

## Ejercicio Vector:

Llenar un vector de dimensión = 80 con valores (1,2,3,4) de forma aleatoria en la misma proporción, 20 veces cada número.

***Secuencial***

```
package Vector;// @Author: CABRERA PICOITA NAHOMI ASTRID
import java.util.Random;

// Llenar un vector de dimensión = 80 con valores (1,2,3,4) de forma
// aleatoria en la misma
// proporción, 20 veces cada numero.

public class Secuencial {
    private static int[] vector = new int[80];
    private static int[] comprueba = new int[4];
    private static Random r = new Random();

    public static void main(String[] args) {
        llenarVector();

        int uwu = 1;
        for (int xd : vector) { // este for each permite que se imprima en
            // filas de 20 el vector principal
            System.out.print(xd + " - ");
            if (uwu == 20 || uwu == 40 || uwu == 60) {
                System.out.println();
            }
            uwu++;
        }

        public static void llenarVector() {
            int n;
            int indx = 0;
            while (!verifica()) {
                n = r.nextInt(4) + 1;
                if (comprueba[n - 1] < 20) { // si es que el contador no ha
                    // llegado a 19 entonces se pueden seguir insertando
                    vector[indx] = n;
                    comprueba[n - 1]++;
                    indx++;
                }
            }
        }

        // verifica es la union de los contadores por cada número, la unión de
        // estos contadores es un array de 4
        // cuando todas las posiciones sean 20 devuelve un true indicando de que
        // el vector principal tiene 20 veces
        // cada uno de los números (1 2 3 4)
        public static boolean verifica() {
            for (int i = 0; i < comprueba.length; i++) {
                if (comprueba[i] < 20)
                    return false;
            }
            return true;
        }
    }
}
```

```
}  
}
```

## **Hilos**

```
package Vector; // @Author: CABRERA PICOITA NAHOMI ASTRID  
  
public class Hilos extends Thread {  
  
    private int n;  
    private static int vector[] = new int[80];  
    private static int cont = 0;  
    private int rango = 20;  
  
    public Hilos(int n) {  
        this.n = n;  
    }  
  
    // cada hilo se ejecutará 20 veces y agregarán un número en el vector que  
    // comparten los 4  
    // se irán agregando los números conforme vayan llegando  
    // lo único que recibe el hilo es el número  
    @Override  
    public void run() {  
        for (int i = 0; i < rango; i++) {  
            System.out.println("Agregado: " + n);  
            vector[cont] = n;  
            cont++;  
        }  
    }  
  
    public int getN() {  
        return n;  
    }  
  
    public static int[] getVector() {  
        return vector;  
    }  
  
    public static void main(String[] args) {  
        Hilos h1 = new Hilos(1);  
  
        Hilos h2 = new Hilos(2);  
  
        Hilos h3 = new Hilos(3);  
  
        Hilos h4 = new Hilos(4);  
  
        h1.start();  
        h2.start();  
        h3.start();  
        h4.start();  
  
        try { // se espera a que todos acaben de llenar el vector para  
            imprimir de igual forma que en la clase secuencial  
            h1.join();  
            h2.join();  
            h3.join();  
            h4.join();  
        } catch (InterruptedException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```

    int uwu = 1;
    for (int xd : getVector()) {
        System.out.print(xd + " - ");
        if (uwu == 20 || uwu == 40 || uwu == 60) {
            System.out.println();
        }
        uwu++;
    }
}

```

Ejercicio Matrix:

Llenar una matriz de dimensión (n x m (par)) con valores de 1 a n de forma aleatoria en la proporción de veces cada número (si la matriz es 3x4 se debe llenar con 1,2,3 4 veces cada número de forma aleatoria)

### ***Secuencial***

```

package Matriz; // @Author: CABRERA PICOITA NAHOMI ASTRID

import java.util.Random;
import java.util.Scanner;

// Llenar una matriz de dimensión (n x m (par)) con valores de 1 a n de forma
// aleatoria en la proporción de veces cada
// numero (si la matriz es 3x4 se debe llenar con 1,2,3 4 veces cada numero
// de forma aleatoria)
public class Secuencial {

    private static int[] comprueba;
    private static int[][] matriz;
    private static Scanner sc = new Scanner(System.in);
    private static Random r = new Random();

    public static void main(String[] args) {

        System.out.println("Ingrese las filas:");
        int f = sc.nextInt();
        System.out.println("Ingrese las columnas (número par):");
        int c = sc.nextInt();

        comprueba = new int[f];
        matriz = new int[f][c];

        if (c % 2 != 0) {
            System.out.println("Las columnas deben ser un número par");
            System.out.println("Adiós :D");
        } else {
            llenarMatriz();

            for (int i = 0; i < f; i++) {
                for (int j = 0; j < c; j++) {
                    System.out.print(matriz[i][j]);
                }
                System.out.println();
            }
        }
    }
}

```

```

// lo que hay que tener en cuenta en este problema es que el número de
// veces que se van a repetir los números del
// 1 al número de filas, serán el número de columnas, por eso el array de
// contadores debe
// llegar al número de columnas - 1
public static void llenarMatriz() {
    int n;
    int ixFila = 0;
    int ixColumn = 0;
    while (!verifica()) {
        n = r.nextInt(matriz.length) + 1; // es lo mismo que esto int n =
        1 + r.nextInt(f);

        if (comprueba[n - 1] < matriz[0].length) { // si es que el
            contador no ha llegado a num de columnas entonces se pueden seguir insertando
            if (ixColumn == matriz[0].length) {
                ixColumn = 0;
                ixFila++;
            }

            matriz[ixFila][ixColumn] = n;
            comprueba[n - 1]++;
            ixColumn++;
        }
    }
}

public static boolean verifica() {
    for (int i = 0; i < comprueba.length; i++) {
        if (comprueba[i] < matriz[0].length)
            return false;
    }
    return true;
}
}

```

## ***Hilos***

```

package Matriz; // @Author: CABRERA PICOITA NAHOMI ASTRID

import java.util.Scanner;
import java.util.concurrent.locks.ReentrantLock;

public class Hilos extends Thread {
    private static int[][] matriz;
    private static Scanner sc = new Scanner(System.in);
    private static ReentrantLock lock = new ReentrantLock();

    private static int ixFila = 0;
    private static int ixColumn = 0;

    private int row;
    private int f;
    private int c;
    private int n;

    public Hilos(int row, int f, int c, int n) {
        this.row = row;
        this.f = f;
        this.c = c;
        this.n = n;
        if (matriz == null) {

```

```

        matriz = new int[f][c];
    }
}

/*
@Override
public void run() {
    for (int i = 0; i < c; i++) {
        lock.lock(); // esto es para evitar la desincronización o
inconsistencia de datos
        try {
            System.out.println(getName() + " Agregado: " + n);
            matriz[row][i] = n;
        } finally {
            lock.unlock();
        }
    }
}
}
ESTE RUN DE AQUÍ EVITA QUE LAS CARRERAS ENTRE HILOS PASEN Y ASÍ EVITA
QUE LOS DATOS SE SOBREPONGAN UNO
SOBRE OTRO
*/
@Override
public void run() {
    for (int i = 0; i < c; i++) {
        System.out.println(getName() + " Agregado: " + n);
        matriz[ixFila][ixColumn] = n;

        ixFila++;
        ixColumn++;

        if (ixFila == f) {
            ixFila = 0;
        }
        if (ixColumn == c) {
            ixColumn = 0;
        }
    }
}

public static int[][] getMatriz() {
    return matriz;
}

public static void main(String[] args) {
    System.out.println("Ingrese las filas:");
    int f = sc.nextInt();
    System.out.println("Ingrese las columnas (número par):");
    int c = sc.nextInt();

    if (c % 2 != 0) {
        System.out.println("Las columnas deben ser un número par");
        System.out.println("Adiós :D");
    } else {
        Thread[] threads = new Thread[f];
        int numeros = 1;

        for (int i = 0; i < threads.length; i++) { // se le envía la row
por el run en el comentario
            threads[i] = new Hilos(i, f, c, numeros); // se inicializan
los hilos dentro del array y se corren
            threads[i].start();
            numeros++;
        }
    }
}

