント	ל(W)			
К		グラフィ	ックス			
		続行(C)				F5
	н	すべて中国	断(K)			Ctrl+Alt
-		デバッグの	の停止(E)			Shift+F
	×	すべてディ	タッチ(D)			
		すべて中」	止(M)			
	ð	再起動(R))			Ctrl+Sh
	ø	プロセス	にアタッチ(P)			
		例外(X)				Ctrl+Alt
-		パフォー	マンスと診断(F)		Alt+F2
	ς.	ステップ	イン(I)			F11
	4	ステップ	オーバー(0)			F10
	¢,	ステップ	アウト(T)			Shift+F
	⇔	クイック	ウォッチ(Q)			Shift+F
		ブレークバ	ポイントの設定	È/解除(G)		F9
		ブレーク	ポイントの作成	聀(B)		



Xモリ1 ≫	:-:-:-:-:-:-		-0-0-0-0-0			:-:-::	-0-0-0-0-0				-0-0-0-0-0	0-0-0-0-0-			
アドレス:	0x004	4FFD	84												
0x004FFI 0x004FFI 0x004FFI)84)82)E0	cc 45 80	cc b1 95	cc 46 Ob	cc 10 01	cc Ob OO	cc 01 00	cc 46 00	cc 10 00	cc Ob bO	cc 01 fd	cc 00 4f	сс 60 00	d7 20 00	e: 0(0(

char message[] = "hello";

XEU 1 🛛															
アドレス:	0x004	4FFD	84												
0x004FF 0x004FF	D84 DB2	<mark>68</mark> 45	65 b1	<mark>6c</mark> 46	<mark>6c</mark> 10	<mark>6f</mark> Ob	<mark>00</mark> 01	сс 46	сс 10	cc Ob	сс 01	сс 00	сс 60	d7 20	e: 0(
0x004FF	DEO	80	95	Ōþ	01	00	00	00	00	bŪ	fd	4f	00	00	0(





メモリの中身は 68 65 6c 6c 6f 00 (こ変化 68→h 65→e 6c→l 6c→l 6f→o 00→文字列の終わり



・プログラム実行の再開の操作を行いなさい。これ で、デバッガーが終了する。





・プログラムを次のように書き換えて、同じことを もう1度行いなさい

