

3.1 本章要点

Java 语言支持结构化程序设计中规定的三种基本控制结构，即顺序结构、分支结构和循环结构。

顺序结构比较简单，其执行过程是从所描述的第一个操作开始，按顺序依次执行后续的操作，直到序列的最后一个操作。

分支结构可以通过 `if-else` 和 `switch` 实现。`if` 语句属于条件分支语句。`if` 语句的语法有以下两种：

```
if(booleanExp){           // 单分支结构
    statement(s)
}

if(booleanExp){           // 双分支结构
    statement(s)
}else{
    statement(s)
}
```

如果 `if` 或 `else` 块中只有一条语句，大括号就可以省略，具体如下：

```
if (a > 3)
    a ++;
else
    a = 3;
```

但是这样可能导致 `else` 悬空问题。考虑下面的示例：

```
if(a > 0)
    if(a > 10)
        System.out.println("a > 10");
else
    System.out.println("a < 0");
```

这里，`else` 语句悬空了。因为 `else` 语句与哪个 `if` 语句相关不清楚。当然，`else` 语句总是与离它最近的前一个 `if` 语句相关。如果使用大括号就可以使代码一目了然。

```
if(a > 0){
    if(a > 10){
        System.out.println("a > 10");
    }
}
```

```
    }else{
        System.out.println("a <= 10");
    }
}
```

switch 结构用来实现多分支，语法如下：

```
switch(expression){
    case value:statements; break;
    case value:statements; break;
    ...
    default:statements
}
```

expression 的类型可以是 byte、short、int、char、enum、String 等类型，value 是上述类型的字面值或一个枚举值。

循环结构包括 while 循环、do-while 循环、for 循环和增强的 for 循环。

while 语句的语法如下：

```
while(booleanExpr){
    statement(s)
}
```

do-while 语句的语法如下：

```
do{
    statement(s)
}while(booleanExpr);
```

大括号中是循环体。while 循环体可能一次也不执行，do-while 循环体至少执行一次。如果循环体只有一条语句，可以省略大括号，但为了清晰起见，建议始终使用大括号。

for 循环的语法如下：

```
for(init; booleanExpr; iterate){
    statement(s)
}
```

for 循环是 4 种循环中最复杂、最灵活的循环。在初始部分 init 和迭代部分 iterate 可以使用逗号运算符，for() 语句中的 3 个部分都可以省略，但括号和分号不能省略。

```
for(int i = 0, j = 10; j > 5 && i < 3; i++, j--){
    System.out.println(i + " " + j);
}
```

在 while 循环、do-while 循环和 for 循环中都可以使用 break 语句或 continue 语句。break 语句用来结束循环，跳出循环体。continue 语句仅结束当前一次迭代，控制转到下一次迭代开始。在 Java 中还支持带标签的 break 语句和 continue 语句，它们用来实现其他语言的 goto 语句的功能。

分支结构和循环结构可以相互嵌套。增强的 for 循环在第 5 章讨论。

3.2 实验指导

【实验目的】

1. 掌握分支结构程序设计。
2. 掌握循环结构程序设计。

【实验内容】

实验题目 1: 编写程序, 从键盘输入 a、b、c、d 四个整数, 计算 $(a+b)/(c-d)$ 的值, 如果 $c-d$ 的值为 0, 应输出错误信息。

实验题目 2: 身体质量指数 (Body Mass Index, BMI) 是衡量一个人是否超重的指标。计算公式为 $BMI = \text{体重} / \text{身高的平方}$, 体重单位为公斤, 身高单位为米。

对于一个成年人的 BMI 值的含义如下:

- 小于 16, 表示严重过轻;
- 16~18, 表示过轻;
- 18~24, 表示体重适中;
- 24~29, 表示过重;
- 29~35, 表示肥胖;
- 大于 35, 表示非常肥胖。

编写程序, 从键盘上输入体重 (单位: 公斤) 和身高 (单位: 米), 输出体重在什么范围。

实验题目 3: 编程打印输出 Fibonacci 数列的前 20 个数。Fibonacci 数列是第一和第二个数都是 1, 以后每个数是前两个数之和, 用公式表示为 $f_1 = f_2 = 1, f_n = f_{n-1} + f_{n-2} (n \geq 3)$ 。

实验题目 4: 编写程序 PerfectNumber.java, 求出 1~1000 之间的所有完全数。完全数是其所有因子 (包括 1 但不包括该数本身) 的和等于该数。例如 $28 = 1 + 2 + 4 + 7 + 14$, 28 就是一个完全数。

提示: 求 n 的所有因子之和的部分代码如下。

```
int sum = 1;
for(int k = 2; k < n/2; k++){
    if(n%k == 0)    // k 是 n 的一个因子
        sum = sum + k;
}
```

实验题目 5: 编写程序 PrimeFactor.java, 从键盘读入一个整数, 计算并显示该整数的所有素数因子。例如, 输入整数为 120, 输出应为 2、2、2、3、5。