```

```

    }
    for (Thread xd : threads) { // se espera a que todos los hilos
acaben
        try {
            xd.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }

    for (int i = 0; i < f; i++) { // se imprime
        for (int j = 0; j < c; j++) {
            System.out.print(getMatriz()[i][j] + " ");
        }
        System.out.println();
    }
}
}
}

```

***Capturas:***

## Vector Secuencial

The screenshot shows the IntelliJ IDEA IDE with the following code in the `Secuencial` class:

```

package Vector; // @Author: CABRERA PICOITA NAHOMI ASTRID

import java.util.Random;

// llenar un vector de dimensi3n = 80 con valores (1,2,3,4) de forma aleatoria en la misma
// proporci3n, 20 veces cada numero.

public class Secuencial {
    2 usages
    private static int[] vector = new int[80];
    4 usages
    private static int[] comprob3a = new int[4];
    1 usage
}

```

The Run console shows the output of the `comprobar` method:

```

"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.5\lib\idea_rt.jar=50098:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.5\bin" -Dfile.encoding=UTF-8
4 - 3 - 4 - 3 - 4 - 1 - 4 - 1 - 4 - 3 - 2 - 1 - 1 - 2 - 1 - 1 - 3 - 3 - 1 - 3 -
4 - 1 - 4 - 3 - 2 - 1 - 3 - 2 - 3 - 3 - 2 - 3 - 1 - 3 - 2 - 1 - 2 - 1 - 2 - 3 -
4 - 4 - 1 - 2 - 3 - 4 - 1 - 1 - 3 - 2 - 3 - 3 - 3 - 1 - 4 - 2 - 4 - 2 - 1 - 3 -
2 - 1 - 1 - 4 - 4 - 4 - 3 - 4 - 1 - 2 - 2 - 2 - 4 - 4 - 4 - 2 - 2 - 2 - 4 - 2 -
Process finished with exit code 0

```

The screenshot displays an IDE with a project named 'pruebaParcialPA'. The code is in a file named 'VectorHilos.java' and is part of a package 'Vector'. The code defines a 'Secuencial' class with two static integer arrays: 'vector' of size 80 and 'comprueba' of size 4. The 'vector' array is initialized with the value 80. The 'comprueba' array is initialized with the value 4. The code also includes a comment indicating that the vector is filled with values (1,2,3,4) in a random order, and the proportion of each value is 20 times the number.

```

package Vector; // @Author: CABRERA PICOITA NAHOMI ASTRID

import java.util.Random;

// llenar un vector de dimensi3n = 80 con valores (1,2,3,4) de forma aleatoria en la misma
// proporci3n, 20 veces cada numero.

public class Secuencial {

    2 usages
    private static int[] vector = new int[80];
    4 usages
    private static int[] comprueba = new int[4];
    1 usage

```

The Run window shows the execution of the program. The output is a single line of 80 integers, representing the contents of the 'vector' array. The integers are 1s, 2s, 3s, and 4s, distributed in a random order. The process finished with exit code 0.

```

C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2\lib\idea_rt.jar=50105:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2\
1 - 3 - 1 - 1 - 3 - 4 - 3 - 1 - 2 - 2 - 3 - 4 - 4 - 4 - 3 - 4 - 3 - 3 - 1 - 1 -
2 - 1 - 3 - 3 - 3 - 2 - 4 - 4 - 2 - 1 - 4 - 2 - 1 - 2 - 1 - 4 - 2 - 4 - 1 - 3 -
1 - 4 - 3 - 1 - 4 - 1 - 2 - 1 - 4 - 4 - 2 - 4 - 4 - 2 - 1 - 4 - 4 - 1 - 4 - 3 -
4 - 3 - 2 - 1 - 3 - 1 - 2 - 1 - 3 - 3 - 3 - 3 - 3 - 2 - 2 - 2 - 2 - 2 - 2 - 2 -
Process finished with exit code 0

```

## Vector Hilos

```

package Vector; // @Author: CABRERA PICOITA NAHOMI ASTRID

import java.util.*;

public class Hilos extends Thread {

    4 usages
    private int n;
    2 usages
    private static int vector[] = new int[80];
    2 usages
    private static int cont = 0;
    1 usage
    private int rango = 20;

    4 usages
    // ... (rest of the code)

