

# エスケープシーケンスとは

---

# エスケープ(Escape)シーケンスとは

---

- Escapeには、「逃げる」、「逃れる」の意味がある
  - エスケープシーケンスには、エスケープ文字とエスケープシーケンスがある
- エスケープ文字
  - C言語で、直接入力できない特殊文字(改行など)は、「¥」後に文字を付けて指定する
  - 例: ¥n は、改行コードを生成する
  - エスケープ文字の詳細は後述の「C言語のエスケープ文字」を参照のこと
- エスケープシーケンス
  - 本来のエスケープシーケンスとは、エスケープ記号(ESC:0x1B)に続く複数の文字で指定
  - エスケープシーケンスは、プログラムの動作モードを切り替える
  - その典型的な使用例が、画面制御や文字コードの切り替え
    - ◆ 画面クリア、カーソル位置の指定、カラー指定などがある
    - ◆ 漢字表示モードと普通の英数字の表示モードとの切り替え
  - 画面制御については、国際規格(ANSI)で定義されている

# テレタイプ

- テレタイプとは、テレフォンタイプライターの略
  - 電話回線で受信した文字コードを印字する装置
- 当初、電話回線で送る文字コードは7ビットだった
  - 英数字なら7ビット(128種類)で十分だった
    - ◆ ビット数が少ないということは、それだけ伝送するビット数が減る
  - その中で、0x00～0x1Fまでは、制御コードが割り当てられていた
    - ◆ 0x0d(CR:Carriage Return) ヘッドを左側に移動しなさいの意味
    - ◆ 0x0a (LF:Line Feed) 1行を紙を送る (改行)という意味
    - ◆ 0x02(STX:Start of TeXt) これから文字を送りますという意味
    - ◆ 0x03(ETX:End of TeXt) これで文字を終わりますという意味

現在は使用されていないテレタイプ →



# 8ビット(8単位)ASCII文字コード表

	上位	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
下位		0x	1x	2x	3x	4x	5x	6x	7x	8x	9x	Ax	Bx	Cx	Dx	Ex	Fx
0000	x0	NUL	DLE	SP	0	@	P	`	p				ー	タ	ミ		
0001	x1	SOH	DC1	!	1	A	Q	a	q			。	ア	チ	ム		
0010	x2	STX	DC2	"	2	B	R	b	r			「	イ	ツ	メ		
0011	x3	ETX	DC3	#	3	C	S	c	s			」	ウ	テ	モ		
0100	x4	EOT	DC4	\$	4	D	T	d	t			。	エ	ト	ヤ		
0101	x5	ENQ	NAK	%	5	E	U	e	u			・	オ	ナ	ユ		
0110	x6	ACK	SYN	&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
0111	x7	BEL	ETB	'	7	G	W	g	w			ア	キ	ヌ	ラ		
1000	x8	BS	CAN	(	8	H	X	h	x			イ	ク	ネ	リ		
1001	x9	HT	EM	)	9	I	Y	i	y			ウ	ケ	ノ	ル		
1010	xA	LF	EOF	*	:	J	Z	j	z			エ	コ	ハ	レ		
1011	xB	VT	ESC	+	;	K	[	k	{			オ	サ	ヒ	ロ		
1100	xC	FF	FS	,	<	L	¥	l				ヤ	シ	フ	ワ		
1101	xD	CR	GS	-	=	M	]	m	}			ユ	ス	ヘ	ン		
1110	xE	SO	RS	.	>	N	^	n	~			ヨ	セ	ホ	ゝ		
1111	xF	SI	US	/	?	O	_	o	DEL			ッ	ソ	マ	。		

