

小马加编信息学教案(六)

C++分支结构

- 一. 课程内容
- 二. 知识讲解
 - 1. if 语句的使用
 - * 1.1 if 语句的格式
 - * 1.2 if 语句的复合
 - 2. if语句的嵌套
 - * 2.1 if 语句的嵌套
 - * 2.2 if 语句的嵌套使用实例
 - * 2.3 if 语句嵌套使用注意
- 三. 经典例题
- 四. 提高巩固

一. 课程内容

1. C++ if 语句的使用
2. C++ if 语句的嵌套

二. 知识讲解

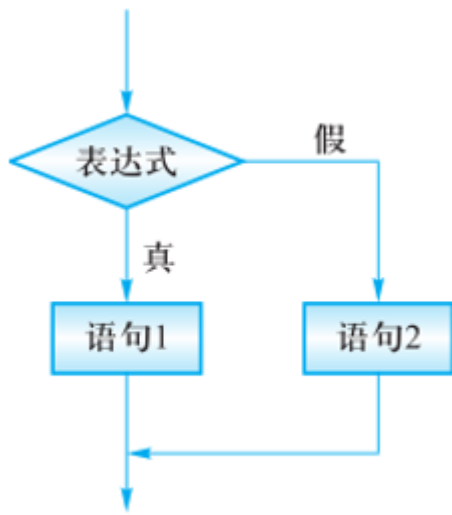
1. if 语句的使用

1.1 if 语句的格式

if 语句有两种格式

- 第一种格式为：

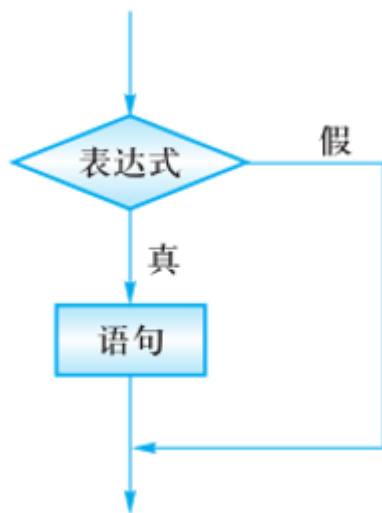
```
if (表达式)  
    语句;
```



- 第二种格式为：

```

if (表达式)
    语句1;
else
    语句2;
    
```



1.2 if 语句的复合

注意，如果 if 语句之内想要进行多条语句，则需要使用花括号将多条语句括起来

ex:

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int a = 1;
    if(a == 1) {
        cout << "a is 1" << endl;
        cout << "a is ok" << endl;
    }
    return 0;
}
```

2. if语句的嵌套

2.1 if 语句的嵌套

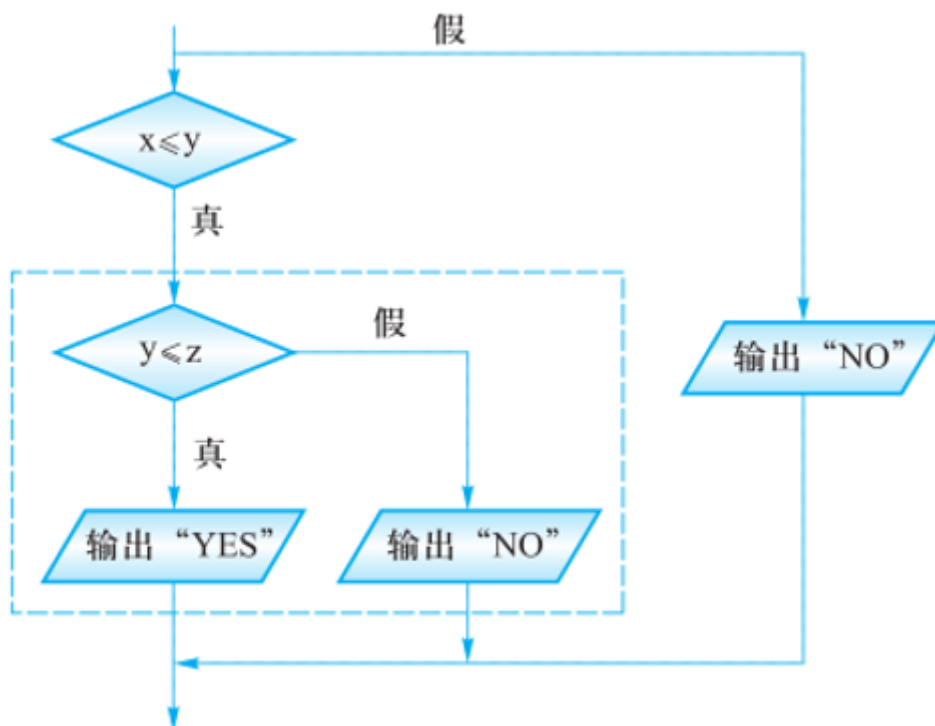
在分支结构中经常会遇到这样的情况：

如果条件1成立，就去做A事情，否则去做B事情，但是，在做A事情（或者B事情）时，还要根据条件2是否成立，决定是做A1（或者B1）这件子事情，还是去做A2（或者B2）这件子事情。

在程序设计中，把这种情况称为分支结构的“嵌套”，一般用来解决三种及以上的分支情况。

2.2 if 语句的嵌套使用实例

- 输入三个整数 x 、 y 、 z ，判断其是否满足 $x \leq y \leq z$ 。
- 若满足，输出 YES；否则，输出 NO