```



The screenshot shows an IDE with a project named 'pruebaParcialPA'. The file explorer on the left shows the project structure, including folders for 'Matriz' and 'Vector'. The main editor displays the code for 'Vector\Hilos.java', which defines a 'Hilos' class extending 'Thread'. The code includes package declarations, class annotations, and several static variables: 'n' (4 usages), 'vector' (2 usages), 'cont' (2 usages), and 'rango' (1 usage). The 'Run' tab at the bottom shows the execution output, which consists of a series of 'Agregado: 2' messages followed by a large sequence of numbers separated by hyphens, representing the results of the matrix calculation. The process finished with exit code 0.

```
package Vector; // @Author: CABRERA PICOITA NAHOMI ASTRID

public class Hilos extends Thread {

    4 usages
    private int n;
    2 usages
    private static int vector[] = new int[80];
    2 usages
    private static int cont = 0;
    1 usage
    private int rango = 20;
    4 usages

    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
    25
    26
    27
    28
    29
    30
    31
    32
    33
    34
    35
    36
    37
    38
    39
    40
    41
    42
    43
    44
    45
    46
    47
    48
    49
    50
    51
    52
    53
    54
    55
    56
    57
    58
    59
    60
    61
    62
    63
    64
    65
    66
    67
    68
    69
    70
    71
    72
    73
    74
    75
    76
    77
    78
    79
    80
    81
    82
    83
    84
    85
    86
    87
    88
    89
    90
    91
    92
    93
    94
    95
    96
    97
    98
    99
    100
    101
    102
    103
    104
    105
    106
    107
    108
    109
    110
    111
    112
    113
    114
    115
    116
    117
    118
    119
    120
    121
    122
    123
    124
    125
    126
    127
    128
    129
    130
    131
    132
    133
    134
    135
    136
    137
    138
    139
    140
    141
    142
    143
    144
    145
    146
    147
    148
    149
    150
    151
    152
    153
    154
    155
    156
    157
    158
    159
    160
    161
    162
    163
    164
    165
    166
    167
    168
    169
    170
    171
    172
    173
    174
    175
    176
    177
    178
    179
    180
    181
    182
    183
    184
    185
    186
    187
    188
    189
    190
    191
    192
    193
    194
    195
    196
    197
    198
    199
    200
    201
    202
    203
    204
    205
    206
    207
    208
    209
    210
    211
    212
    213
    214
    215
    216
    217
    218
    219
    220
    221
    222
    223
    224
    225
    226
    227
    228
    229
    230
    231
    232
    233
    234
    235
    236
    237
    238
    239
    240
    241
    242
    243
    244
    245
    246
    247
    248
    249
    250
    251
    252
    253
    254
    255
    256
    257
    258
    259
    260
    261
    262
    263
    264
    265
    266
    267
    268
    269
    270
    271
    272
    273
    274
    275
    276
    277
    278
    279
    280
    281
    282
    283
    284
    285
    286
    287
    288
    289
    290
    291
    292
    293
    294
    295
    296
    297
    298
    299
    300
    301
    302
    303
    304
    305
    306
    307
    308
    309
    310
    311
    312
    313
    314
    315
    316
    317
    318
    319
    320
    321
    322
    323
    324
    325
    326
    327
    328
    329
    330
    331
    332
    333
    334
    335
    336
    337
    338
    339
    340
    341
    342
    343
    344
    345
    346
    347
    348
    349
    350
    351
    352
    353
    354
    355
    356
    357
    358
    359
    360
    361
    362
    363
    364
    365
    366
    367
    368
    369
    370
    371
    372
    373
    374
    375
    376
    377
    378
    379
    380
    381
    382
    383
    384
    385
    386
    387
    388
    389
    390
    391
    392
    393
    394
    395
    396
    397
    398
    399
    400
    401
    402
    403
    404
    405
    406
    407
    408
    409
    410
    411
    412
    413
    414
    415
    416
    417
    418
    419
    420
    421
    422
    423
    424
    425
    426
    427
    428
    429
    430
    431
    432
    433
    434
    435
    436
    437
    438
    439
    440
    441
    442
    443
    444
    445
    446
    447
    448
    449
    450
    451
    452
    453
    454
    455
    456
    457
    458
    459
    460
    461
    462
    463
    464
    465
    466
    467
    468
    469
    470
    471
    472
    473
    474
    475
    476
    477
    478
    479
    480
    481
    482
    483
    484
    485
    486
    487
    488
    489
    490
    491
    492
    493
    494
    495
    496
    497
    498
    499
    500
    501
    502
    503
    504
    505
    506
    507
    508
    509
    510
    511
    512
    513
    514
    515
    516
    517
    518
    519
    520
    521
    522
    523
    524
    525
    526
    527
    528
    529
    530
    531
    532
    533
    534
    535
    536
    537
    538
    539
    540
    541
    542
    543
    544
    545
    546
    547
    548
    549
    550
    551
    552
    553
    554
    555
    556
    557
    558
    559
    560
    561
    562
    563
    564
    565
    566
    567
    568
    569
    570
    571
    572
    573
    574
    575
    576
    577
    578
    579
    580
    581
    582
    583
    584
    585
    586
    587
    588
    589
    590
    591
    592
    593
    594
    595
    596
    597
    598
    599
    600
    601
    602
    603
    604
    605
    606
    607
    608
    609
    610
    611
    612
    613
    614
    615
    616
    617
    618
    619
    620
    621
    622
    623
    624
    625
    626
    627
    628
    629
    630
    631
    632
    633
    634
    635
    636
    637
    638
    639
    640
    641
    642
    643
    644
    645
    646
    647
    648
    649
    650
    651
    652
    653
    654
    655
    656
    657
    658
    659
    660
    661
    662
    663
    664
    665
    666
    667
    668
    669
    670
    671
    672
    673
    674
    675
    676
    677
    678
    679
    680
    681
    682
    683
    684
    685
    686
    687
    688
    689
    690
    691
    692
    693
    694
    695
    696
    697
    698
    699
    700
    701
    702
    703
    704
    705
    706
    707
    708
    709
    710
    711
    712
    713
    714
    715
    716
    717
    718
    719
    720
    721
    722
    723
    724
    725
    726
    727
    728
    729
    730
    731
    732
    733
    734
    735
    736
    737
    738
    739
    740
    741
    742
    743
    744
    745
    746
    747
    748
    749
    750
    751
    752
    753
    754
    755
    756
    757
    758
    759
    760
    761
    762
    763
    764
    765
    766
    767
    768
    769
    770
    771
    772
    773
    774
    775
    776
    777
    778
    779
    780
    781
    782
    783
    784
    785
    786
    787
    788
    789
    790
    791
    792
    793
    794
    795
    796
    797
    798
    799
    800
    801
    802
    803
    804
    805
    806
    807
    808
    809
    810
    811
    812
    813
    814
    815
    816
    817
    818
    819
    820
    821
    822
    823
    824
    825
    826
    827
    828
    829
    830
    831
    832
    833
    834
    835
    836
    837
    838
    839
    840
    841
    842
    843
    844
    845
    846
    847
    848
    849
    850
    851
    852
    853
    854
    855
    856
    857
    858
    859
    860
    861
    862
    863
    864
    865
    866
    867
    868
    869
    870
    871
    872
    873
    874
    875
    876
    877
    878
    879
    880
    881
    882
    883
    884
    885
    886
    887
    888
    889
    890
    891
    892
    893
    894
    895
    896
    897
    898
    899
    900
    901
    902
    903
    904
    905
    906
    907
    908
    909
    910
    911
    912
    913
    914
    915
    916
    917
    918
    919
    920
    921
    922
    923
    924
    925
    926
    927
    928
    929
    930
    931
    932
    933
    934
    935
    936
    937
    938
    939
    940
