## Contents

## 1 Others

## 1.1 java

```
1 // Java
2
3 // 基礎運算判斷
  // java 中要用 .hasNext() 來判斷是否輸入結束
6 while( input.hasNext() )
8 // java 用 ==
      進行比較的時候,比較的是他們在記憶體中的存放地址
9 // 而 String, Integer, Date 這些類中 equals
      可以用來做比較
10 if( str.equals("0") ){
11
      break;
12 }
13 // ---
14
15 // Scanner
16
17 // String 的 input 是 .nextLine()
18 String str = input.nextLine();
20 // int 的 input 是 .nextInt()
21 int n = input.nextInt()
22
23 // boolean 的 input 是 .nextBoolean()
24 boolean flag = input.nextBoolean()
25 // ---
26
27 // vector
28
29 // java 的 vector 寫法似乎是如此 :D
  |// <> 內不能寫 int 要寫 Integer
30
31 Vector < Integer > v = new Vector();
32
      // 像 C++ 的 v.push_back(i)
33
34
      v.add(i);
35
36
      for( i = 0; i < v.size()-1; i++ ){</pre>
37
38
          // 不能直接寫 v[i] 要寫 v.get(i)
39
40
          System.out.printf( "%d ", v.get(i) );
41
42
      System.out.printf( "%d\n", v.get(i) );
43
44
45
     memset
46
  // Arrays.fill 類似 C++ 的 memset(a, false,
47
      sizeof(a) );
48 Arrays.fill( prime, false );
49 // ---
50
51 // Math
52
53 // java 的 abs 前面要加 Math.
54 if( Math.abs( first - second ) % 11 == 0 ){
      System.out.printf("%s is a multiple of 11.\n",
          str );
56 }
57
  // ---
58
     變數型態轉換
```

```
for( int i = 0; i < array.length; i++ ){</pre>
 61
 62
               // java 的 string to double 寫法
               tmp[i] = Double.parseDouble(array[i]);
1 63
 64
            }
 65
 66
    // str.split
 67
 68
    // 當字串中帶有 + * / \ 等符號時要寫 \\
        轉義,因為他們在正則表達示中有相應的不同意義
            // limit is 0; array contains all substrings
 70
            // Positive Lookahead or Lookbehind 觀念
 71
               "((?=@)|(?<=@))" -> 才會把 運算子 還有
 72
                運算元 都單獨切開
  73
            String[] array =
                str.split("((?=\\+|-|\\*|/|\\)|\\(|%)|(?<=\\+|-|\\*
 74
 75
            // 預設 ans 賦值 可控制小數點位數 ex 0.000 ->
                小數點後三位
  76
            BigDecimal ans = new BigDecimal("0.000");
 77
 78
 79
    // StringTokenizer
 80
    // 取運算子的部分用 StringTokenizer 更為方便
 81
 82
            StringTokenizer st = new
                StringTokenizer(str, "0123456789.");
            boolean flag = true;
 83
            while(st.hasMoreTokens()){
 84
               System.out.print( flag ? st.nextToken() :
 85
                    " " + st.nextToken());
               flag = false;
 86
            }
 87
 88
 89
 90
    // string.contains()
 91
 92
    boolean flag = true;
            for( String check : array){
 93
                // string.contains
                    一次查詢多個關鍵字的寫法
                    要另外存一個數組 token
                // 且在查詢時要以此 for ( String check :
 95
                    array ) 去做掃描
               if( !token.contains(check) ){
 96
                   // 將字串型態的數字轉成實數並相加
 97
 98
                   BigDecimal b = new BigDecimal(check);
                    // (boolean 判斷式)? (true的輸出):
 99
                       (false 的輸出)
                    System.out.print( flag ? check : " "
 100
                       + check);
 101
                    flag = false;
 102
                    // BigDecimal 內建 .add 可以直接相加
 103
                    ans = ans.add(b);
 104
               }
 105
           }
 106
 107
    11
 108
    // BigDecimal
 109
 110
    BigDecimal b = new BigDecimal(check);
 111
 112
                   // (boolean 判斷式)? (true的輸出):
                       (false 的輸出)
                    System.out.print( flag ? check : " "
 113
                       + check):
 114
                   flag = false;
 115
                   // BigDecimal 內建 .add 可以直接相加
 116
 117
                   ans = ans.add(b);
 118
    // -
 119
 120 // BigDecimal 四捨五入的寫法 .setScale(
        想要留取的位數, RoundingMode. HALF_UP)
```

