**public** **class** calendar\_2008 {//输出2008年日历，要求有月份、日期与星期。统计并输//出2008年日期的个位数与对应的星期恰好相同的总天数

**public** **static** **void** main(String[] args) {

**int**[] date\_data = {2,5,6,2,4,7,2,5,1,3,6,1};

**int**[] space\_data = {1,4,5,1,3,6,1,4,0,2,5,0};

**int**[] days\_data = {31,29,31,30,31,30,31,31,30,31,30,31};

String s = "";

**for**(**int** i =0;i<12;i++) {

System.***out***.printf("\t\t%d月\n",i+1);

s += *ShowCalendar*(date\_data[i],space\_data[i],days\_data[i],i+1);

}

System.***out***.println(s);

}

**public** **static** String ShowCalendar(**int** date,**int** space,**int** days,**int** month) {

//date:初始星期数;space:开头空白天数;days:一个月的天数;month:月份

System.***out***.println("星期一\t星期二\t星期三\t星期四\t星期五\t星期六\t星期日\t");

String str = "";//记录日期的个位数与对应的星期恰好相同

**for**(**int** i=1;i<=days+space;i++) {

**if**(i <= space) {

System.***out***.printf("\t");

}**else** **if**(i == space+1){

System.***out***.printf("%d\t",1);

**if**(date<7) {

date += 1;

}**else** {

System.***out***.println('\n');

date = 1;

}

}**else** **if**(date < 7) {

System.***out***.printf("%d\t",i-space);

**if**(date == i-space) {

str += Integer.*toString*(month);str += "月";

str += Integer.*toString*(i);str += "日\n";

}**else** **if**(((i-space)%10) == date) {

str += Integer.*toString*(month);str += "月";

str += Integer.*toString*(i);str += "日\n";

}**else** {

str += "";

}

date += 1;

}**else** **if**(date == 7){

System.***out***.printf("%d\n",i-space);

**if**(date == i-space) {

str += Integer.*toString*(month);str += "月";

str += Integer.*toString*(i);str += "日\n";

}**else** **if**(((i-space)%10) == date) {

str += Integer.*toString*(month);str += "月";

str += Integer.*toString*(i);str += "日\n";

}**else** {

str += "";

}

date = 1;

}

}

System.***out***.println('\n');

**return** str;

}

}

**public** **class** Queen8 {//八皇后

**int** Max=8;//八个皇后

**static** **int** *Count*=0;//计数

**int**[] arry=**new** **int**[Max];

**public** **static** **void** main(String[] args) {

// new Queue8().Check(0);其中Check(0)代表从第一个开始检测

**new** Queen8().Check(0);

//统计合格种数 92种

System.***out***.println("一共有"+*Count*+"种");

}

//主要检查入口

**private** **void** Check(**int** n) {

//n=Max代表一行全部检测完毕

**if**(n==Max) {

print();

**return**;

}

**for** (**int** i = 0; i < Max; i++) {

arry[n]=i;

**if**(Judge(n)) {

Check(n+1);

}

}

}

//打印一行的数据

**private** **void** print() {

*Count*++;

**for** (**int** i = 0; i < Max; i++) {

System.***out***.print(arry[i]+" ");

}

System.***out***.println();

}

//判断是否同一列(arry[i]==arry[n])或者在对角线(Math.abs(n-i)==Math.abs(arry[n]-arry[i]))

**private** **boolean** Judge(**int** n) {

**for** (**int** i = 0; i < n; i++) {

**if**((arry[i]==arry[n])||(Math.*abs*(n-i)==Math.*abs*(arry[n]-arry[i]))) {

**return** **false**;

}

}

**return** **true**;

}

}

**public** **class** enum\_test {//枚举输出星期对应的名称

**public** **enum** WeekName{

***周一***,***周二***,***周三***,***周四***,***周五***,***周六***,***周日***

}

**public** **static** **void** main(String[] args) {

WeekName[] N = WeekName.*values*();

**for**(**int** i=0;i<N.length;i++){

System.***out***.println(N[i]);

}

}

}

**public** **class** random\_10000 {//运行random()函数计算//(int)(Math.random()\*20+0.5)10000次，统计0-20的个数分别是多少并输出

**public** **static** **void** main(String[] args) {

**int**[] Matchlist = {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20};

**int**[] result = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

**for**(**int** i=0;i<10000;i++) {

**int** num = (**int**)(Math.*random*()\*20+0.5);

**for**(**int** j=0;j<Matchlist.length;j++) {

**if**(num==Matchlist[j]) {

result[j]+=1;

