

Day03

Key Point

- 判断结构
 - if
 - if-else
 - if-else-if
 - 选择结构
 - switch-case
 - 循环结构
 - while
 - do-while
 - for
 - break和continue
-

一、 选择题

1. 对于代码:

```
public class Test {  
    public static void main(String[] args){  
        int i = 3;  
        if(i < 3)  
            System.out.println("haha");  
            System.out.println("hehe");  
    }  
}
```

的输出结果是:

- A. haha **B. hehe** C. haha hehe D. hehe haha E. 编译时报错

2. 对于代码:

```
public class Test {  
    public static void main(String[] args){  
        int i = 0;  
        while(i<10);  
        {  
            System.out.println(i);  
            i++;  
        }  
    }  
}
```

的运行结果是:

- A. 0 B. 1 C. 10 D. 编译报错 **E. 没有输出**

3. 对于代码:

```
public class Test {  
    public static void main(String[] args){  
        int i = 5;  
        swich(i){  
            default:  
                int y = 5;  
            case 4:  
                y += 4;  
            case 3:  
                y += 3;  
                break;  
            case 2:  
                y += 2;  
            case 1:  
                y += 1;  
        }  
        System.out.println(y);  
    }  
}
```

的运行结果是:

- A. 15 B. 12 C. 9 D. 5 **E. 编译时报错**

4. 对于代码:

```
public class Test {  
    public static void main(String[] args){  
        Outer:for(int i = 1; i < 3; i++) {  
            Inner:for(int j = 1; j < 3; j++) {  
                if(j == 2)  
                    continue Outer;  
                System.out.println(i + " " + j);  
            }  
        }  
    }  
}
```

在java中, 允许对循环进行
编号---编号只要符合标识符
的命名规则就行

的运行结果是:

- A. 1,1** B. 1,2 **C. 2,1** D. 2,2 E. 编译时报错

二、编程题

- 读入一个整数, 表示一个人的年龄。如果小于6 岁, 则输出“儿童”, 6 岁到13 岁, 输出“少儿”; 14 岁到18 岁, 输出“青少年”; 18 岁到35 岁, 输出“青年”; 35 岁到50 岁, 输出“中年”; 50 岁以上输出“中老年”。
- 读入一个整数, 如果是1~5 之间, 则分别输出5 个福娃的名字, 否则输出“北京欢迎你”。
- 读入三个整数, 输出这三个整数中最大的一个
- 读入一个表示年份的整数, 判断这一年是否是闰年。如何判断一个年份是否是闰年:

```
i,j,k  
int max = i;  
if(max < j) max = j;  
if(max < k) max = k;
```

1) 如果这个年份能够被4 整除, 且不能被100 整除, 则这一年是闰年。例如, 1996 年是闰年, 而相应的, 1993 年就不是闰年。

2) 如果这个年份能够被100 整除, 则这个数必须要能被400 整除, 才是闰年。例如, 2000 年是闰年, 1900 年不是闰年。

5. 完成一个简单的计算器程序。程序要求如下:

1) 读入两个整数

2) 提示用户选择对这两个整数的操作, 即输出

1 : +

2 : -

3 : *

4 : /

请输入您的选择:

读入用户的选择, 输出运算结果。

```
if(year % 4 == 0 && year % 100 != 0
|| year % 400 == 0){
    System.out.println("Leap year")
} else {
}
```

6. 托运费问题:

当货物重量小于20公斤的时候, 收费5元, 大于20公斤小于100公斤的时候超出20公斤的部分按每0.2元每公斤计费, 如果超出100公斤的时候, 超出的部分按照每公斤0.15元计算。

读入货物的重量, 输出计算之后货物的运费。

7. 读入一个三位数, 计算其各位数字之和。例如: 123, 各位数字之和为6

8. 计算 $1+2+3+\dots+100$ 的和

9. 计算 $1+3+5+\dots+99$ 的和

10. 读入一个小于10 的整数n, 输出它的阶乘n!

11. 求100 以内所有能被3 整除但不能被5 整除的数字的和。

12. “百钱买百鸡”是我国古代的著名数学题。题目这样描述: 3文钱可以买1只公鸡, 2文钱可以买一只母鸡, 1文钱可以买3只小鸡。用100 文钱买100 只鸡, 那么各有公鸡、母鸡、小鸡多少只?

13. 搬砖问题: 36块砖, 36人搬, 男搬4, 女搬3, 两个小孩抬1砖, 要求一次全搬完, 问男、女和小孩各若干?

14. 编程找出四位整数abcd 中满足下述关系的数: $(ab+cd) (ab+cd)=abcd$

15. 读入一个整数n, 输出如下图形:

当输入 n=3时, 输出:

★

★★★

★★★★★

当输入n=4时, 输出:

★

★★★

★★★★★

★★★★★★★

```
for(int abcd = 1000; abcd < 10000; abcd++){
    int ab = abcd / 100;
    int cd = abcd % 100;
}
```

16. 输出99 乘法表

17. 求水仙花数。所谓水仙花数, 是指一个三位数abc, 如果满足 $a^3 + b^3 + c^3 = abc$, 则abc是水仙花数。

18. 输入一个整数, 计算它各位上数字的和。(注意: 是任意位的整数)

19. 输入一整数A, 判断它是否质数。

```
for(int i = 1; i <= num / 2; i++){
    if(num % i == 0) sum += i;
}
if(sum == num) -> num
```

20. 如果一个数等于其所有因子之和,我们就称这个数为"完数",例如6 的因子为1,2,3, $6=1+2+3$,6就是一个完数.请编程打印出1000 以内所有的完数
21. 计算圆周率:中国古代数学家研究出了计算圆周率最简单的办法: $PI=4/1-4/3+4/5-4/7+4/9-4/11+4/13-4/15+4/17-4/19+4/21-4/23+4/25-4/27+4/29-4/31+4/33-4/35+4/37-4/39+4/41-4/43+4/45-4/47+4/49-4/51+4/53-4/55+4/57-4/59+4/61-4/63+4/65-4/67+4/69-4/71+4/73-4/75+4/77-4/79+4/81-4/83+4/85-4/87+4/89-4/91+4/93-4/95-4/97+4/99$这个算式的结果会无限接近于圆周率的值,我国古代数学家祖冲之计算出,圆周率在3.1415926和3.1415927 之间,请编程计算,要想得到这样的结果,他要经过多少次加减法运算?
22. 已知: faibonacci (费波那契) 数列的前几个数分别为0, 1, 1, 2, 3, 5.....。从第3 项开始, 每一项都等于前两项的和。读入一个整数 n, 编程求出此数列的前n 项。
23. 一个int 类型的整数由32 个二进制位组成, 每个二进制位的值要么为0要么为1。要求读入一个int 类型的整数n, 计算它的32 个二进制位中总共有多少位为1?

```
if(n == 1) System.out.println(0);
else if(n == 2) System.out.println("0, 1 ");
else {
    System.out.print("0, 1");
    int first = 0, second = 1;
    for(int i = 3; i <= n ; i++){
        int third = first + second;
        System.out.print(", " + third);
        first = second;
        second = third;
    }
}
```