

1.C is for low-level software development.

Like WeChat, drive or operating system.

本章重点：

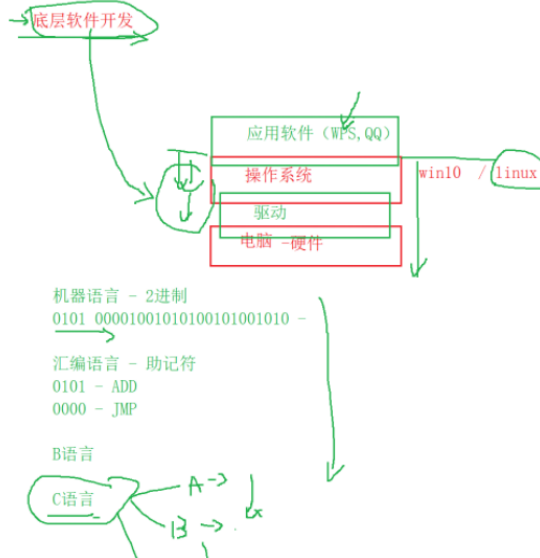
- 什么是C语言
- 第一个C语言程序
- 数据类型
- 变量、常量
- 字符串+转义字符+注释
- 选择语句
- 循环语句
- 函数
- 数组
- 操作符
- 常见关键字
- define 定义常量和宏
- 指针 ✓
- 结构体

ANSI C - C国际标准

C89
C90
C99
...

什么是计算机语言呢？
人和计算机交流的语言 - C/C++/JAVA

语言 - 汉语、英语 - 人和人交流的语言



2. bit and byte

1 byte = 8 bits

数据类型

char //字符数据类型
short //短整型
int //整形
long //长整型
long long //更长的整形
float //单精度浮点数
double //双精度浮点数
//C语言有没有字符串类型?

计算机中的单位:

bit 比特
byte - 字节 = 8bit
kb = 1024byte
mb = 1024kb
gb = 1024mb
tb = 1024gb
pb = 1024tb

```
int main()
{
    //如何计算一个类型创建的变量所占空间的大小呢?
    printf("%d\n", sizeof(char));
    printf("%d\n", sizeof(short));
    printf("%d\n", sizeof(int));
    printf("%d\n", sizeof(long));
    printf("%d\n", sizeof(long long));
    printf("%d\n", sizeof(float));
    printf("%d\n", sizeof(double));
}
```

short age;
short -32878~32767
int

sizeof(long) >= sizeof(int)

9:05 回来

3 ACCII chart

Just an example

ASCII表																												
(American Standard Code for Information Interchange 美国标准信息交换代码)																												
高四位	ASCII控制字符														ASCII打印字符													
	0000							0001							0010		0011		0100		0101		0110		0111			
	十进制	字符	Ctrl	代码	转义	字符解释	十进制	字符	Ctrl	代码	转义	字符解释	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	Ctrl	
0000	0	0		^@	NUL	\0 空字符	16	▶	^P	DLE		数据链路转义	32		48	0	64	@	80	P	96	`	112	p				
0001	1	1	␣	^A	SOH	标题开始	17	◀	^Q	DC1		设备控制 1	33	!	49	1	65	A	81	Q	97	a	113	q				
0010	2	2	␣	^B	STX	正文开始	18	↑	^R	DC2		设备控制 2	34	"	50	2	66	B	82	R	98	b	114	r				
0011	3	3	♥	^C	ETX	正文结束	19	!!	^S	DC3		设备控制 3	35	#	51	3	67	C	83	S	99	c	115	s				
0100	4	4	♦	^D	EOF	传输结束	20	␣	^T	DC4		设备控制 4	36	\$	52	4	68	D	84	T	100	d	116	t				
0101	5	5	♣	^E	ENQ	查询	21	§	^U	NAK		否定应答	37	%	53	5	69	E	85	U	101	e	117	u				
0110	6	6	♠	^F	ACK	肯定应答	22	—	^V	SYN		同步空闲	38	&	54	6	70	F	86	V	102	f	118	v				
0111	7	7	♣	^G	BEL	响铃	23	↑↓	^W	ETB		传输块结束	39	'	55	7	71	G	87	W	103	g	119	w				
1000	8	8	␣	^H	BS	退格	24	↑	^X	CAN		取消	40	(56	8	72	H	88	X	104	h	120	x				
1001	9	9	○	^I	HT	横向制表	25	↓	^Y	EM		介质结束	41)	57	9	73	I	89	Y	105	i	121	y				
1010	10	10	␣	^J	LF	换行	26	→	^Z	ELB		替代	42	*	58	:	74	J	90	Z	106	j	122	z				
1011	11	11	♂	^K	VT	纵向制表	27	←	^_	ESC	␣	输出	43	+	59	;	75	K	91	[107	k	123	{				
1100	12	12	♀	^L	FF	换页	28	␣	^_	FS		文件分隔符	44	,	60	<	76	L	92	\	108	l	124					
1101	13	13	♪	^M	CR	回车	29	↔	^_	GS		组分隔符	45	-	61	=	77	M	93]	109	m	125	}				
1110	14	14	♠	^N	SO	移出	30	▲	^_	RS		记录分隔符	46	.	62	>	78	N	94	^	110	n	126	~				
1111	15	15	♠	^O	SI	移入	31	▼	^_	US		单元分隔符	47	/	63	?	79	O	95	_	111	o	127	␣			*Backspace 代码 1 DEL	
注：表中的ASCII字符可以用“Alt + 小键盘上的数字键”方法输入。																												
2013/08/08																												