下面是部分代码:

```
do{
    for(int k = 2; k <= n; k++){
```

```

        if(n % k == 0){
            System.out.println(k);
            n = n / k;
            break;
        }
    }
}while(n != 1);

```

【思考题】

1. switch 结构中的表达式允许使用哪些类型？
2. while 循环结构和 do-while 循环结构有什么异同？

3.3 习题解析

1. 写出下面程序运行结果。

```

public class Foo{
    public static void main(String[] args){
        int i = 1;
        int j = i++;
        if((i > ++j) && (i++ == j)){
            i += j;
        }
        System.out.println("i = "+i+", j = "+j) ;
    }
}

```

【答】 i=2,j=2。

2. 给定下面代码段，问变量 i 可使用哪三种数据类型？（ ）

```

switch (i) {
    default:
        System.out.println("Hello");
}

```

- | | | |
|-----------|-----------|----------|
| A. char | B. byte | C. float |
| D. double | E. Object | F. enum |

【答】 A、B、F。

3. 给定下面程序段，求输出结果。（ ）

```

int i = 1, j = 0 ;
switch(i) {
    case 2: j += 6;

```

```

        case 4: j += 1;
        default: j += 2;
        case 0: j += 4;
    }
    System.out.println(j);

```

A. 6 B. 1 C. 2 D. 4

【答】 A。

4. 下面的程序有错误:

```

public class IfWhileTest {
    public static void main (String []args) {
        int x = 1, y = 6;
        if(x = y)
            System.out.println("Equal ");
        else
            System.out.println("Not equal ");
        while (y--) { x++ ; }
        System.out.println("x = " + x+ " y = " + y);
    }
}

```

若使程序输出下面结果, 应如何修改程序。

```

Not equal
x = 7 y = -1

```

【答】 if(x = y) 改为 if(x == y), while (y--)改为 while (y-->0)。

5. 下面程序段执行后, i、j 的值分别为 ()。

```

int i = 1, j = 10;
do{
    if(i++>--j) continue;
}while(i < 5);

```

A. i=6 j=5 B. i=5 j=5
 C. i=6 j=4 D. i=5 j=6

【答】 D。

6. 下面程序输出 2~100 之间的所有素数, 请填空。

```

public class PrimeDemo {
    public static void main(String[] args){
        int i = 0, j = 0;
        for(i = 2; i <= 100; i++){
            for(j = 2; j < i; j++){

```



```
public static void main(String[] args){
    Scanner sc = new Scanner(System.in);
    System.out.print("请输入第 1 个整数: ");
    int max = sc.nextInt();
    int min = max;
    for(int i =2;i <=10 ; i++){
        System.out.print("请输入第"+i+"个整数: ");
        int num = sc.nextInt();
        if (num >max) max = num;
        if(num < min) min = num;
    }
    System.out.println("max= "+max);
    System.out.println("min= "+min);
}
}
```

10. 求解“鸡兔同笼问题”：鸡和兔在一个笼里，共有腿 100 条，头 40 个，问鸡兔各有几只？

参考程序如下：

```
public class ChickenHare{
    public static void main(String[] args){
        int legs = 100;
        int heads = 40;
        int chick, hare;
        for(chick = 0;chick <= 50;chick++){
            for(hare=0;hare<=25;hare++){
                if((chick+hare)==40&&(chick*2+hare*4)==100)
                    System.out.println("chick="+chick+" ,hare="+hare);
            }
        }
    }
}
```

11. 从键盘输入一个百分制的成绩，输出五级制的成绩，如输入 85，输出“良好”，要求使用 switch 结构实现。

参考程序如下：

```
import java.util.Scanner;
public class GradeTest{
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);
        System.out.print("请输入成绩: ");
        double score = input.nextDouble() ;
        String grade = "";
        if(score >100 || score <0){
```

```

        System.out.println("输入的成绩不正确。");
        System.exit(0);    // 结束程序运行
    }else{
        int g = (int)score /10;
        switch(g){
            default: grade = "不及格"; break;
            case 10: case 9: grade = "优秀"; break;
            case 8: grade = "良好"; break;
            case 7: grade = "中等"; break;
            case 6: grade = "及格"; break;
        }
    }
    System.out.println("你的成绩为: " + grade);
}
}

```

12. 编写程序，从键盘输入一个整数，计算并输出该数的各位数字之和。

例如：

请输入一个整数：8899123

各位数字之和为：40

参考程序如下：

```

import java.util.Scanner;
public class DigitSum {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("请输入一个整数：");
        int n,sum = 0;
        n = input.nextInt();
        while(n > 0){
            sum = sum + n%10;
            n = n / 10;
        }
        System.out.println("各位数字之和为： " + sum);
    }
}

```

13. 假设大学的学费年增长率为 7.8%，编程计算多少年后学费翻一番？

参考程序如下：

```

public class TuitionCal{
    public static void main(String[] args){
        double rate = 0.078;
        int n = 0;
        double sum = 1.0;
    }
}

```

```
        while(sum < 2){
            sum = sum + sum * rate;
            n = n+1;
        }
        System.out.println("year = " + n);
        System.out.println("tuition = " + sum);
    }
}
```

