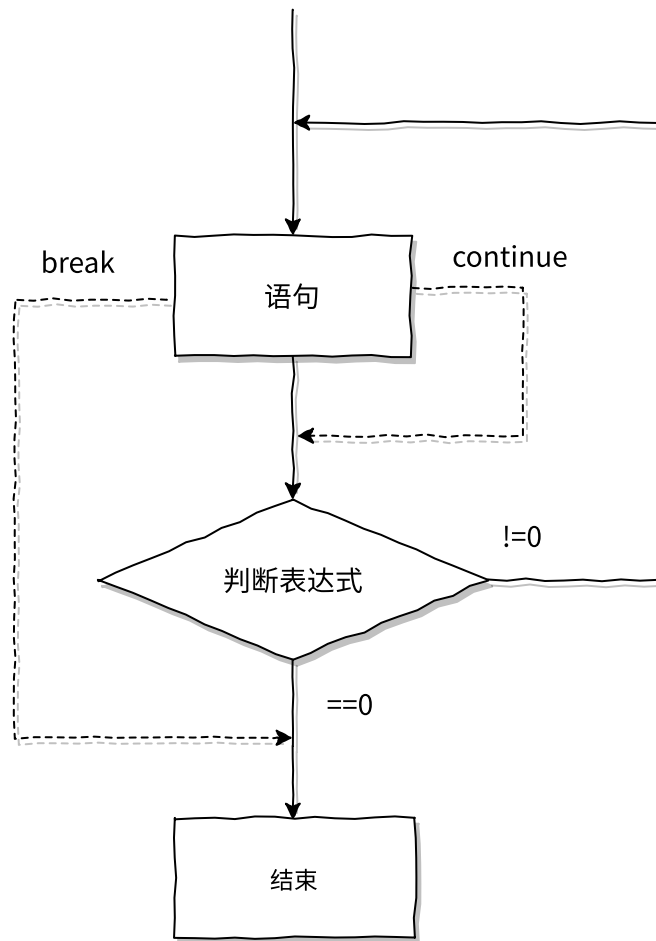




43.do while语句中的break和continue

在上一小节学习do while循环的执行流程的时候，有细心的同学一定看到了图中的break和continue，那在do while循环中break和continue有什么作用呢？



do while 循环的执行流程

1. break

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

1 2 3 4

D:\code\2022\test\test_6_3\Debug\test_6_3.exe (进程 17272) 已退出，代码为 0。
按任意键关闭此窗口. . .

break的作用在循环中都是一样的，只要代码中有机会执行到break，循环就直接终止了，所以在do while循环中break的作用也是永久的终止循环，不管循环还可能要循环多少次，都直接结束了。

2. continue

练习：

如果把上面代码中的break换成continue会怎么样呢？

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

1 2 3 4

这次代码执行打印完1~4的数字后也是陷入了死循环，为什么呢？

还是因为i==5的时候，执行了continue语句，continue跳过了本次循环中continue后边橙色的代码，i没能改变，在判断时候恒为真，就死循环了。

所以continue的作用也是跳过本次循环后边的代码，和在while循环中的continue非常相似，但是有for循环的略有差异。

总之：学完了三种循环中的break和continue两个语句，在循环中使用是用来改变程序的执行流程的，了解清楚语句的特点，在适合的场景下使用，才能达到想要的效果。

完