```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int x,y,z;
    cin >> x >> y >> z;
    if(x <= y)
        if(y <= z)
            cout << " YES " << endl;
        else
            cout << " NO " << endl;
    else
        cout << " NO " << endl;
    return 0;
}
```

2.3 if 语句嵌套使用注意

- 分支结构可以嵌套很多层
- if 与 else 的配对关系;

else 永远与其上方最近的未被匹配的 if 配对
if-else 的配对与缩进无关，注意区分！

- 编程要形成良好的代码风格

三. 经典例题

1. 阅读下列程序，写出程序结果，并上机编程检验答案

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int x = 4, y = 3, z = 2;
    if(x <= y)
        if(y <= z)
            cout << "YES" << endl;
    else
        cout << "NO" << endl;
    return 0;
}
```

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int x = 5, y = 2;
    if(x && y) {
        if(y == 3)
            cout << "HaHa" << endl;
        else
            cout << "PaPa" << endl;
    }
    else
        cout << "LaLa" << endl;
    return 0;
}
```

2. 旅游费用

风华公司“南京一日游”的收费标准为：
5人以内（含5人）按散客标准，每人300元。超过5人，按团体标准，每人 280 元。
请根据输入的人数，输出其旅游费用。

输入格式

一行一个正整数 n ，表示人数。

输出格式

一行一个正整数，表示需要的总旅游费用。

样例输入	样例输出
9	2520
2	600

数据范围

$1 \leq n \leq 5000000$

3. 三角形判断

输入三角形的三边长，判断是否能组成一个三角形
如果可以，输出 Yes ，不然输出 No

样例输入	样例输出
3 3 3	Yes
1 2 4	No

四. 提高巩固

1. 奇偶数判断

给定一个整数，判断该数是奇数还是偶数。
如果是奇数，输出 `odd` ；如果是偶数，输出 `even`

输入格式

输入仅一行，一个大于零的正整数 n 。

输出格式

输出仅一行，如果 n 是奇数，输出 `odd` ；如果 n 是偶数，输出 `even` 。

样例输入	样例输出
5	odd
4	even

2. 判断闰年

输入某一年份 x ，判断该年份是否是闰年。
是则输出 `YES` ，否则输出 `NO`

输入格式

一行一个正整数 x

输出格式

一行一个单词，表示判断结果。

样例输入	样例输出
2016	Yes
2100	No

数据范围

x 在 `int` 范围以内

3. 自制计算器

制作一个简单的计算器，支持 `+` `-` `*` `/` 四种运算。仅需考虑输入输出为整数的情况，数据和运算结果不会超过`int`表示的范围。

输入格式

输入只有一行，共有三个参数，其中第1、2个参数为整数，第3个参数为操作符 (`+`, `-`, `*`, `/`) 。

输出格式

输出只有一行，一个整数，为运算结果。

如果出现除数为0的情况，则输出： Divided by zero!

如果出现无效的操作符(即不为 + - * / 之一) ，则输出： Invalid operator!

样例输入	样例输出
1 2 +	3
2 0 /	Divided by zero!