    941
    942
    943
    944
    945
    946
    947
    948
    949
    950
    951
    952
    953
    954
    955
    956
    957
    958
    959
    960
    961
    962
    963
    964
    965
    966
    967
    968
    969
    970
    971
    972
    973
    974
    975
    976
    977
    978
    979
    980
    981
    982
    983
    984
    985
    986
    987
    988
    989
    990
    991
    992
    993
    994
    995
    996
    997
    998
    999
    1000
    1001
    1002
    1003
    1004
    1005
    1006
    1007
    1008
    1009
    1010
    1011
    1012
    1013
    1014
    1015
    1016
    1017
    1018
    1019
    1020
    1021
    1022
    1023
    1024
    1025
    1026
    1027
    1028
    1029
    1030
    1031
    1032
    1033
    1034
    1035
    1036
    1037
    1038
    1039
    1040
    1041
    1042
    1043
    1044
    1045
    1046
    1047
    1048
    1049
    1050
    1051
    1052
    1053
    1054
    1055
    1056
    1057
    1058
    1059
    1060
    1061
    1062
    1063
    1064
    1065
    1066
    1067
    1068
    1069
    1070
    1071
    1072
    1073
    1074
    1075
    1076
    1077
    1078
    1079
    1080
    1081
    1082
    1083
    1084
    1085
    1086
    1087
    1088
    1089
    1090
    1091
    1092
    1093
    1094
    1095
    1096
    1097
    1098
    1099
    1100
    1101
    1102
    1103
    1104
    1105
    1106
    1107
    1108
    1109
    1110
    1111
    1112
    1113
    1114
    1115
    1116
    1117
    1118
    1119
    1120
    1121
    1122
    1123
    1124
    1125
    1126
    1127
    1128
    1129
    1130
    1131
    1132
    1133
    1134
    1135
    1136
    1137
    1138
    1139
    1140
    1141
    1142
    1143
    1144
    1145
    1146
    1147
    1148
    1149
    1150
    1151
    1152
    1153
    1154
    1155
    1156
    1157
    1158
    1159
    1160
    1161
    1162
    1163
    1164
    1165
    1166
    1167
    1168
    1169
    1170
    1171
    1172
    1173
    1174
    1175
    1176
    1177
    1178
    1179
    1180
    1181
    1182
    1183
    1184
    1185
    1186
    1187
    1188
    1189
    1190
    1191
    1192
    1193
    1194
    1195
    1196
    1197
    1198
    1199
    1200
    1201
    1202
    1203
    1204
    1205
    1206
    1207
    1208
    1209
    1210
    1211
    1212
    1213
    1214
    1215
    1216
    1217
    1218
    1219
    1220
    1221
    1222
    1223
    1224
    1225
    1226
    1227
    1228
    1229
    1230
    1231
    1232
    1233
    1234
    1235
    1236
    1237
    1238
    1239
    1240
    1241
    1242
    1243
    1244
    1245
    1246
    1247
    1248
    1249
    1250
    1251
    1252
    1253
    1254
    1255
    1256
    1257
    1258
    1259
    1260
    1261
    1262
    1263
    1264
    1265
    1266
    1267
    1268
    1269
    1270
    1271
    1272
    1273
    1274
    1275
    1276
    1277
    1278
    1279
    1280
    1281
    1282
    1283
    1284
    1285
    1286
    1287
    1288
    1289
    1290
    1291
    1292
    1293
    1294
    1295
    1296
    1297
    1298
    1299
    1300
    1301
    1302
    1303
    1304
    1305
    1306
    1307
    1308
    1309
    1310
    1311
    1312
    1313
    1314
    1315
    1316
    1317
    1318
    1319
    1320
    1321
    1322
    1323
    1324
    1325
    1326
    1327
    1328
    1329
    1330
    1331
    1332
    1333
    1334
    1335
    1336
    1337
    1338
    1339
    1340
    1341
    1342
    1343
    1344
    1345
    1346
    1347
    1348
    1349
    1350
    1351
    1352
    1353
    1354
    1355
    1356
    1357
    1358
    1359
    1360
    1361
    1362
    1363
    1364
    1365
    1366
    1367
    1368
    1369
    1370
    1371
    1372
    1373
    1374
    1375
    1376
    1377
    1378
    1379
    1380
    1381
    1382
    1383
    1384
    1385
    1386
    1387
    1388
    1389
    1390
    1391
    1392
    1393
    1394
    1395
    1396
    1397
    1398
    1399
    1400
    1401
    1402
    1403
    1404
    1405
    1406
    1407
    1408
    1409
    1410
    1411
    1412
    1413
    1414
    1415
    1416
    1417
    1418
    1419
    1420
    1421
    1422
    1423
    1424
    1425
    1426
    1427
    1428
    1429
    1430
    1431
    1432
    1433
    1434
    1435
    1436
    1437
    1438
    1439
    1440
    1441
    1442
    1443
    1444
    1445
    1446
    1447
    1448
    1449
    1450
    1451
    1452
    1453
    1454
    1455
    1456
    1457
    1458
    1459
    1460
    1461
    1462
    1463
    1464
    1465
    1466
    1467
    1468
    1469
    1470
    1471
    1472
    1473
    1474
    1475
    1476
    1477
    1478
    1479
    1480
    1481
    1482
    1483
    1484
    1485
    1486
    1487
    1488
    1489
    1490
    1491
    1492
    1493
    1494
    1495
    1496
    1497
    1498
    1499
    1500
    1501
    1502
    1503
    1504
    1505
    1506
    1507
    1508
    1509
    1510
    1511
    1512
    1513
    1514
    1515
    1516
    1517
    1518
    1519
    1520
    1521
    1522
    1523
    1524
    1525
    1526
    1527
    1528
    1529
    1530
    1531
    1532
    1533
    1534
    1535
    1536
    1537
    1538
    1539
    1540
    1541
    1542
    1543
    1544
    1545
    1546
    1547
    1548
    1549
    1550
    1551
    1552
    1553
    1554
    1555
    1556
    1557
    1558
    1559
    1560
    1561
    1562
    1563
    1564
    1565
    1566
    1567
    1568
    1569
    1570
    1571
    1572
    1573
    1574
    1575
    1576
    1577
    1578
    1579
    1580
    1581
    1582
    1583
    1584
    1585
    1586
    1587
    1588
    1589
    1590
    1591
    1592
    1593
    1594
    1595
    1596
    1597
    1598
    1599
    1600
    1601
    1602
    1603
    1604
    1605
    1606
    1607
    1608
    1609
    1610
    1611
    1612
    1613
    1614
    1615
    1616
    1617
    1618
    1619
    1620
    1621
    1622
    1623
    1624
    1625
    1626
    1627
    1628
    1629
    1630
    1631
    1632
    1633
    1634
    1635
    1636
    1637
    1638
    1639
    1640
    1641
    1642
    1643
    1644
    1645
    1646
    1647
    1648
    1649
    1650
    1651
    1652
    1653
    1654
    1655
    1656
    1657
    1658
    1659
    1660
    1661
    1662
    1663
    1664
    1665
    1666
    1667
    1668
    1669
    1670
    1671
    1672
    1673
    1674
    1675
    1676
    1677
    1678
    1679
    1680
    1681
    1682
    1683
    1684
    1685
    1686
    1687
    1688
    1689
    1690
    1691
    1692
    1693
    1694
    1695
    1696
    1697
    1698
    1699
    1700
    1701
    1702
    1703
    1704
    1705
    1706
    1707
    1708
    1709
    1710
    1711
    1712
    1713
    1714
    1715
    1716
    1717
    1718
    1719
    1720
    1721
    1722
    1723
    1724
    1725
    1726
    1727
    1728
    1729
    1730
    1731
    1732
    1733
    1734
    1735
    1736
    1737
    1738
    1739
    1740
    1741
    1742
    1743
    1744
    1745
    1746
    1747
    1748
    1749
    1750
    1751
    1752
    1753
    1754
    1755
    1756
    1757
    1758
    1759
    1760
    1761
    1762
    1763
    1764
    1765
    1766
    1767
    1768
    1769
    1770
    1771
    1772
    1773
    1774
    1775
    1776
    1777
    1778
    1779
    1780
    1781
    1782
    1783
    1784
    1785
    1786
    1787
    1788
    1789
    1790
    1791
    1792
    1793
    1794
    1795
    1796
    1797
    1798
    1799
    1800
    1801
    1802
    1803
    1804
    1805
    1806
    1807
    1808
    1809
    1810
    1811
    1812
    1813
    1814
    1815
    1816
    1817
    1818
    1819
    1820
    1821
    1822
    1823
    1824
    1825
    1826
    1827
    1828
    1829
    1830
    1831
    1832
    1833
    1834
    1835
    1836
    1837
    1838
    1839
    1840
    1841
    1842
    1843
    1844
    1845
    1846
    1847
    1848
    1849
    1850
    1851
    1852
    1853
    1854
    1855
    1856
    1857
    1858
    1859
    1860
    1861
    1862
    1863
    1864
    1865
    1866
    1867
    1868
    1869
    1870
    
```

