Lesson 9. Fundamental Data Types — Character

Brief overview on character data type

ASCII Table

ASCII	TABLE
Decimal Hey Char	Decimal Hev

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22		66	42	В	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	C
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27		71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	н	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	1	105	69	
10	A	[LINE FEED]	42	2A		74	4A	J	106	6A	j
11	В	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	1
13	D	[CARRIAGE RETURN]	45	2D		77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E		78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	0	111	6F	0
16	10	[DATA LINK ESCAPE]	48	30	O	80	50	P	112	70	р
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	Jm 52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	S
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	V
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	w	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	X
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	10	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	10	[GROUP SEPARATOR]	61	3D	=	93	5D	1	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]
								_			

define and declare a character variable

```
char some_variable_name = 'N';
//注意字符要加上单引号
//字符变量一次只能保存一个字符,因为它的大小是 1 byte,一次不能容纳超过一个字符
```

```
//并不是必须向字符变量提供字符,还可以为他们分配整数值 char some_variable_name = 65; printf("%c", some_variable_name); //格式说明符为字符,所以用 %c 而不是 %d, 如果用 %d 打印的就是十进制
```

```
1
     #include <stdio.h>
 2
     #include <stdlib.h>
 3
 4
     int main()
 5
 6
         char var = 65;
 7
         printf("%c", var);
 8
         return 0;
 9
10
            "E:\C Programming & Data St X
                                        execution time : 0.628 s
        Process returned 0 (0x0)
        Press any key to continue.
 1
     #include <stdio.h>
     #include <stdlib.h>
 3
 4
     int main()
 5
   ₽ {
 6
         char var = 65;
 7
         printf("%d", var);
 8
         return 0;
 9
10
                                         + | ~
            "E:\C Programming & Data St X
        65
        Process returned 0 (0x0)
                                         execution time : 0.357 s
        Press any key to continue.
```

Size/Range of characters

```
Size: Range:

1 byte = 8 bits

Unsigned: 0 to 255

Signed: -128 to +127
```

Difference between signed and unsigned characters

二进制补码

$$-2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0}$$

$$-128 = 1 0 0 0 0 0 0 0$$

$$2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0}$$

$$+128 = 1 0 0 0 0 0 0 0$$

$$-2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0}$$

$$-127 = 1 0 0 0 0 0 0 1$$

$$2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0}$$

$$2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0}$$

Note: -128 和 +128 , -127 和 +129 的二进制相同,类似的 -126 和 +130 的二进制补码相同.... -1 和 +255 的 二进制表示也相同,所以它们打印出来的效果相同

+129 =

```
1
     #include <stdio.h>
2
     #include <stdlib.h>
3
 4
     int main()
 5
   □ {
 6
         char var = 128;
7
         printf("%c", var);
8
         return 0;
9
10
         © "E:\C Programming & Data St ×
        €
       Process returned 0 (0x0) execution time : 0.383 s
       Press any key to continue.
     #include <stdio.h>
 1
 2
     #include <stdlib.h>
 3
 4
     int main()
 5
 6
         char var = -128;
 7
         printf("%c", var);
 8
         return 0;
 9
10
         © "E:\C Programming & Data St ×
        €
        Process returned 0 (0x0)
                                       execution time : 0.580 s
        Press any key to continue.
```

```
1
     #include <stdio.h>
2
     #include <stdlib.h>
3
4
     int main()
5
6
         char var = -1;
7
         printf("%c", var);
8
         return 0;
9
10
           "E:\C Programming & Data St
                                        + | ~
       Process returned 0 (0x0)
                                       execution time : 0.407 s
       Press any key to continue.
      #include <stdio.h>
 1
 2
      #include <stdlib.h>
 3
 4
     int main()
 5
 6
         char var = 255;
 7
         printf("%c", var);
 8
          return 0;
 9
10
         □S "E:\C Programming & Data St
                                        execution time : 0.512 s
       Process returned 0 (0x0)
       Press any key to continue.
```

conclusion

在字符变量的情况下,负值不会带来任何额外作用,每个负值都相当于 Extended ASCII characters 中的某个正值

Summary

- 1. Size of character = 1 byte
- 2. Signed character range: -128 to +127
- 3. Unsigned character range: 0 to 255
- 4. Negative values won't buy you any additional powers
- 5. In traditional ASCII table, each character requires 7 bits.
- In extended ASCII table, each character utilize all 8 bits.