



# 知识点总结

· **getchar**和**putchar**函数是用于字符输入输出的标准函数，位于stdio头文件中，使用前要包含stdio头文件。



· **getchar**函数用于从标准输入（通常是键盘）获取一个字符。

语法格式：`getchar()`;

例：把用户输入的一个字符存到变量c里，对应的语句 `char c = getchar()`;

· 注意：

1、如果**缓冲区**里已经存在字符，程序执行到 `getchar` 函数时，会直接读取缓冲区里的字符，而不会等用户输入。

其中，缓冲区用于临时存放数据，当用户输入数据后，数据会先被存放到缓冲区，然后程序再按顺序从缓冲区里读取。

2、用户输入结束按下的回车键会产生一个**换行符**留在缓冲区中，可以用**getchar**函数来读取掉这个换行符，避免影响后续的读取。



- **putchar函数**用于向标准输出（通常是屏幕）输出一个字符。

语法格式：putchar(**字符常量 / 字符变量 / 字符的ASCII码值**);

例：用putchar函数输出大写字母A(ASCII码值为65)，代码如下：

```
#include <stdio.h>
```

```
int main() {  
    // putchar(字符常量);  
    putchar('A');  
  
    // putchar(字符变量);  
    char c = 'A';  
    putchar(c);  
  
    // putchar(字符的ASCII码值);  
    putchar(65);  
}
```



## ·关于字符运算：

因为字符是**以其对应的ASCII码值进行存储**的，所以字符可以直接参与运算。例，`'A' + 1`（结果为`'B'`），`'A' < 'E'`（结果为真）等。

**大小写字母之间的关系：大写字母的ASCII码值加上32，即为对应小写字母的ASCII码值。**

例，`'A' + 32 == 'a'`为真。



题目1、有以下程序

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
    char c1, c2, c3, c4, c5, c6;
```

```
    scanf("%c%c%c%c", &c1, &c2, &c3, &c4);
```

```
    c5 = getchar();
```

```
    c6 = getchar();
```

```
    putchar(c1);
```

```
    putchar(c2);
```

```
    printf("%c%c\n", c5, c6);
```

```
}
```

程序运行后，若从键盘输入（从第1列开始）

123<回车>

45678<回车>

则输出结果是（ A ）

A. 1245

B. 1256

C. 1278

D. 1267

题目2、有以下程序：

```
#include <stdio.h>
main()
{
    char a, b, c, d;
    scanf("%c%c", &a, &b);
    c = getchar();
    d = getchar();
    printf("%c%c%c%c\n", a, b, c, d);
}
```

当执行程序时，按下列方式输入数据  
(从第一列开始，<CR>代表回车，注意：回车是一个字符)

12<CR>

34<CR>

则输出结果是 ( C )

A. 1234

B. 12

C. 12<CR>3

D. 12<CR>34





题目3、有以下程序：

```
#include <stdio.h>
main()
{
    char ch = 'Z';
    ch = (ch - 'A' + 1) % 26 + 'A';
    putchar(ch);
}
```

程序的运行结果是（ D ）

A. Z

B. Y

C. B

D. A



大写字母的ASCII码值加上32，就是对应小写字母的码值

题目4、以下不能输出小写字母a的选项是 (A)

A. `printf("%c\n", "a");`

B. `printf("%c\n", 'A' + 32);`

C. `putchar(97);`

D. `putchar('a');`