



知识点多总结





- 嵌套循环是指在一个循环的循环体中包含另一个循环，比如在for循环里包含一个for循环。
- 除了for循环可以嵌套之外，while循环、do while循环，以及它们互相之间也可以嵌套，比如for循环里嵌套while循环。
- 关于嵌套循环的执行过程：
 - 1、程序首先从外层循环开始执行，只有外层循环的条件为真时，才会进入内层循环。
 - 2、每执行一次外层循环，内层循环都会从头开始执行，直到内层循环的条件为假。
 - 3、只有在内层循环结束退出后，程序才会返回到外层循环，执行外层循环体中的剩余代码，然后继续进行下一次外层循环的判断。
 - 4、上述过程会持续重复，直到外层循环的条件为假，外层循环结束。



- 嵌套循环是指在一个循环的循环体中包含另一个循环，比如在for循环里包含一个for循环。
- 除了for循环可以嵌套之外，while循环、do while循环，以及它们互相之间也可以嵌套，比如for循环里嵌套while循环。

· 语法格式：

for循环嵌套while循环：

```
for(初始化表达式1; 条件表达式1; 更新表达式1) {  
    // 外层for循环体  
    while(条件表达式2) {  
        // 内层while循环体  
    }  
    // 外层for循环体  
}
```

while循环嵌套for循环：

```
while(条件表达式1) {  
    // 外层while循环体  
    for(初始化表达式; 条件表达式; 更新表达式) {  
        // 内层for循环体  
    }  
    // 外层while循环体  
}
```



·语法格式：

for循环嵌套**for**循环：

```
for(初始化表达式1; 条件表达式1; 更新表达式1) {
```

```
    // 外层for循环体
```

```
    for(初始化表达式2; 条件表达式2; 更新表达式2) {
```

```
        // 内层for循环体
```

```
}
```

```
    // 外层for循环体
```

```
}
```

while循环嵌套**while**循环：

```
while(条件表达式1) {
```

```
    // 外层while循环体
```

```
    while(条件表达式2) {
```

```
        // 内层while循环体
```

```
}
```

```
    // 外层while循环体
```

```
}
```



例，

```
for(int i = 0; i < 3; i++) {  
    for(int j = 0; j < 5; j++) {  
        printf("*");  
    }  
    printf("\n");  
}
```

对应的输出结果如下：

执行过程：

- 1、从外层for循环开始执行，i被初始化为0，判断条件*i < 3*为真，执行内层for循环。
- 2、内层for循环执行一次会打印出一个星号，j从0递增到4，一共执行五次，会打印出5个星号。
- 3、内层for循环结束退出后，程序回到外层循环体，执行剩余的代码，打印出一个换行符，然后外层的i自增1，继续执行下一次的外层循环。
- 4、程序会重复上面的过程，直到i变为3，外层for循环结束。



题目1、有以下程序：

```
#include<stdio.h>
main() {
    int x = 0, y = 6;
    do {
        while(--y) x++;
    } while(y--);
    printf("%d,%d\n", x, y);
}
```

程序的运行结果是 (C)

A. 5,0

B. 6,0

C. 5,-1

D. 6,-1



题目2、有以下程序：

```
#include <stdio.h>
main() {
    int i, j;
    for(i = 3; i >= 1; i--) {
        for(j = 1; j <= 2; j++)
            printf("%d", i + j);
        printf("\n");
    }
}
```

程序运行的结果是 (D)

- A. 234<换行>345<换行>
- B. 432<换行>543<换行>45<换行>
- C. 23<换行>34<换行>23<换行>
- D. 45<换行>34<换行>23<换行>



题目3、有以下程序：

```
#include <stdio.h>

main() {
    int i, j, x = 0;
    for(i = 0; i < 2; i++) {
        x++;
        for(j = 0; j <= 3; j++) {
            if(j % 2) continue;
            x++;
        }
        x++;
    }
    printf("x = %d\n", x);
}
```

程序的运行结果是 (A)

- A. x=8
- B. x=4
- C. x=6
- D. x=12



题目4、有以下程序：

```
#include <stdio.h>
main() {
    int i, j, m = 1;
    for(i = 1; i < 3; i++) {
        for(j = 3; j > 0; j--) {
            if(i * j > 3) break;
            m *= i * j;
        }
    }
    printf("m=%d\n", m);
}
```

程序运行后的输出结果是 (A)

- A. m=6
- B. m=2
- C. m=4
- D. m=5



题目5、请编写一个程序，打印出1到100之间的所有素数。

素数是指在大于1的自然数中，除了1和它自身外，不能被其他自然数整除的数。

以下已经给出部分代码的程序框架，其中，变量isPrime用来标记当前的数i是否为素数。请根据上述要求补全代码：

```
#include <stdio.h>

int main() {
    int i, j;
    printf("1到100之间的素数有: \n");
    for (i = 2; i <= 100; i++) {
        // 初始化isPrime
        int isPrime = 1;
        for (j = 2; j < i; j++) {
            // 判断i是否能被j整除，如果能整除，更新isPrime的值，并跳出内层循环
            if (i % j == 0) {
                isPrime = 0;
                break;
            }
        }
        // 打印出素数
        if (isPrime) {
            printf("%d ", i);
        }
    }
    printf("\n");
    return 0;
}
```