


```

121         System.out.println(ans.setScale(3,
122             // ---
123             RoundingMode.HALF_UP));
124
125 // 1-13
126
127 package com.company;
128
129 import java.util.Scanner;
130
131 public class Main {
132
133     public static void main(String[] args) {
134         Scanner input = new Scanner(System.in);
135
136         double x1 = 0, y1 = 0, a1 = 0, x2 = 0, y2 =
137             0, a2 = 0;
138
139         x1 = input.nextDouble();
140         y1 = input.nextDouble();
141         a1 = input.nextDouble();
142         x2 = input.nextDouble();
143         y2 = input.nextDouble();
144         a2 = input.nextDouble();
145
146         // 克萊姆公式
147         // [a b][x] = [e]
148         // [c d][y] = [f]
149         // x = ed - bf / ad - bc
150         // y = af - ec / ad - bc
151
152         double x = 0, y = 0;
153         x = ( ( a1 * y2 ) - ( y1 * a2 ) ) / ( ( x1 *
154             y2 ) - ( y1 * x2 ) );
155         y = ( ( x1 * a2 ) - ( a1 * x2 ) ) / ( ( x1 *
156             y2 ) - ( y1 * x2 ) );
157
158         System.out.println(x + " " + y);
159     }
160 }
161 // ---
162 // Chapter 2-6
163
164 package com.company;
165
166 import java.util.Scanner;
167
168 public class Main {
169
170     public static void main(String[] args) {
171         Scanner input = new Scanner(System.in);
172         String str = input.nextLine();
173
174         // 先把字串內的每位數單獨拆分進 array 內
175         String[] array = str.split("");
176
177         Integer ans = new Integer(1);
178         for( int i = 0; i < str.length(); i++ ){
179
180             // 用 Integer.parseInt() 把 array[i]
181             // 內的字符轉數字
182             ans = Integer.parseInt(array[i]) * ans;
183         }
184
185         System.out.println(ans);
186     }
187 }
188 // ---
189 // 多筆測資輸入判斷
190
191 double x1 = 0, y1 = 0, x2 = 0, y2 = 0;
192 while(input.hasNext()){
193     x1 = input.nextDouble();
194     y1 = input.nextDouble();
195     x2 = input.nextDouble();
196     y2 = input.nextDouble();
197 }
198 // ---
199 // Chapter 2-15
200
201 import java.util.Scanner;
202
203 public class Main {
204
205     public static void main(String[] args) {
206         Scanner input = new Scanner(System.in);
207
208         double x1 = 0, y1 = 0, x2 = 0, y2 = 0;
209
210         // 讀取四個輸入
211         x1 = input.nextDouble();
212         y1 = input.nextDouble();
213         x2 = input.nextDouble();
214         y2 = input.nextDouble();
215
216         // 利用 Math.pow 和 Math.sqrt 做運算
217         System.out.println( Math.sqrt( Math.pow(x2 -
218             x1, 2) + Math.pow(y2 - y1, 2) ) );
219     }
220 }
221 // ---
222 // 2-19
223
224 import java.math.BigDecimal;
225 import java.math.RoundingMode;
226 import java.util.Arrays;
227 import java.util.Scanner;
228
229 public class Main {
230
231     private static Math Precision;
232
233     public static void main(String[] args) {
234         Scanner input = new Scanner(System.in);
235
236         double x1 = 0, y1 = 0, x2 = 0, y2 = 0, x3 =
237             0, y3 = 0;
238
239         x1 = input.nextDouble();
240         y1 = input.nextDouble();
241         x2 = input.nextDouble();
242         y2 = input.nextDouble();
243         x3 = input.nextDouble();
244         y3 = input.nextDouble();
245
246         double[] array = new double[3];
247         array[0] = Math.sqrt( Math.pow(x2 - x1, 2) +
248             Math.pow(y2 - y1, 2) );
249         array[1] = Math.sqrt( Math.pow(x3 - x2, 2) +
250             Math.pow(y3 - y2, 2) );
251         array[2] = Math.sqrt( Math.pow(x3 - x1, 2) +
252             Math.pow(y3 - y1, 2) );
253
254         // 判斷是否為三角形
255         // Arrays.sort(陣列名稱) -> 自動排序