The screenshot displays the IntelliJ IDEA interface. The top toolbar includes icons for running and debugging. The 'Project' view on the left shows the project structure: 'pruebaParcialPA' (C:\Users\DEL\Documents) containing '.idea', 'out', 'src', and 'Vector'. The 'src' folder contains 'Matriz' (with 'Hilos' and 'Secuencial') and 'Vector' (with 'Hilos' and 'Secuencial'). The 'Run' tab is active, showing the execution of 'Secuencial'. The console output shows the program's execution, including prompts for rows and columns, and the resulting matrix values.

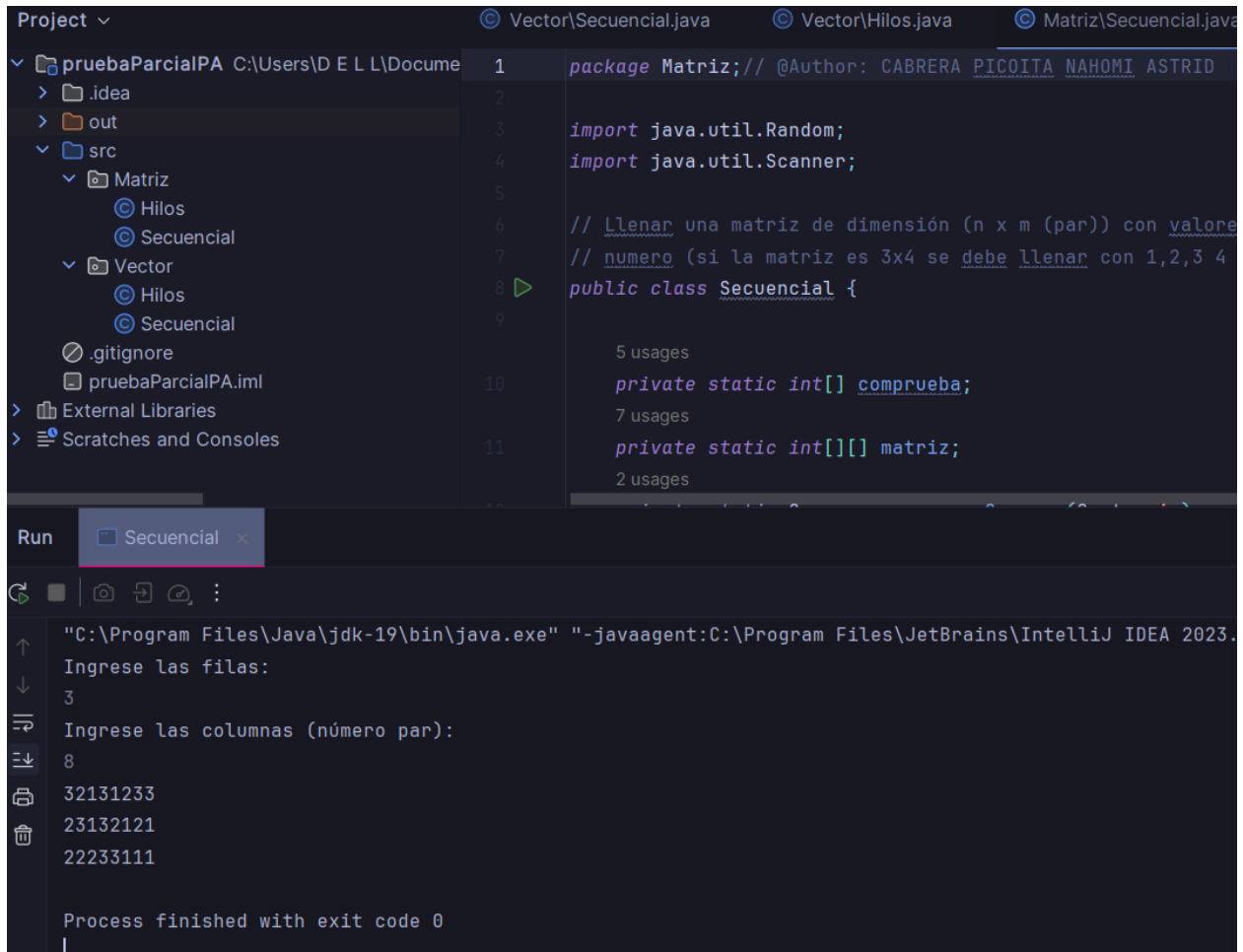
```
Project ▾
  ▾ pruebaParcialPA C:\Users\DEL\Documents
    > .idea
    > out
    ▾ src
      ▾ Matriz
        © Hilos
        © Secuencial
      ▾ Vector
        © Hilos
        © Secuencial
    .gitignore
    pruebaParcialPA.iml
  > External Libraries
  > Scratches and Consoles
```

```
Run ▾ Secuencial x
```

```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023
Ingrese las filas:
3
Ingrese las columnas (número par):
4
3333
2212
2111

Process finished with exit code 0
```

```
1 package Matriz; // @Author: CABRERA PICOITA NAHOMI ASTRID
2
3 import java.util.Random;
4 import java.util.Scanner;
5
6 // Llenar una matriz de dimensión (n x m (par)) con valor
7 // numero (si la matriz es 3x4 se debe llenar con 1,2,3 4
8 public class Secuencial {
9
10     5 usages
11     private static int[] comprueba;
12     7 usages
13     private static int[][] matriz;
14     2 usages
```



The screenshot displays the IntelliJ IDEA IDE interface. The top toolbar shows the 'Run' button (a green play icon) and a 'Run' tab labeled 'Secuencial'. The main editor window shows the source code for 'Matriz\Secuencial.java'. The code is as follows:

```
1 package Matriz; // @Author: CABRERA PICOITA NAHOMI ASTRID
2
3 import java.util.Random;
4 import java.util.Scanner;
5
6 // Llenar una matriz de dimensión (n x m (par)) con valores
7 // numero (si la matriz es 3x4 se debe llenar con 1,2,3 4
8 public class Secuencial {
9
10     5 usages
11     private static int[] comprueba;
12     7 usages
13     private static int[][] matriz;
14     2 usages
```

The bottom panel shows the 'Run' output for the 'Secuencial' configuration. The output is as follows:

```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.
Ingrese las filas:
3
Ingrese las columnas (número par):
8
32131233
23132121
22233111

Process finished with exit code 0
```