14. 编写程序, 求出所有的水仙花数。水仙花数是这样的三位数, 它的各位数字的立方和等于这个三位数本身, 例如 $371=3^3+7^3+1^3$, 371 就是一个水仙花数。

参考程序如下:

```
public class Narcissus{
    public static void main(String[] args){
        for(int i = 100; i < 1000; i++){
            int a = i %10;
            int b = (i / 10)%10;
            int c = i / 100;
            if(a*a*a+b*b*b+c*c*c==i)
                System.out.println(i);
        }
    }
}
```

15. 从键盘输入两个整数, 计算这两个数的最小公倍数和最大公约数并输出。

参考程序如下:

```
import java.util.Scanner;
public class GCDLCM{
    // 求最大公约数(Greatest Common Divisor)
    // 若  $x > y$ ,  $\text{gcd}(x,y)=\text{gcd}(x-y,y)$ 
    // 若  $x < y$ ,  $\text{gcd}(x,y)=\text{gcd}(x,y-x)$ 
    // 若  $x = y$ ,  $\text{gcd}(x,y)=x=y$ 
    public static int gcd(int x, int y){
        if (x > y)
            return gcd(x - y, y);
        if (x < y)
            return gcd(x, y - x);
        return x;
    }
    // 求最小公倍数 (Least Common Multiple)
    // 最小公倍数等于两数之积除以最大公约数
    public static int lcm(int x, int y){
        int gcd = gcd(x,y); // 求最大公约数
```

```

        return x*y / gcd;
    }
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.print("请输入正整数 x: ");
        int a = sc.nextInt();
        System.out.print("请输入正整数 y: ");
        int b = sc.nextInt();

        System.out.println("最大公约数=" + gcd(a,b));
        System.out.println("最小公倍数=" + lcm(a,b));
    }
}

```

16. 编写程序，求出 1~1000 之间的所有完全数。完全数是其所有因子（包括 1 但不包括该数本身）的和等于该数。例如， $28=1+2+4+7+14$ ，28 就是一个完全数。

参考程序如下：

```

public class PerfectNumber{
    public static boolean perfect(int n){
        int sum=1;
        for(int k=2;k<n;k++){
            if(n%k==0)
                sum=sum+k;
        }
        if(sum==n)
            return true;
        else
            return false;
    }
    public static void main(String args[]){
        for(int i = 1;i <= 1000; i++){
            if(perfect(i))
                System.out.println(i);
        }
    }
}

```

17. 编写程序读入一个整数，显示该整数的所有素数因子。例如，输入整数为 120，输出应为 2、2、2、3、5。

参考程序如下：

```

import java.util.Scanner;
public class PrimeFactor{
    public static void main(String[] args){
        Scanner sc =new Scanner(System.in);

```

```

        System.out.print("Input a number:");
        int n = sc.nextInt();
        do{
            for(int k=2;k<=n;k++){
                if(n%k==0){
                    System.out.println(k);
                    n=n/k;
                    break;
                }
            }
        }while(n!=1);
    }
}

```

18. 编写程序, 计算当 $n=10000, 20000, \dots, 100000$ 时 π 的值。求 π 的近似值公式如下。

$$\pi = 4 \times \left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \frac{1}{13} + \dots + \frac{1}{2n-1} - \frac{1}{2n+1} \right)$$

参考程序如下:

```

public class ComputePI{
    public static void main(String[] args){
        double pi;
        double sum;
        int n = 10000;
        int sign = 1;
        for(n=10000;n<=100000;n+=10000){
            sum = 0;
            for(int i = 1;i<= n;i++){
                sign = (i%2==0)?-1:1;
                sum =sum + sign*(1.0/(2*i-1));
            }
            pi = 4 * sum;
            System.out.println("n = " + n);
            System.out.println("PI = " + pi);
        }
    }
}

```

19. 编写程序, 计算贷款的每月支付额。程序要求用户输入贷款的年利率、总金额和年数, 程序计算月支付金额和总偿还金额, 并将结果显示输出。计算贷款的月支付额公式如下:

$$\frac{\text{贷款总额} \times \text{月利率}}{1 - \frac{1}{(1 + \text{月利率})^{\text{年数} \times 12}}$$

参考程序如下:

```
import java.util.Scanner;
public class ComputeLoan {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("请输入年利率: ");
        double annualRate = sc.nextDouble();
        double monthlyRate = annualRate / 1200; // 计算月利率
        System.out.print("请输入贷款总额: ");
        double loanAmount = sc.nextDouble();
        System.out.print("请输入年数: ");
        double numberOfYear = sc.nextDouble();

        double monthlyPayment = loanAmount*monthlyRate / (1 -
            1/Math.pow(1 + monthlyRate,numberOfYear * 12));
        double totalPayment = monthlyPayment*12*numberOfYear;
        System.out.printf("月支付额: %.2f%n",monthlyPayment);
        System.out.printf("总支付额: %.2f%n",totalPayment);
    }
}
```