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

```
265
                 a = Math.sqrt( s * (s-array[0]) *
                                                                 336
                     (s-array[1]) * (s-array[2]) );
                                                                 337
                                                                 338
266
                 // 利用 BigDecimal 去取小數點並四捨五入
                                                                 339
267
268
                 BigDecimal ans = new BigDecimal(a);
                                                                 340
269
                 System.out.println( ans.setScale(3,
                     RoundingMode.HALF_UP));
                                                                 341
270
            }
                                                                 342
271
272
273
                                                                 343
274
   }
                                                                 344
275
                                                                 345
276
277
                                                                 346
278
   - hypot
                                                                 347
279
                                                                 348
280
       primelist
                                                                 349
281
                                                                 350
282
   package com.company;
                                                                 351
283
284
   import java.util.Arrays;
                                                                 352
285
   import java.util.Scanner;
                                                                 353
286 import java.util.Vector;
                                                                 354
287
                                                                 355
   public class Main {
288
                                                                 356
289
                                                                 357
        public static void main(String[] args) {
290
291
            Scanner input = new Scanner(System.in);
                                                                 358
292
            int n = input.nextInt();
                                                                 359
293
                                                                 360
            // 建質數表
294
                                                                 361
            // java 中沒有全域變數
295
                                                                 362
                 所以這邊建表目前是每次輸入都重建一次
                 但不夠漂亮
                                                                 363
            boolean[] prime = new boolean[10000000+5];
296
                                                                 365
297
298
            // 用 vector 存質數們
            Vector<Integer> v = new Vector<Integer>();
                                                                 366
299
                                                                 367
                                                                         }
            for( int i = 2; i <= n; i++ ){</pre>
300
                                                                 368 }
301
                if(!prime[i]){
                                                                 369 // ---
302
                     v.add(i);
303
                     for( int j = i * i; j <= n; j += i ){</pre>
304
                         prime[j] = true;
305
                     }
                }
306
            }
307
308
            for( int i = 0, k = 0; i < v.size(); i++, k++</pre>
309
310
                 if( k == 9 || i == v.size() - 1){
311
312
                     System.out.println(v.get(i));
                     // 把 k 歸回去 -1 是因為下次輪迴會先
313
                         k++
314
                     k = -1;
                }
315
                 else{
316
                     System.out.print(v.get(i) + " ");
317
                }
318
319
            }
            v.clear();
320
321
322 }
323
324
       3-11
325
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) {
```

```
Scanner input = new Scanner(System.in);
int month = input.nextInt():
int year = input.nextInt();
// 分別存 30 天的月份 和 31 天的月份
    以及他們的英文月份名
int[] thirty = {1,3,5,7,8,10,12};
String[] thirty_eng = {"January", "March",
    "May", "July", "August", "October",
    "December"};
int[] thirty_one = {4,6,9,11};
String[] thirty_one_eng = { "April", "June",
    "September", "November"};
// 查詢他是否是 30 天的月份
// 最後放 return 後面的其他程式碼就不會執行了
for( int i = 0; i < thirty.length; i++){</pre>
    if(month == thirty[i]){
       {\tt System.out.println(thirty\_eng[i] + "}
            " + year + " had 31 Days" );
        return;
   }
for( int i = 0; i < thirty_one.length; i++){</pre>
   if(month == thirty_one[i]){
       System.out.println(thirty_one_eng[i]
           + " " + year + " had 31 Days" );
        return:
   }
if( year % 4 == 0){
    System.out.println("February " + year + "
        had 29 Days");
    return:
System.out.println("February " + year + " had
    28 Days");
```

3