## ***Matriz Hilos***

The screenshot shows an IDE with a project named 'pruebaParcialPA'. The left sidebar displays the project structure, including folders like '.idea', 'out', 'src', and 'Matriz'. The main editor window shows the code for 'Hilos.java' in the 'Matriz' package. The code defines a 'Hilos' class that extends 'Thread' and implements a multithreaded process. The 'Run' button is highlighted, and the output console shows the execution results.

```
Project ▾
  ▾ pruebaParcialPA C:\Users\DELL\Documents
    > .idea
    > out
    ▾ src
      ▾ Matriz
        © Hilos
        © Secuencial
      ▾ Vector
        © Hilos
        © Secuencial
      .gitignore
      pruebaParcialPA.iml
    > External Libraries
    > Scratches and Consoles

Vector\Secuencial.java Vector\Hilos.java Matriz\Secuencial.java
1 package Matriz; // @Author: CABRERA PICOITA NAHOMI ASTRID
2
3 import java.util.Scanner;
4 import java.util.concurrent.locks.ReentrantLock;
5
6 public class Hilos extends Thread {
7     private static int[][] matriz;
8     private static Scanner sc = new Scanner(System.in);
9     private static ReentrantLock lock = new ReentrantLock();
10
11     4 usages
12     2 usages
13     no usages
14     4 usages
15
16     4 usages
17     2 usages
18     no usages
19     4 usages
20
21     4 usages
22     2 usages
23     no usages
24     4 usages
25
26     4 usages
27     2 usages
28     no usages
29     4 usages
30
31     4 usages
32     2 usages
33     no usages
34     4 usages
35
36     4 usages
37     2 usages
38     no usages
39     4 usages
40
41     4 usages
42     2 usages
43     no usages
44     4 usages
45
46     4 usages
47     2 usages
48     no usages
49     4 usages
50
51     4 usages
52     2 usages
53     no usages
54     4 usages
55
56     4 usages
57     2 usages
58     no usages
59     4 usages
60
61     4 usages
62     2 usages
63     no usages
64     4 usages
65
66     4 usages
67     2 usages
68     no usages
69     4 usages
70
71     4 usages
72     2 usages
73     no usages
74     4 usages
75
76     4 usages
77     2 usages
78     no usages
79     4 usages
80
81     4 usages
82     2 usages
83     no usages
84     4 usages
85
86     4 usages
87     2 usages
88     no usages
89     4 usages
90
91     4 usages
92     2 usages
93     no usages
94     4 usages
95
96     4 usages
97     2 usages
98     no usages
99     4 usages
100
101     4 usages
102     2 usages
103     no usages
104     4 usages
105
106     4 usages
107     2 usages
108     no usages
109     4 usages
110
111     4 usages
112     2 usages
113     no usages
114     4 usages
115
116     4 usages
117     2 usages
118     no usages
119     4 usages
120
121     4 usages
122     2 usages
123     no usages
124     4 usages
125
126     4 usages
127     2 usages
128     no usages
129     4 usages
130
131     4 usages
132     2 usages
133     no usages
134     4 usages
135
136     4 usages
137     2 usages
138     no usages
139     4 usages
140
141     4 usages
142     2 usages
143     no usages
144     4 usages
145
146     4 usages
147     2 usages
148     no usages
149     4 usages
150
151     4 usages
152     2 usages
153     no usages
154     4 usages
155
156     4 usages
157     2 usages
158     no usages
159     4 usages
160
161     4 usages
162     2 usages
163     no usages
164     4 usages
165
166     4 usages
167     2 usages
168     no usages
169     4 usages
170
171     4 usages
172     2 usages
173     no usages
174     4 usages
175
176     4 usages
177     2 usages
178     no usages
179     4 usages
180
181     4 usages
182     2 usages
183     no usages
184     4 usages
185
186     4 usages
187     2 usages
188     no usages
189     4 usages
190
191     4 usages
192     2 usages
193     no usages
194     4 usages
195
196     4 usages
197     2 usages
198     no usages
199     4 usages
200
201     4 usages
202     2 usages
203     no usages
204     4 usages
205
206     4 usages
207     2 usages
208     no usages
209     4 usages
210
211     4 usages
212     2 usages
213     no usages
214     4 usages
215
216     4 usages
217     2 usages
218     no usages
219     4 usages
220
221     4 usages
222     2 usages
223     no usages
224     4 usages
225
226     4 usages
227     2 usages
228     no usages
229     4 usages
230
231     4 usages
232     2 usages
233     no usages
234     4 usages
235
236     4 usages
237     2 usages
238     no usages
239     4 usages
240
241     4 usages
242     2 usages
243     no usages
244     4 usages
245
246     4 usages
247     2 usages
248     no usages
249     4 usages
250
251     4 usages
252     2 usages
253     no usages
254     4 usages
255
256     4 usages
257     2 usages
258     no usages
259     4 usages
260
261     4 usages
262     2 usages
263     no usages
264     4 usages
265
266     4 usages
267     2 usages
268     no usages
269     4 usages
270
271     4 usages
272     2 usages
273     no usages
274     4 usages
275
276     4 usages
277     2 usages
278     no usages
279     4 usages
280
281     4 usages
282     2 usages
283     no usages
284     4 usages
285
286     4 usages
287     2 usages
288     no usages
289     4 usages
290
291     4 usages
292     2 usages
293     no usages
294     4 usages
295
296     4 usages
297     2 usages
298     no usages
299     4 usages
300
301     4 usages
302     2 usages
303     no usages
304     4 usages
305
306     4 usages
307     2 usages
308     no usages
309     4 usages
310
311     4 usages
312     2 usages
313     no usages
314     4 usages
315
316     4 usages
317     2 usages
318     no usages
319     4 usages
320
321     4 usages
322     2 usages
323     no usages
324     4 usages
325
326     4 usages
327     2 usages
328     no usages
329     4 usages
330
331     4 usages
332     2 usages
333     no usages
334     4 usages
335
336     4 usages
337     2 usages
338     no usages
339     4 usages
340
341     4 usages
342     2 usages
343     no usages
344     4 usages
345
346     4 usages
347     2 usages
348     no usages
349     4 usages
350
351     4 usages
352     2 usages
353     no usages
354     4 usages
355
356     4 usages
357     2 usages
358     no usages
359     4 usages
360
361     4 usages
362     2 usages
363     no usages
364     4 usages
365
366     4 usages
367     2 usages
368     no usages
369     4 usages
370
371     4 usages
372     2 usages
373     no usages
374     4 usages
375
376     4 usages
377     2 usages
378     no usages
379     4 usages
380
381     4 usages
382     2 usages
383     no usages
384     4 usages
385
386     4 usages
387     2 usages
388     no usages
389     4 usages
390
391     4 usages
392     2 usages
393     no usages
394     4 usages
395
396     4 usages
397     2 usages
398     no usages
399     4 usages
400
401     4 usages
402     2 usages
403     no usages
404     4 usages
405
406     4 usages
407     2 usages
408     no usages
409     4 usages
410
411     4 usages
412     2 usages
413     no usages
414     4 usages
415
416     4 usages
417     2 usages
418     no usages
419     4 usages
420
421     4 usages
422     2 usages
423     no usages
424     4 usages
425
426     4 usages
427     2 usages
428     no usages
429     4 usages
430
431     4 usages
432     2 usages
433     no usages
434     4 usages
435
436     4 usages
437     2 usages
438     no usages
439     4 usages
440
441     4 usages
442     2 usages
443     no usages
444     4 usages
445
446     4 usages
447     2 usages
448     no usages
449     4 usages
450
451     4 usages
452     2 usages
453     no usages
454     4 usages
455
456     4 usages
457     2 usages
458     no usages
459     4 usages
460
461     4 usages
462     2 usages
463     no usages
464     4 usages
465
466     4 usages
467     2 usages
468     no usages
469     4 usages
470
471     4 usages
472     2 usages
473     no usages
474     4 usages
475
476     4 usages
477     2 usages
478     no usages
479     4 usages
480
481     4 usages
482     2 usages
483     no usages
484     4 usages
485
486     4 usages
487     2 usages
488     no usages
489     4 usages
490
491     4 usages
492     2 usages
493     no usages
494     4 usages
495
496     4 usages
497     2 usages
498     no usages
499     4 usages
500
501     4 usages
502     2 usages
503     no usages
504     4 usages
505
506     4 usages
507     2 usages
508     no usages
509     4 usages
510
511     4 usages
512     2 usages
513     no usages
514     4 usages
515
516     4 usages
517     2 usages
518     no usages
519     4 usages
520
521     4 usages
522     2 usages
523     no usages
524     4 usages
525
526     4 usages
527     2 usages
528     no usages
529     4 usages
530
531     4 usages
532     2 usages
533     no usages
534     4 usages
535
536     4 usages
537     2 usages
538     no usages
539     4 usages
540
541     4 usages
542     2 usages
543     no usages
544     4 usages
545
546     4 usages
547     2 usages
548     no usages
549     4 usages
550
551     4 usages
552     2 usages
553     no usages
554     4 usages
555
556     4 usages
557     2 usages
558     no usages
559     4 usages
560
561     4 usages
562     2 usages
563     no usages
564     4 usages
565
566     4 usages
567     2 usages
568     no usages
569     4 usages
570
571     4 usages
572     2 usages
573     no usages
574     4 usages
575
576     4 usages
577     2 usages
578     no usages
579     4 usages
580
581     4 usages
582     2 usages
583     no usages
584     4 usages
585
586     4 usages
587     2 usages
588     no usages
589     4 usages
590
591     4 usages
592     2 usages
593     no usages
594     4 usages
595
596     4 usages
597     2 usages
598     no usages
599     4 usages
600
601     4 usages
602     2 usages
603     no usages
604     4 usages
605
606     4 usages
607     2 usages
608     no usages
609     4 usages
610
611     4 usages
612     2 usages
613     no usages
614     4 usages
615
616     4 usages
617     2 usages
618     no usages
619     4 usages
620
621     4 usages
622     2 usages
623     no usages
624     4 usages
625
626     4 usages
627     2 usages
628     no usages
629     4 usages
630
631     4 usages
632     2 usages
633     no usages
634     4 usages
635
636     4 usages
637     2 usages
638     no usages
639     4 usages
640
641     4 usages
642     2 usages
643     no usages
644     4 usages
645
646     4 usages
647     2 usages
648     no usages
649     4 usages
650
651     4 usages
652     2 usages
653     no usages
654     4 usages
655
656     4 usages
657     2 usages
658     no usages
659     4 usages
660
661     4 usages
662     2 usages
663     no usages
664     4 usages
665
666     4 usages
667     2 usages
668     no usages
669     4 usages
670
671     4 usages
672     2 usages
673     no usages
674     4 usages
675
676     4 usages
677     2 usages
678     no usages
679     4 usages
680
681     4 usages
682     2 usages
683     no usages
684     4 usages
685
686     4 usages
687     2 usages
688     no usages
689     4 usages
690
691     4 usages
692     2 usages
693     no usages
694     4 usages
695
696     4 usages
697     2 usages
698     no usages
699     4 usages
700
701     4 usages
702     2 usages
703     no usages
704     4 usages
705
706     4 usages
707     2 usages
708     no usages
709     4 usages
710
711     4 usages
712     2 usages
713     no usages
714     4 usages
715
716     4 usages
717     2 usages
718     no usages
719     4 usages
720
721     4 usages
722     2 usages
723     no usages
724     4 usages
725
726     4 usages
727     2 usages
728     no usages
729     4 usages