}

}

}

**for**(**int** i=0;i<result.length;i++) {

System.***out***.println(result[i]);

}

}

}

**public** **class** FACT {//要求分别采用递归方法和非递归方法计算n的阶乘

**public** **static** **void** getFACT\_1(**int** n){

**int** sum = 1;

**for**(**int** i=1;i<=n;i++){

sum = sum\*i;

}

System.***out***.println(sum);

}

**public** **static** **long** getFACT\_2(**int** n){

**if**(n <= 1)

{

**return** 1;

}

**else**

{

**return** n \* *getFACT\_2*(n - 1);

}

}

**public** **static** **void** main(String[] args) {

**for**(**int** i=1;i<=10;i++){

*getFACT\_1*(i);

}

**for**(**int** i=1;i<=10;i++){

System.***out***.println(*getFACT\_2*(i));

}

}

}

**public** **class** MoneyMeans {//计算n由多少个1元、2元和5元组成

**public** **static** **void** Allmeans(**int** n)

{

**for**(**int** i=0;i<=n;i++)

**for**(**int** j=0;j<=n/2;j++)

**for**(**int** k=0;k<=n/5;k++)

**if**(i+2\*j+5\*k==n)

{ System.***out***.print(n+"可以由"+i+"个1元 "+j+"个2元 "

+k+"个5元组成");

System.***out***.println();

}

}

**public** **static** **void** main(String[] args) {

System.***out***.println("请输入金额n:");

Scanner input = **new** Scanner(System.***in***);

**int** n = input.nextInt();

*Allmeans*(n);

}

}

**public** **class** Draw\_I {//将大写字母I保存到data.txt中

**public** **void** SaveFile() **throws** IOException {

FileOutputStream f = **new** FileOutputStream("E:\\JavaStudy\\Test\_5\\src\\Draw\_I\\data.txt");

String str = "";

**for**(**int** i=0;i<10;++i) {

**if**(i==0||i==9) {

str += "\*\*\*\*\*\*\n";

}**else** {

str += " \*\* \n";

}

}

**byte**[] b = str.getBytes();

f.write(b);

f.close();

System.***out***.println("输入字母I完成！");

}

**public** **static** **void** main(String args[]) **throws** IOException {

Draw\_I i = **new** Draw\_I();

i.SaveFile();

}

}

**public** **class** GrandData{//保存五名学生成绩到data.txt，并计算平均值和排序输出

**public** **void** Output() **throws** IOException {

FileOutputStream fos = **new** FileOutputStream("E:\\JavaStudy\\Test\_5\\src\\GrandingData\\GrandData.txt");

String str = "";

**for**(**int** i=0;i<5;++i) {

System.***out***.printf("请输入第%d个成绩:",i+1);

Scanner input = **new** Scanner(System.***in***);

str += Float.*toString*(input.nextFloat());

**if**(i!=4) {

str += ",";

}

}

System.***out***.println("输入完成！");

**byte**[] b = str.getBytes();

fos.write(b);

fos.close();

}

**public** **void** InputAndAveragevalue() **throws** IOException {

FileInputStream in = **new** FileInputStream("E:\\JavaStudy\\Test\_5\\src\\GrandingData\\GrandData.txt");

String data = "";

**byte**[] bys = **new** **byte**[1024];

**if**(in.read(bys)!=-1) {

data = **new** String(bys);

}

String[] nums = data.split(",");

**float** sum = 0;

**float**[] g = **new** **float**[5];

**for**(**int** i=0;i<nums.length;i++) {

sum += Float.*parseFloat*(nums[i]);

g[i] = Float.*parseFloat*(nums[i]);

}

System.***out***.printf("平均值：%.1f\n",(sum/5.0));

Arrays.*sort*(g);

System.***out***.println("成绩排序：");

**for**(**float** i:g) {

System.***out***.println(i);

}

in.close();

}

**public** **static** **void** main(String[] args) **throws** IOException {

GrandData t = **new** GrandData();

t.Output();

t.InputAndAveragevalue();

}

}

**public** **class** SearchStr {//在当前路径下的所有文件中查找给定的字符串

**public** **static** **int** *mount* = 0;

**public** **static** **void** main(String[] args)

{

String filename = "E:\\JavaStudy\\Test\_5\\src\\SearchStr";

//创建一个 File 实例，表示路径名是指定路径参数的文件

File file = **new** File(filename);