4 global scope variable and local scope variables

全局变量和局部变量有什么区别？

转载

cywosp

2011-08-27 13:30:17

3459

收藏 1

分类专栏:

C/C++

文章标签:

编译器

存储

全局变量和局部变量有什么区别？怎么实现的？操作系统和编译器是怎么知道的？

全局变量是整个程序都可访问的变量，生存期从程序开始到程序结束；局部变量存在于模块中(比如某个函数)，只有在模块中才可以访问，生存期从模块开始到模块结束。

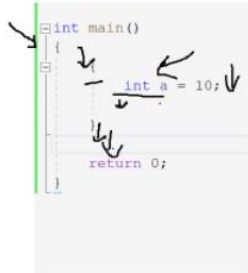
全局变量分配在全局数据段，在程序开始运行的时候被加载。局部变量则分配在程序的堆栈中。因此，操作系统和编译器可以通过内存分配的位置来知道来区分全局变量和局部变量。

全局变量和局部变量的区别是在存储器中位置不同，具体说，全局变量存储在数据段中，局部变量一般来说在堆栈段

Global variables are initialized and loaded when the program start, local variables is stored into temp memory.

✓ 作用域:
 局部变量的作用域: 局部变量所在的局部范围
 全局变量的作用域: 整个工程

✓ 生命周期:
 局部变量的生命周期: 进入局部变量所在的范围, 生命周期开始, 出局部范围生命周期结束
 全局变量的生命周期: 程序的生命周期



Lec2

5. C mainly has 4 following types

Normal variables 1, 2, 3

Const -> constant variables `const int num = 1;` (it can't be changed)

#define variables

And Enum variables

6:

```

// "abcdef" - 字符串
// 数组 - 一组数
// 1 2 3 4 5
// int arr[] = { 1, 2, 3, 4, 5 };
// 字符串面值
// %s - 打印字符串的格式
char ch1[] = "abcdef";
printf("%s\n", ch1);
char ch2[] = { 'a', 'b', 'c', 'd', 'e', 'f' };
// abcdef\0\0\0\0
return 0; 已用时间 <= 1ms

```

名称	值	类型
ch1	0x00fd850 "abcdef"	char[7]
[0]	97 'a'	char
[1]	98 'b'	char
[2]	99 'c'	char
[3]	100 'd'	char
[4]	101 'e'	char
[5]	102 'f'	char
[6]	0 '\0'	char
ch2	0x00fd840 "abcdef..."	char[6]
[0]	97 'a'	char
[1]	98 'b'	char
[2]	99 'c'	char
[3]	100 'd'	char
[4]	101 'e'	char
[5]	102 'f'	char

添加要监视的项

```

// int arr[] = { 1, 2, 3, 4, 5 };
// 字符串面值
// %s - 打印字符串的格式
char ch1[] = "abcdef";
printf("%s\n", ch1);

char ch2[] = { 'a', 'b', 'c', 'd', 'e', 'f' };
printf("%s\n", ch2);

```

Microsoft Visual Studio 调试控制台

```

abcdef
abcdef
D:\2021_code\class102\test_6_5\Debug\tes
按任意键关闭此窗口...

```

Lec 3:

1: C operators

!	✓	逻辑反操作
-	✓	负值
+	✓	正值
&		取地址
sizeof		操作数的类型长度（以字节为单位）
~		对一个数的二进制按位取反
--		前置、后置--
++		前置、后置++
*		间接访问操作符(解引用操作符)
(类型)		强制类型转换

关系操作符:

>

>=

<

<=

!=

==

>

=

逻辑操作符:

&& - 逻辑与 - 并且

|| - 逻辑或 - 或者

7:56 回来

条件操作符 (三目操作符)

? :

```
int main()
{
    int a = 10;
    int b = 0;
    if (a == 5)
    {
        b = -6;
    }
    else
    {
        b = 6;
    }
    b = (a == 5) ? -6 : 6;
    return 0;
}
```

& * . ->

```
8
9 #include <stdio.h>
10
11 int main()
12 {
13     int a = 5;
14     int b = 0;
15     if ( a == 0 ){
16         b = -6;
17     }
18     else{
19         b = 6;
20     }
21
22     b = ( ( a == 5) ? -6 : 6);
23     printf("%d\n",b);
24
25     b = ( ( a == 10) ? -6 : 6);
26     printf("%d",b);
27
28     return 0;
29 }
30
```

-6
6

...Program finished with exit code 0
Press ENTER to exit console.

If statement `a == 5` return true, then it will be -6

If statement `a == 10` return false, then it will be 6.

2 how to use static

static 在C语言中用法:

1. 修饰局部变量 ✓
2. 修饰全局变量 ✓
3. 修饰函数 ✓✓

```
void test()
{
    int a = 1;
    a++;
    printf("%d ", a);
}

int main()
{
    int i = 0;
    while (i < 10)
    {
        test();
        i++;
    }

    return 0;
}
```

Microsoft Visual Studio 调试控制台
2 2 2 2 2 2 2 2 2 2
D:\2021_code\class102\test_6_7\
按任意键关闭此窗口...

✓ static修饰局部变量:
改变了变量的生命周期

不影响作用域

9:05 回来

```
void test()
{
    //static修饰局部变量
    static int a = 1;
    a++;
    printf("%d ", a);
}

int main()
{
    int i = 0;
    while (i < 10)
    {
        test();
        i++;
    }

    return 0;
}
```

Microsoft Visual Studio 调试控制台
2 3 4 5 6 7 8 9 10 11
D:\2021_code\class102\test_6_7\
按任意键关闭此窗口...