730
731     4 usages
732     2 usages
733     no usages
734     4 usages
735
736     4 usages
737     2 usages
738     no usages
739     4 usages
740
741     4 usages
742     2 usages
743     no usages
744     4 usages
745
746     4 usages
747     2 usages
748     no usages
749     4 usages
750
751     4 usages
752     2 usages
753     no usages
754     4 usages
755
756     4 usages
757     2 usages
758     no usages
759     4 usages
760
761     4 usages
762     2 usages
763     no usages
764     4 usages
765
766     4 usages
767     2 usages
768     no usages
769     4 usages
770
771     4 usages
772     2 usages
773     no usages
774     4 usages
775
776     4 usages
777     2 usages
778     no usages
779     4 usages
780
781     4 usages
782     2 usages
783     no usages
784     4 usages
785
786     4 usages
787     2 usages
788     no usages
789     4 usages
790
791     4 usages
792     2 usages
793     no usages
794     4 usages
795
796     4 usages
797     2 usages
798     no usages
799     4 usages
800
801     4 usages
802     2 usages
803     no usages
804     4 usages
805
806     4 usages
807     2 usages
808     no usages
809     4 usages
810
811     4 usages
812     2 usages
813     no usages
814     4 usages
815
816     4 usages
817     2 usages
818     no usages
819     4 usages
820
821     4 usages
822     2 usages
823     no usages
824     4 usages
825
826     4 usages
827     2 usages
828     no usages
829     4 usages
830
831     4 usages
832     2 usages
833     no usages
834     4 usages
835
836     4 usages
837     2 usages
838     no usages
839     4 usages
840
841     4 usages
842     2 usages
843     no usages
844     4 usages
845
846     4 usages
847     2 usages
848     no usages
849     4 usages
850
851     4 usages
852     2 usages
853     no usages
854     4 usages
855
856     4 usages
857     2 usages
858     no usages
859     4 usages
860
861     4 usages
862     2 usages
863     no usages
864     4 usages
865
866     4 usages
867     2 usages
868     no usages
869     4 usages
870
871     4 usages
872     2 usages
873     no usages
874     4 usages
875
876     4 usages
877     2 usages
878     no usages
879     4 usages
880
881     4 usages
882     2 usages
883     no usages
884     4 usages
885
886     4 usages
887     2 usages
888     no usages
889     4 usages
890
891     4 usages
892     2 usages
893     no usages
894     4 usages
895
896     4 usages
897     2 usages
898     no usages
899     4 usages
900
901     4 usages
902     2 usages
903     no usages
904     4 usages
905
906     4 usages
907     2 usages
908     no usages
909     4 usages
910
911     4 usages
912     2 usages
913     no usages
914     4 usages
915
916     4 usages
917     2 usages
918     no usages
919     4 usages
920
921     4 usages
922     2 usages
923     no usages
924     4 usages
925
926     4 usages
927     2 usages
928     no usages
929     4 usages
930
931     4 usages
932     2 usages
933     no usages
934     4 usages
935
936     4 usages
937     2 usages
938     no usages
939     4 usages
940
941     4 usages
942     2 usages
943     no usages
944     4 usages
945
946     4 usages
947     2 usages
948     no usages
949     4 usages
950
951     4 usages
952     2 usages
953     no usages
954     4 usages
955
956     4 usages
957     2 usages
958     no usages
959     4 usages
960
961     4 usages
962     2 usages
963     no usages
964     4 usages
965
966     4 usages
967     2 usages
968     no usages
969     4 usages
970
971     4 usages
972     2 usages
973     no usages
974     4 usages
975
976     4 usages
977     2 usages
978     no usages
979     4 usages
980
981     4 usages
982     2 usages
983     no usages
984     4 usages
985
986     4 usages
987     2 usages
988     no usages
989     4 usages
990
991     4 usages
992     2 usages
993     no usages
994     4 usages
995
996     4 usages
997     2 usages
998     no usages
999     4 usages
1000
1001     4 usages
1002     2 usages
1003     no usages
1004     4 usages
1005
1006     4 usages
1007     2 usages
1008     no usages
1009     4 usages
1010
1011     4 usages
1012     2 usages
1013     no usages
1014     4 usages
1015
1016     4 usages
1017     2 usages
1018     no usages
1019     4 usages
1020
1021     4 usages
1022     2 usages
1023     no usages
1024     4 usages
1025
1026     4 usages
1027     2 usages
1028     no usages
1029     4 usages
1030
1031     4 usages
1032     2 usages
1033     no usages
1034     4 usages
1035
1036     4 usages
1037     2 usages
1038     no usages
1039     4 usages
1040
1041     4 usages
1042     2 usages
1043     no usages
1044     4 usages
1045
1046     4 usages
1047     2 usages
1048     no usages
1049     4 usages
1050
1051     4 usages
1052     2 usages
1053     no usages
1054     4 usages
1055
1056     4 usages
1057     2 usages
1058     no usages
1059     4 usages
1060
1061     4 usages
1062     2 usages
1063     no usages
1064     4 usages
1065
1066     4 usages
1067     2 usages
1068     no usages
1069     4 usages
1070
1071     4 usages
1072     2 usages
1073     no usages
1074     4 usages
1075
1076     4 usages
1077     2 usages
1078     no usages
1079     4 usages
1080
1081     4 usages
1082     2 usages
1083     no usages
1084     4 usages
1085
1086     4 usages
1087     2 usages
1088     no usages
1089     4 usages
1090
1091     4 usages
1092     2 usages
1093     no usages
1094     4 usages
1095
1096     4 usages
1097     2 usages
1098     no usages
1099     4 usages
1100
1101     4 usages
1102     2 usages
1103     no usages
1104     4 usages
1105
1106     4 usages
1107     2 usages
1108     no usages
1109     4 usages
1110
1111     4 usages
1112     2 usages
1113     no usages
1114     4 usages
1115
1116     4 usages
1117     2 usages
1118     no usages
1119     4 usages
1120
1121     4 usages
1122     2 usages
1123     no usages
1124     4 usages
1125
1126     4 usages
1127     2 usages
1128     no usages
1129     4 usages
1130
1131     4 usages
1132     2 usages
1133     no usages
1134     4 usages
1135
1136     4 usages
1137     2 usages
1138     no usages
1139     4 usages
1140
1141     4 usages
1142     2 usages
1143     no usages
1144     4 usages
1145
1146     4 usages
1147     2 usages
1148     no usages
1149     4 usages
1150
1151     4 usages
1152     2 usages
1153     no usages
1154     4 usages
1155
1156     4 usages
1157     2 usages
1158     no usages
1159     4 usages
1160
1161     4 usages
1162     2 usages
1163     no usages
1164     4 usages
1165
1166     4 usages
1167     2 usages
1168     no usages
1169     4 usages
1170
1171     4 usages
1172     2 usages
1173     no usages
1174     4 usages
1175
1176     4 usages
1177     2 usages
1178     no usages
1179     4 usages
1180
1181     4 usages
1182     2 usages
1183     no usages
1184     4 usages
1185
1186     4 usages
1187     2 usages
1188     no usages
1189     4 usages
1190
1191     4 usages
1192     2 usages
1193     no usages
1194     4 usages
1195
1196     4 usages
1197     2 usages
1198     no usages
1199     4 usages
1200
1201     4 usages
1202     2 usages
1203     no usages
1204     4 usages
1205
1206     4 usages
1207     2 usages
1208     no usages
1209     4 usages
1210
1211     4 usages
1212     2 usages
1213     no usages
1214     4 usages
1215
1216     4 usages
1217     2 usages
1218     no usages
1219     4 usages
1220
1221     4 usages
1222     2 usages
1223     no usages
1224     4 usages
1225
1226     4 usages
1227     2 usages
1228     no usages
1229     4 usages
1230
1231     4 usages
1232     2 usages
1233     no usages
1234     4 usages
1235
1236     4 usages
1237     2 usages
1238     no usages
1239     4 usages
1240
1241     4 usages
1242     2 usages
1243     no usages
1244     4 usages
1245
1246     4 usages
1247     2 usages
1248     no usages
1249     4 usages
1250
1251     4 usages
1252     2 usages
1253     no usages
1254     4 usages
1255
1256     4 usages
1257     2 usages
1258     no usages
1259     4 usages
1260
1261     4 usages
1262     2 usages
1263     no usages
1264     4 usages
1265
1266     4 usages
1267     2 usages
1268     no usages
1269     4 usages
1270
1271     4 usages
1272     2 usages
1273     no usages
1274     4 usages
1275
1276     4 usages
1277     2 usages
1278     no usages
1279     4 usages
1280
1281     4 usages
1282     2 usages
1283     no usages
1284     4 usages
1285
1286     4 usages
1287     2 usages
1288     no usages
1289     4 usages
1290
1291     4 usages
1292     2 usages
1293     no usages
1294     4 usages
1295
1296     4 usages
1297     2 usages
1298     no usages
1299     4 usages
1300
1301     4 usages
1302     2 usages
1303     no usages
1304     4 usages
1305
1306     4 usages
1307     2 usages
1308     no usages
1309     4 usages
1310
1311     4 usages
1312     2 usages
1313     no usages
1314     4 usages
1315
1316     4 usages
1317     2 usages
1318     no usages
1319     4 usages
1320
1321     4 usages
1322     2 usages
1323     no usages
1324     4 usages
1325
1326     4 usages
1327     2 usages
1328     no usages
1329     4 usages
1330
1331     4 usages
1332     2 usages
1333     no usages
1334     4 usages
1335
1336     4 usages
1337     2 usages
1338     no usages
1339     4 usages
1340
1341     4 usages
1342     2 usages
1343     no usages
1344     4 usages
1345
1346     4 usages
1347     2 usages
1348     no usages
1349     4 usages
1350
1351     4 usages
1352     2 usages
1353     no usages
1354     4 usages
1355
1356     4 usages
1357     2 usages
1358     no usages
1359     4 usages
1360
1361     4 usages
1362     2 usages
1363     no usages
1364     4 usages
1365
1366     4 usages
1367     2 usages
1368     no usages
1369     4 usages
1370
1371     4 usages
1372     2 usages
1373     no usages
1374     4 usages
1375
1376     4 usages
1377     2 usages
1378     no usages
1379     4 usages
1380
1381     4 usages
1382     2 usages
1383     no usages
1384     4 usages
1385
1386     4 usages
1387     2 usages
1388     no usages
1389     4 usages
1390
1391     4 usages
1392     2 usages
1393     no usages
1394     4 usages
1395
1396     4 usages
1397     2 usages
1398     no usages
1399     4 usages
1400
1401     4 usages
1402     2 usages
1403     no usages
1404     4 usages
1405
1406     4 usages
1407     2 usages
1408     no usages
1409     4 usages
1410
1411     4 usages
1412     2 usages
1413     no usages
1414     4 usages
1415
1416     4 usages
1417     2 usages
1418     no usages
1419     4 usages
1420
1421     4 usages
1422     2 usages
1423     no usages
1424     4 usages
1425
1426     4 usages
1427     2 usages
1428     no usages
1429     4 usages
1430
1431     4 usages
1432     2 usages
1433     no usages
1434     4 usages
1435
1436     4 usages
1437     2 usages
1438     no usages
1439     4 usages
1440
1441     4 usages
1442     2 usages
1443     no usages
1444     4 usages
1445
1446     4 usages
1447     2 usages
1448     no usages
1449     4 usages
1450
1451     4 usages
1452     2 usages
1453     no usages
1454     4 usages
1455
1456     4 usages
1457     2 usages
1458     no usages
1459     4 usages
1460
1461     4 usages
1462     2 usages
1463     no usages
1464     4 usages
1465
1466     4 usages
1467     2 usages
1468     no usages
1469     4 usages
1470
1471     4 usages
1472     2 usages
1473     no usages
1474     4 usages
1475
1476     4 usages
1477     2 usages
1478     no usages
1479     4 usages
1480
1481     4 usages
1482     2 usages
1483     no usages
1484     4 usages
1485
1486     4 usages
1487     2 usages
1488     no usages
148
```

