



知识点多总结





- **for循环**: 常用于已知循环次数的情况。

语法格式：

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

- **初始化表达式**: 用于初始化循环控制变量，只在循环开始时执行一次。

(**循环控制变量**: 用于控制循环执行次数和条件的变量)。

- **条件表达式**: 用于判断是否继续循环，如果条件为真，就执行循环体。如果条件为假，就结束循环。

- **更新表达式**: 在每次执行完循环体后执行，通常用于修改循环控制变量的值。



执行流程：

- 1、执行**初始化表达式**
- 2、判断**条件表达式**，如果条件为真，则执行循环体中的语句，然后执行更新表达式。
- 3、一直重复第2步，直到条件为假，结束循环。

例：

```
for (int i = 0; i < 5; i++) { // 当 i < 5 时继续循环  
    // 循环体语句  
    printf("%d ", i);  
}
```

对应的输出结果为0 1 2 3 4



·循环里的控制语句：

1、break语句：在循环里，break语句用于跳出当前的循环。

例：

```
for (int i = 0; i < 10; i++) { // 当 i < 10 时继续循环,  
直到 i >= 10 时结束循环  
    if (i == 5) {  
        break; // 当 i 等于 5 时, 跳出循环  
    }  
    printf("%d ", i);  
}
```

对应的输出结果为0 1 2 3 4



2、**continue语句**：用于结束本次循环，跳过位于**continue语句**之后的尚未执行的语句，直接开始下一次循环。

在**for**循环中，执行**continue**后，先执行更新表达式，然后再判断条件表达式。

在**while**和**do while**循环中，执行**continue**后，会直接判断条件表达式。所以需要在适当的位置，手动更新循环变量的值，避免进入死循环。

例：

```
for (int i = 0; i < 10; i++) {  
    if (i == 5) {  
        continue;  
    }  
    printf("%d ", i);  
}
```

对应的输出结果：0 1 2 3 4 6 7 8 9

```
int i = 0;  
while (i < 10) {  
    if (i == 5) {  
        i++; // 手动更新循环变量i的值  
        continue;  
    }  
    printf("%d ", i);  
    i++; // 更新循环变量i的值  
}
```

对应的输出结果：0 1 2 3 4 6 7 8 9



·注意：

1、for后面括号里的三个表达式都可以为空，但分号不能省略。如果条件表达式为空，默认为真，会进入死循环，除非在循环体中有控制跳出的语句。

例1：

```
int i = 0;  
for ( ; ; ) { // 条件表达式为空，默认为真  
    // 循环体里没有控制跳出的语句，会进入死循环  
    printf("%d ", i++);  
}
```

例2：

```
int i = 0;  
for ( ; ; ) { // 条件表达式为空，默认为真  
    printf("%d ", i++);  
    if (i == 2) {  
        // 循环体里有控制跳出的语句，不会进入死循环  
        break;  
    }  
}
```

对应的输出结果：0 1



2、循环控制变量可以在初始化表达式中定义，也可以在外部提前定义。

例：

```
for( int i = 0; i < 5; i++) {  
    printf("%d ", i);  
}
```

或

```
int i;  
for (i = 0; i < 5; i++) {  
    printf("%d ", i);  
}
```



题目1、有以下程序：

```
#include <stdio.h>

main() {
    int i, a;
    for(i = 0; i <= 10; i++)
        a = i;
    printf("%d,%d\n", i, a);
}
```

程序的运行结果是 (A)

- A. 11,10
- B. 10,10
- C. 10,11
- D. 11,11



题目2、有以下程序：

```
#include <stdio.h>
main() {
    int a = 1, b = 2;
    for(; a < 8; a++) {
        b += a;
        a += 2;
    }
    printf("%d,%d\n", a, b);
}
```

程序运行后的输出结果是 (D)

- A. 9,18
- B. 8,11
- C. 7,11
- D. 10,14



题目3、有如下程序：

```
#include <stdio.h>
main() {
    char a = '3', b = 'A';
    int i;
    for (i = 0; i < 6; i++) {
        if(i % 3) putchar(a + i);
        else putchar(b + i);
    }
    printf("\n");
}
```

程序运行后的输出结果是 (A)

- A. A45D78
- B. ABC678
- C. 34CD78
- D. 34AB78



題目4、有以下程序：

```
#include <stdio.h>
main() {
    int a = 1, b = 0;
    for(; a < 5; a++) {
        if(a % 2 == 0) break;
        b+=a;
    }
    printf("%d\n", b);
}
```

程序的运行结果是 (A)

- A. 1
- B. 10
- C. 0
- D. 4



题目5、有以下程序：

```
#include <stdio.h>
main() {
    int x = 8;
    for(; x > 0; x--) {
        if(x % 3) {
            printf("%d,", x--);
            continue;
        }
        printf("%d,", --x);
    }
}
```

程序的运行结果是 (D)

- A. 7,4,2,
- B. 8,7,5,2,
- C. 9,7,6,4,
- D. 8,5,4,2,



题目6、有以下程序

```
#include <stdio.h>
main() {
    int a = 1, b = 0;
    for ( ; a < 5; a++) {
        if (a % 2 == 0) break;
        continue;
        b += a;
    }
    printf("%d \n", b);
}
```

程序运行后的输出结果是 (A)

- A. 0
- B. 1
- C. 10
- D. 4