シリアル通信では、この部分のコードを、  
文字として転送できない

Shift-JISコードの第1バイト目のコード  
0x81~0x9f || 0xE0~0xEC

# エスケープシーケンスの誕生

- テレタイプ時代は、制御コード(0x00~0x1f)で十分だった
- しかし、その後テレタイプから画面表示になり、より複雑な制御が必要になった
- そこで、登場したのが、エスケープシーケンス
  - 1バイトの文字コードだけ(0x00~0xff)では、文字コードが足りない
  - そこで、エスケープ文字(ESC:0x1b)に続けて複数の文字を組み合わせることで実現
  - 受信側は、エスケープ文字が来ると、続く文字を調べて動作モードを切り替える
  - ライフゲームでは、下記のエスケープシーケンスを使いました
    - ◆ ESC[2J          画面クリア
    - ◆ ESC[m;nH      カーソル位置を、高さm行目、横n文字目に移動
  - カレンダーのカラー表示に使う
    - ◆ ESC[ccm      ccの色にする
  - この他に、文字コードの切り替えなどがあります
    - ◆ ESC\$B          これから、JIS漢字コードに切り替えます
    - ◆ ESC(B          これから、半角の英数数字に切り替えます
    - ◆ ESC\$A          これから、中国語に切り替えます

# C言語のエスケープ文字

- 前ページまでで、文字と文字列、書式指定の説明をしました。その中で、
  - 「“(ダブルクォーテーション)で挟まれた文字群が文字列
  - 書式指定は、文字列の中に、%(指定子)と次に続く英文字で指定する
  - そこで、文字列の中に「“(ダブルクォーテーション)や「%(指定子)記号を入れるにはどうした良いかという疑問がでませんか?
- そのために、エスケープ文字が用意されています。
  - エスケープ文字とは「¥」\*1)で表し、「¥」に続く文字は、特殊文字として扱われます。

エスケープ文字	コード(16進)	機能
¥0	00	文字列の最後(NULL)
¥a	07	ベルを鳴らす(BEL)
¥b	08	バックスペース(BS)
¥f	0C	改頁(FF)
¥n	0A	改行(LF)
¥r	0D	復帰(CR)
¥t	09	水平タブ(HT)
¥v	0B	垂直タブ(VT)
¥¥	5C	「¥」¥記号自身
¥'	27	「'」シングルクォーテーション
¥"	22	「"」ダブルクォーテーション
¥?	3F	「?」クエスチョン マーク
¥ooo		1～3桁の8進数の文字
¥xhh		1～2桁の16進数の文字
%%	25	「%」%の表示

\*1): ¥マークは、日本語文字表示です。国際文字では、「\」バックスラッシュの表示になります。

エスケープ文字と類似の言葉に、画面制御を行う、ESCコード(0x1B)で始まる「エスケープシーケンス」と言われるものがあります。混同しないください。

%は、%%です。間違わないように

# 画面制御のエスケープシーケンス

記述	意味
ESC[nA	カーソルを上にな移動させる。(nには整数が入る、省略すると1)
ESC[nB	カーソルを下にな移動させる。(nには整数が入る、省略すると1)
ESC[nC	カーソルを右にな移動させる。(nには整数が入る、省略すると1)
ESC[nD	カーソルを左にな移動させる。(nには整数が入る、省略すると1)
ESC[nE	カーソルをn行下の先頭に移動させる。(nには整数が入る、省略すると1)
ESC[nF	カーソルをn行上の先頭に移動させる。(nには整数が入る、省略すると1)
ESC[nG	カーソルを現在の横位置に関係なく左端からnの場所に移動させる。 (左端を1とする。nには整数が入る、省略すると1)
ESC[n;mH	カーソルを現在の位置に関係なく上端からn、左端からmの場所に移動させる。 左上を1,1とする座標指定だが、1つ目の数字が縦位置で、2つ目の数字が横位置であることに注意 (n,mには整数が入る、省略すると1)
ESC[nJ	画面消去、nを省略、もしくは0を指定した場合、カーソルより後ろを消去、 1を指定するとカーソルより前を消去、2を指定すると全体を消去となる。
ESC[nK	行消去、nを省略、もしくは0を指定した場合、カーソルより後ろを消去、 1を指定するとカーソルより前を消去、2を指定すると行全体を消去となる。
ESC[nS	n行分、コンソールを次にスクロールする。(nには整数が入る、省略すると1)
ESC[nT	n行分、コンソールを前にスクロールする。(nには整数が入る、省略すると1)
ESC[n;mf	ESC[n;mHと同じで、カーソルを座標移動する
ESC[nm	SGR(Select Graphic Rendition)というコマンドで、グラフィカルな指示を行う、バリエーション豊富なので後述