System.***out***.println("请输入要查询的内容：");

Scanner input = **new** Scanner(System.***in***);

String in = input.nextLine();

**if**(in.length()>0) //输入要查找的关键字

{

*findFile*(file, in);

*print*(in);

}

input.close();

}

**public** **static** **boolean** isTrueFile(File file)

{

**if**(!file.exists() || !file.canRead())

**return** **false**;

**if** (file.getName().startsWith("."))

**return** **false**;

**if** (file.getName().endsWith("."))

**return** **false**;

**return** **true**;

}

**public** **static** **void** findFile(File file, String word)

{

File[] listFiles = file.listFiles();

//得到一个File数组，它默认是按文件最后修改日期排序的

**for** (**int** i = 0; i < listFiles.length; i++)

{

**if** (listFiles[i].isDirectory())

*findFile*(listFiles[i], word);

**else** **if** (*isTrueFile*(listFiles[i]))

*search*(listFiles[i], word);

}

}

**public** **static** **void** search(File file, String word)

{

**try**

{

**int** j = 0, k = 0, ch = 0;

String str = **null**;

FileReader in = **new** FileReader(file);

**while** ((ch = in.read()) != -1)

{

str += (**char**) ch;

}

**if** (str != **null**)

{

**while** (str.indexOf(word, j) != -1)

{

k++;

j = str.indexOf(word, j) + 1; // 返回第一次出现的指定子字符串在此字符串中的索引

}

}

**if** (k > 0)

{

System.***out***.println("在" + file.getAbsolutePath() + "有" + k+ "个关键字");

*mount*++;

}

in.close();

}

**catch** (FileNotFoundException e)

{

e.printStackTrace();

}

**catch** (IOException e)

{

e.printStackTrace();

}

}

**public** **static** **void** print(String word)

{

**if** (*mount* != 0)

{

System.***out***.println("找到" + *mount* + "个文本包含关键字" + word + "!");

}

**else**

{

System.***out***.println("没有找到相应的文件");

}

}

}

**public** **class** Prime {//在文件“data.txt"中写入所有比n小的素数，最后通过控制台窗//口分别显示每个数字(从0到9)在这些素数(比n小的素数)中出现的总次数

**public** **static** **int**[] *result* = **new** **int**[1000];

**public** **static** **int** *flag* = 0;//记录素数个数

**public** **static** **int**[] *count\_num* = {0,1,2,3,4,5,6,7,8,9};//统计的数字

**public** **static** **int**[] *count\_sum* = {0,0,0,0,0,0,0,0,0,0};//统计的个数

**public** **void** GetPrime(**int** n) **throws** IOException {

**if**(n<=2) {

System.***out***.println("没有！");

}**else** {

**for**(**int** i=2;i<n;i++) {

**boolean** b = **true**;

**for**(**int** j=2;j<=Math.*sqrt*(i);j++) {

**if**(i%j==0) {

b = **false**;

**break**;

}

}

**if**(b) {

*result*[*flag*] = i;

*flag* += 1;

}

}

}

FileOutputStream fos = **new** FileOutputStream("E:\\JavaStudy\\Test\_5\\src\\ShowPrime\\PrimeData.txt");

String str = "";

**for**(**int** i=0;i<*flag*;++i) {

**int** num = *result*[i];

str += Integer.*toString*(num);

**if**(i!=(*flag*-1)) {

str += ",";

}

}

System.***out***.println("输入完成！");

**byte**[] b = str.getBytes();

fos.write(b);

fos.close();

}

**public** **void** GetCountSum() {//int转字符，然后分割再转int，最后按个比较

**for**(**int** i=0;i<*flag*;i++) {

String r\_str = Integer.*toString*(*result*[i]);

String[] num\_str = r\_str.split("");

**for**(String s:num\_str) {

**int** num = Integer.*parseInt*(s);

**for**(**int** j=0;j<10;j++) {

**if**(*count\_num*[j]==num)

*count\_sum*[j] += 1;

}

}

}

**for**(**int** i=0;i<10;i++) {

System.***out***.printf("%d个数:%d ",i,*count\_sum*[i]);

}

}

**public** **static** **void** main(String[] args) **throws** IOException {

System.***out***.println("请输入正整数n:");

Scanner input = **new** Scanner(System.***in***);

**int** n = input.nextInt();

Prime p = **new** Prime();

p.GetPrime(n);

p.GetCountSum();

}

}