256         Arrays.sort(array);
257
258         if ( array[0] + array[1] <= array[2] ){
259             System.out.println("Not Triangle");
260         }
261         else{
262             // 海龍公式
263             double s = 0;
264             for( int i = 0; i < 3; i++ ){
265                 s = s + array[i];
266             }
267             s = s / 2;
268
269             double a = 0;

```

```

265         a = Math.sqrt( s * (s-array[0]) *
266             (s-array[1]) * (s-array[2]) );
267         // 利用 BigDecimal 去取小數點並四捨五入
268         BigDecimal ans = new BigDecimal(a);
269         System.out.println( ans.setScale(3,
270             RoundingMode.HALF_UP));
271     }
272 }
273 }
274 }
275 // ---
276 - hypot
277
278 // primelist
279
280 package com.company;
281
282 import java.util.Arrays;
283 import java.util.Scanner;
284 import java.util.Vector;
285
286 public class Main {
287
288     public static void main(String[] args) {
289         Scanner input = new Scanner(System.in);
290         int n = input.nextInt();
291
292         // 建質數表
293         // java 中沒有全域變數
294         // 所以這邊建表目前是每次輸入都重建一次
295         // 但不夠漂亮
296         boolean[] prime = new boolean[10000000+5];
297
298         // 用 vector 存質數們
299         Vector<Integer> v = new Vector<Integer>();
300         for( int i = 2; i <= n; i++ ){
301             if(!prime[i]){
302                 v.add(i);
303                 for( int j = i * i; j <= n; j += i ){
304                     prime[j] = true;
305                 }
306             }
307         }
308
309         for( int i = 0, k = 0; i < v.size(); i++, k++ )
310         {
311             if( k == 9 || i == v.size() - 1 ){
312                 System.out.println(v.get(i));
313                 // 把 k 歸回去 -1 是因為下次輪迴會先
314                 // k++
315                 k = -1;
316             }
317             else{
318                 System.out.print(v.get(i) + " ");
319             }
320         }
321         v.clear();
322     }
323 }
324 // ---
325 // 3-11
326
327 package com.company;
328
329 import java.util.Arrays;
330 import java.util.Scanner;
331 import java.util.Vector;
332
333 public class Main {
334
335     public static void main(String[] args) {
336         Scanner input = new Scanner(System.in);
337         int month = input.nextInt();
338         int year = input.nextInt();
339
340         // 分別存 30 天的月份 和 31 天的月份
341         // 以及他們的英文月份名
342         int[] thirty = {1,3,5,7,8,10,12};
343         String[] thirty_eng = { "January", "March",
344             "May", "July", "August", "October",
345             "December"};
346
347         int[] thirty_one = {4,6,9,11};
348         String[] thirty_one_eng = { "April", "June",
349             "September", "November"};
350
351         // 查詢他是否是 30 天的月份
352         // 最後放 return 後面的其他程式碼就不會執行了
353         for( int i = 0; i < thirty.length; i++){
354             if(month == thirty[i]){
355                 System.out.println(thirty_eng[i] + "
356                     " + year + " had 31 Days" );
357                 return;
358             }
359         }
360         for( int i = 0; i < thirty_one.length; i++){
361             if(month == thirty_one[i]){
362                 System.out.println(thirty_one_eng[i]
363                     + " " + year + " had 31 Days" );
364                 return;
365             }
366         }
367         if( year % 4 == 0 ){
368             System.out.println("February " + year + "
369                 had 29 Days");
370             return;
371         }
372         System.out.println("February " + year + " had
373             28 Days");
374     }
375 }
376 // ---

```