# フォントやカラー指定のエスケープシーケンス

記述	意味
ESC[0m	指定をリセットし未指定状態に戻す。 SGRによって変化させたあとは、必ずこのコードを出力して他の出力に影響を与えないようにする
ESC[1m	ボールド指定、フォントが対応していれば太字になる。TeraTermでは色が変わり、この色は設定で指定する
ESC[2m	薄く表示する、あまり対応されていないらしいが、gnome-terminalでは文字色が変化している
ESC[3m	イタリック表示、フォントが対応していればイタリック体になる
ESC[4m	アンダーライン
ESC[5m	ブリンク、毎分150回以下のペースで点滅する。TeraTermでは色が変わり、この色は設定で指定する
ESC[6m	高速ブリンク、あまり対応されていないらしい
ESC[7m	反転、前景色と背景色を入れ替えて表示
ESC[8m	表示を隠す、gnome-terminalとxtermでは見えない。 選択による反転でも見えないが、コピーアンドペーストすると、ちゃんとその文字が出力されていることが分かる
ESC[9m	取り消し、gnome-terminalでは取り消し線が表示された
ESC[30m～ ESC[37m	文字色指定、色番号0～7は後述
ESC[38m	文字色指定拡張用、引き続き引数をとる。5;xで、0～255のカラーインデックス指定。 2;r;g;bでRGB指定。 ただし、24bitカラーが表示できる端末は少ない
ESC[39m	文字色をデフォルトに戻す
ESC[40m～ ESC[47m	背景色指定、色番号0～7は後述
ESC[48m	背景色指定拡張用、引き続き引数をとる。5;xで、0～255のカラーインデックス指定。 2;r;g;bでRGB指定。 ただし、24bitカラーが表示できる端末は少ない
ESC[49m	背景色をデフォルトに戻す
ESC[90m～ ESC[97m	前景色指定、30番代の指定より強い色。標準ではないらしい
ESC[100m ～ ESC[107m	背景色指定、40番代の指定より強い色。標準ではないらしい



# JIS漢字コードの場合

文字	A	B	C	(漢字)	あ	い	う	(ASCII)	X	Y	Z
コード	41	42	43	1B 24 42	24 22	24 24	24 26	1B 28 42	58	59	5A

規格	記号表記	16進表記	意味
ISO-2022-JP	ESC ( B	1B 28 42	ASCII。
	ESC ( J	1B 28 4A	JIS X 6220(JIS X 0201)-1976 ラテン文字集合
	ESC \$ @	1B 24 40	JIS C 6226(JIS X 0208)-1978(通称：旧JIS)
	ESC \$ B	1B 24 42	JIS X 0208-1983(通称：新JIS) または JIS X 0208-1990
ISO-2022-JP-1 で追加	ESC \$ ( D	1B 24 28 44	JIS X 0212-1990(JIS補助漢字)
ISO-2022-JP-2 で追加	ESC \$ A	1B 24 41	GB 2312-80(中国語)
	ESC \$ ( C	1B 24 28 43	KS X 1001-1992(韓国語)
	ESC . A	1B 2E 41	ISO/IEC 8859-1 の右半分
	ESC . F	1B 2E 46	ISO/IEC 8859-7 の右半分
ISO-2022-JP-3 で追加	ESC \$ ( O	1B 24 28 4F	JIS X 0213:2000の1面
	ESC \$ ( P	1B 24 28 50	JIS X 0213:2000の2面
ISO-2022-JP-2004	ESC \$ ( Q	1B 24 28 51	JIS X 0213:2004の1面
<a href="#">ISO-IR 013</a>	ESC ( I	1B 28 49	JIS X 0201(半角カナ) ... 一部のソフトでのみ利用

参考URL <https://www.tohoho-web.com/ex/charset.html>