The screenshot displays an IDE with a project named 'pruebaParcialPA'. The project structure includes a 'src' directory with sub-packages 'Matriz' and 'Vector'. The 'Matriz' package contains 'Hilos' and 'Secuencial' classes. The 'Vector' package also contains 'Hilos' and 'Secuencial' classes. The 'Hilos' class in the 'Matriz' package is the active file, showing the following code:

```
1 package Matriz; // @Author: CABRERA PICOITA NAHOMI ASTRID
2
3 import java.util.Scanner;
4 import java.util.concurrent.locks.ReentrantLock;
5
6 public class Hilos extends Thread {
7     private static int[][] matriz;
8     private static Scanner sc = new Scanner(System.in);
9     private static ReentrantLock lock = new ReentrantLock();
10
11     // ... (code is partially obscured) ...
```

The 'Run' button is highlighted, and the 'Run' tab is active. The output console shows the following sequence of events:

```
inread-2 Agregado: 3
Thread-1 Agregado: 2
Thread-2 Agregado: 3
Thread-1 Agregado: 2
Thread-2 Agregado: 3
Thread-2 Agregado: 3
Thread-2 Agregado: 3
Thread-1 Agregado: 2
Thread-1 Agregado: 2
Thread-1 Agregado: 2
1 2 3 1 3 2 1 2
3 1 2 3 1 3 2 1
2 2 1 3 3 1 3 2
Process finished with exit code 0
```