

程序设计与C语言

2020年7月9日 20:22

枚举太多? 二分

程序的执行

- 解释: 借助一个程序, 那个程序能试图理解你的程序, 然后按照你的要求执行。
- 编译: 借助一个程序, 就像一个翻译, 把你的程序翻译成计算机真正能懂的语言——机器语言——写的程序, 然后, 这个机器语言写的程序就能直接执行了。

- `printf("Hello World!\n");`
- ""里面的内容叫做“字符串”, `printf` 会把其中的内容原封不动地输出
- `\n`表示需要在输出的结果后面换一行

C语言中空格随意, 分号放上就可以, 位置可以随意

变量: 变量赋值与初始化

初始化

- 当赋值发生在定义变量的时候, 就像程序1中的第7行那样, 就是变量的初始化。虽然C语言并没有强制要求所有的变量都在定义的地方做初始化, 但是所有的变量在第一次被使用(出现在赋值运算符的右边)之前被应该赋值一次
- 如果没有初始化?

C语言中的=是赋值的意思

变量定义: 类型名称 变量名称;

变量使用前必须先定义

如果变量没有初始化, 就直接拿到右边去用, 会出现什么呢?

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int i;
6     int j;
7     j = i+10;
8     printf("%d\n", j);
9     return 0;
10 }
11
12
13
14
15
```

```
1856970856
[Finished in 0.1s]
```

得到了一个非常奇怪的结果, 这是因为, 在内存当中, 我们有一个变量*i* 我们没有给它一个初始值, 那么, 它正好在内存当中, 在什么地方, 那个地方原本有一些什么样的值在里头, 它就是那个值了。

做计算

- `printf("%d\n", 23+43);`
- `%d`说明后面有一个整数要输出在这个位置上
- `printf("23+43=%d\n", 23+43);`

变量的名字

- 变量需要一个名字, 变量的名字是一种“标识符”, 意思是它是用来识别这个和那个的不同的名字。

- 标识符有标识符的构造规则。

基本的原则是: 标识符只能由字母、数字和下划线组成, 数字不可以出现在第一个位置上, C语言的关键字(有的地方叫它们保留字) 不可以用做标识符。

变量初始化

- `<类型名称> <变量名称> = <初始值>;`
- `int price = 0;`
- `int amount = 100;`
- 组合变量定义的时候, 也可以在这个定义中单独给单个变量赋初值, 如:
- `int price = 0, amount = 100;`

C是一种有类型的语言, 所有的变量在使用之前必须定义或声明, 所有的变量必须具有确定的数据类型。数据类型表示在变量中可以存放什么样的数据, 变量中只能存放指定类型的数据, 程序运行过程中也不能改变变量的类型。C99可以在任何地方定义变量

输入是'hello'返回100 (100-0=100), 不是数字, 当成0

const变量一般大写

用户输入东西, 要有地方存放, 先定义变量再scanf
scanf可以读入一个或多个变量

const

- const是一个修饰符，加在int的前面，用来给这个变量加上一个const（不变的）的属性。这个const的属性表示这个变量的值一旦初始化，就不能再修改了。
- int change = AMOUNT - price;
- 如果你试图对常量做修改，把它放在赋值运算符的左边，就会被编译器发现，指出为一个错误。

C99!

读整数

- scanf("%d", &price);
- 要求scanf这个函数读入下一个整数，读到的结果赋值给变量price
- 小心price前面的&

如果输入的不是整数会怎样?

```
int price = 0;
printf("请输入金额 (元): ");
scanf("%d", &price);
int change = 100 - price;
printf("找您%d元.\n", change);
```

常量

```
int price = 0;
printf("请输入金额 (元): ");
scanf("%d", &price);
int change = 100 - price;
printf("找您%d元.\n", change);
const int AMOUNT = 100;
int price = 0;
printf("请输入金额 (元): ");
scanf("%d", &price);
int change = AMOUNT - price;
printf("找您%d元.\n", change);
```

- int change = 100 - price;
- 100这个固定不变的数，叫做常数。直接写在程序里，我们称作直接量（literal，字面量）。
- 更好的方式，是定义一个常量变量：
 - const int AMOUNT = 100;

C99!

tips

- 程序要求读入多个数字时，可以在一行输入，中间用空格分开，也可以在多行输入
- 在scanf的格式字符串中有几个%d，它就等待用户输入一个整数，当然，字符串后面也需要对应有那么多个整数

```
int a;
int b;

printf("请输入两个整数: ");
scanf("%d %d", &a, &b);
printf("%d + %d = %d\n", a, b, a + b);
```

```
int main()
{
    int a = 0;
    int b = 0;
    scanf("%d %d", &a, &b);
    printf("%d %d\n", a, b);
    return 0;
}
```

