## 第九章

### 9.3

package work5;  
  
import java.util.Date;  
  
public class work9\_3 {  
 public static void main(String[] args) {  
 long l = 10000;  
  
 Date date = new Date(l);  
 printDate(l, date);  
 l \*= 10;  
  
 for (int i = 0; i < 7; i++) {  
 date.setTime(l);  
 printDate(l, date);  
 l \*= 10;  
 }  
 }  
  
 // 用于显示日期  
 public static void printDate(long l, Date date) {  
 System.out.println(l + ": " + date.toString());  
 }  
}

### 9.4

package work5;  
  
import java.util.Random;  
  
public class work9\_4 {  
 public static void main(String[] args) {  
  
 Random random = new Random(1000);  
  
 for (int i = 0, n = 0; i < 50; i++, n++) {  
 if (n == 5) {  
 System.out.println("");  
 n = 0;  
 }  
  
 System.out.print(random.nextInt(100) + "\t");  
 }  
  
 }  
}

### 9.5

package work5;  
  
import java.util.GregorianCalendar;  
  
public class work9\_5 {  
  
 public static void main(String[] args) {  
  
 GregorianCalendar calender = new GregorianCalendar();  
 System.out.println("now: " + calender.get(GregorianCalendar.YEAR) + "." + (calender.get(GregorianCalendar.MONTH) + 1) + "." + calender.get(GregorianCalendar.DAY\_OF\_MONTH));  
  
 calender.setTimeInMillis(1234567898765L);  
 System.out.println("after set: " + calender.get(GregorianCalendar.YEAR) + "." + (calender.get(GregorianCalendar.MONTH) + 1) + "." + calender.get(GregorianCalendar.DAY\_OF\_MONTH));  
  
 }  
  
}

### 9.6

package work5;  
  
import java.util.Date;  
import java.util.Random;  
  
public class work9\_6 {  
 public static void main(String[] args) {  
 final int NUM = 100000;  
  
 int[] number = new int[NUM];  
 Random random = new Random();  
  
 for (int i = 0; i < NUM; i++)  
 number[i] = random.nextInt(100000);  
  
 StopWatch stopWatch = new StopWatch();  
 sort(number);  
 stopWatch.stop();  
  
 System.out.println(stopWatch.getElapsedTime() + "ms");  
  
 }  
  
 // 选择排序  
 public static void sort(int[] a) {  
 for (int i = 0; i < a.length; i++) {  
 int tmp = i;  
 for (int j = i + 1; j < a.length; j++) {  
 if (a[j] < a[tmp])  
 tmp = j;  
 }  
 int temp = a[tmp];  
 a[tmp] = a[i];  
 a[i] = temp;  
 }  
 }  
  
}  
  
class StopWatch {  
  
 private Date startTime;  
 private Date endTime;  
  
 public StopWatch() {  
 startTime = new Date();  
 }  
  
 public void start() {  
 startTime = new Date();  
 }  
  
 public void stop() {  
 endTime = new Date();  
 }  
  
 public long getElapsedTime() {  
 return endTime.getTime() - startTime.getTime();  
 }  
  
 public Date getStartTime() {  
 return startTime;  
 }  
  
 public Date getEndTime() {  
 return endTime;  
 }  
  
}

### 9.10

package work5;
  
  
import java.util.Scanner;
  
  
public class work9\_10 {
  
 public static void main(String[] args) {
  
 Scanner input = new Scanner(System.in);
  
 double a, b, c;
  
 System.out.print("请输入a,b,c：");
  
 a = input.nextDouble();
  
 b = input.nextDouble();
  
 c = input.nextDouble();
  
 QuadraticEquation qe = new QuadraticEquation(a, b, c);
  
 if (qe.isSolvable()) {
  
 System.out.println("x1 = " + qe.getRoot1());
  
 System.out.println("x2 = " + qe.getRoot2());
  
 } else {
  
 System.out.println("无实数根");
  
 }
  
 input.close();
  
 }
  
}
  
  
class QuadraticEquation {
  
 private double a, b, c;
  
  
 public QuadraticEquation(double a, double b, double c) {
  
 this.a = a;
  
 this.b = b;
  
 this.c = c;
  
 }
  
  
 public double getDiscriminant() {
  
 return b \* b - 4 \* a \* c;
  
 }
  
  
 public double getRoot1() {
  
 return (-b + Math.sqrt(getDiscriminant())) / (2 \* a);
  
 }
  
  
 public double getRoot2() {
  
 return (-b - Math.sqrt(getDiscriminant())) / (2 \* a);
  
 }
  
  
 public boolean isSolvable() {
  
 return getDiscriminant() >= 0;
  
 }
  
  
}

### 9.11

package work5;
  
  
import java.util.Scanner;
  
  
public class work9\_11 {
  
 public static void main(String[] args) {
  
 Scanner input = new Scanner(System.in);
  
 double a, b, c, d, e, f;
  
 System.out.print("请输入a,b,c,d,e,f：");
  
 a = input.nextDouble();
  
 b = input.nextDouble();
  
 c = input.nextDouble();
  
 d = input.nextDouble();
  
 e = input.nextDouble();
  
 f = input.nextDouble();
  
 LinearEquation le = new LinearEquation(a, b, c, d, e, f);
  
 if (le.isSolvable()) {
  
 System.out.println("x = " + le.getX());
  
 System.out.println("y = " + le.getY());
  
 } else {
  
 System.out.println("无实数根");
  
 }
  
 input.close();
  
 }
  
}
  
  
class LinearEquation {
  
 private double a, b, c, d, e, f;
  
  
 public LinearEquation(double a, double b, double c, double d, double e, double f) {
  
 this.a = a;
  
 this.b = b;
  
 this.c = c;
  
 this.d = d;
  
 this.e = e;
  
 this.f = f;
  
 }
  
  
 public double get\_a() {
  
 return a;
  
 }
  
  
 public double get\_b() {
  
 return b;
  
 }
  
  
 public double get\_c() {
  
 return c;
  
 }
  
  
 public double get\_d() {
  
 return d;
  
 }
  
  
 public double get\_e() {
  
 return e;
  
 }
  
  
 public double get\_f() {
  
 return f;
  
 }
  
  
 public double getX() {
  
 return (e \* d - b \* f) / (a \* d - b \* c);
  
 }
  
  
 public double getY() {
  
 return (a \* f - e \* c) / (a \* d - b \* c);
  
 }
  
  
 public boolean isSolvable() {
  
 return a \* d - b \* c != 0;
  
 }
  
  
}