```
MOOC:cc\ $gcc test.c; ./a.out
1 2
1 0
```

```
MOOC:cc\ $gcc test.c; ./a.out
1,2
1 2
```

```
MOOC:cc\ $more plus.c
#include <stdio.h>

int main()
{
    int a;
    int b;

    printf("请输入两个整数: ");
    scanf("%d %d", &a, &b);
    printf("%d + %d = %d\n", a, b, a + b);

    return 0;
}
MOOC:cc\ $gcc plus.c
MOOC:cc\ $. /a.out
请输入两个整数: 1 2
1 + 2 = 3
MOOC:cc\ $. /a.out
请输入两个整数: 1
2
1 + 2 = 3
MOOC:cc\ $. /a.out
请输入两个整数:
```

```
1
2
1 + 2 = 3
MOOC:cc\ $. /a.out
请输入两个整数: hello
1837482078 + 32767 = 1837514845
MOOC:cc\ $
```

C语言中，两个整数的运算的结果只能是整数

C语言中没有定点数，只有浮点数

当浮点数和整数放到一起运算时，C会将整数转换成浮点数，然后进行浮点数的运算。

- 整数
 - int
 - printf("%d",...)
 - scanf("%d",...)
- 带小数点的数
 - double
 - printf("%f",...)
 - scanf("%lf",...)

要加%d啊啊啊啊啊啊啊啊

```
int main()
{
    int a = 0;
    int b = 0;
    scanf("%d %d", &a, &b);
    printf("%d %d\n", a, b);
    return 0;
}
```

```
int main()
{
    int a = 0;
    int b = 0;

    scanf("%d %d\n", &a, &b);
    printf("%d %d\n", a, b);

    return 0;
}
```

M00C:cc\ \$gcc test.c; ./a.out

1 2

2

1 2

M00C:cc\ \$gcc test.c; ./a.out

1 2

a

1 2

出现在scanf中的东西不是它要输出的，而是它要你输入的!!! 不是它给你看的，小心!

如果出现空格回车，要有一个东西满足空格回车

```
int main()
{
    int a = 0;
    int b = 0;

    scanf("price%d", &a, &b);
    printf("%d %d\n", a, b);

    return 0;
}
```

M00C:cc\ \$gcc test.c; ./a.out

price 1 2

1 2

表达式

- 一个表达式是一系列运算符和算子的组合，用来计算一个值
- 运算符是指进行运算的动作，比如加法运算符“+”，减法运算符“-”
- 算子是指参与运算的值，这个值可能是常数，也可能是变量，还可能是一个方法的返回值

运算符优先级

优先级	运算符	运算	结合关系	举例
1	+	单目不变	自右向左	a+b
1	-	单目取负	自右向左	a-b
2	*	乘	自左向右	a*b
2	/	除	自左向右	a/b
2	%	取余	自左向右	a%b
3	+	加	自左向右	a+b
3	-	减	自左向右	a-b
4	=	赋值	自右向左	a=b

程序是按步执行的

- 程序表达的是顺序执行的动作，而不是关系
- ```
a=b;
b=a;
```
- 是依次执行的，结果使得a和b都得到b原来的值

## 赋值运算符

- 赋值也是运算，也有结果
- a=6的结果是a被赋予的值，也就是6
- a=b=6 → a=(b=6)

## 复合赋值

- 5个算术运算符，+ - \* / %，可以和赋值运算符“=”结合起来，形成复合赋值运算符：“+=”、“-=”、“\*=”、“/=”和“%=”
- total += 5;
- total = total + 5;
- 注意两个运算符中间不要有空格

## “嵌入式赋值”

```
int a = 6;
int b;
int c = 1+(b=a);
```

## 复合赋值

- 不利于阅读
- 容易产生错误

- `total += (sum+100)/2;`
- `total = total + (sum+100)/2;`
- `total *= sum+12;`
- `total = total*(sum+12);`
- `total /= 12+6;`
- `total = total / (12+6);`

## 前缀后缀

- ++和--可以放在变量的前面，叫做前缀形式，也可以放在变量的后面，叫做后缀形式。
- `a++`的值是a加1以前的值，而`++a`的值是加了1以后的值，无论哪个，a自己的值都加了1了。

| 表达式                  | 运算       | 表达式的值       |
|----------------------|----------|-------------|
| <code>count++</code> | 给count加1 | count原来的值   |
| <code>++count</code> | 给count加1 | count+1以后的值 |
| <code>count--</code> | 给count减1 | count原来的值   |
| <code>--count</code> | 给count减1 | count-1以后的值 |

编程时，先想需要几个变量！需要输入输出什么，几个变量

`printf scanf` 需要 `%d` 或者其他

什么浮点数的，需要的是整数还是浮点数要想好

`scanf` 里面的`&`不要忘记！！

## 递增递减运算符

- “++”和“--”是两个很特殊的运算符，它们是单目运算符，这个算子还必须是变量。这两个运算符分别叫做递增和递减运算符，他们的作用就是给这个变量+1或者-1。
- `count++;`
- `count += 1;`
- `count = count + 1;`

- 可以单独使用，但是不要组合进表达式
  - `++j++ -->?`
  - `i++++ -->?`
- `a = b+=c++-d+--e/-f`