|  |  |  |
| --- | --- | --- |
| **Question 1** |  | 1 / 1 point |

A set can contain duplicate values as it is just a sequence of values.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | True | | |
|  |  | False | | |
| **Question 2** | | |  | 1 / 1 point | |

Map can contain duplicate keys which can point to different values.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | True | | |
|  |  | False | | |
| **Question 3** | | |  | 1 / 1 point | |

The best performing Map for search operation is

|  |  |  |  |
| --- | --- | --- | --- |
|  | HashMap as it has **a hashtable-based implementation** | | |
|  | Tree Map as it stores it elements in natural order | | |
|  | LinkHashMap as it stores order in which keys are inserted into it | | |
| **Question 4** | |  | 0 / 3 points | |

Write a program that takes the numbers 145623 to 145723 and displays the results in a table :

* Display the number
* Display the last two digits of the number
* Display the middle two digits of the number
* Add each of the digits together and display the last two digits of the sum

Example:

Number           last two digits               middle two digits          last digits of sum

145623        23         56       21

145624        24         56       22

Copy your code below.

public class hash {  
    public static int sumNumbers(int s) {   
       int sum = 0;   
       while (s != 0) {   
               sum += s % 10;   
               s = s / 10;   
       }   
       return sum;   
}  
      
      
    public static void main(String[] args) {   
       int numbers[] = {145623, 145723};   
  
       System.out.printf("%-20s%-20s%-20s%-20s\n",   
                   "number", "last two digits",   
                       "middle two digits", "last digits of sum");   
  
       for(int number: numbers) {   
               System.out.printf("%-20d%-20d%-20d%-20d\n",   
                       number, number % 100, (number / 100) % 100, sumNumbers(number) % 100);   
       }  
         
    }  
}

**This question has not been graded.**

**The correct answer is not displayed for Written Response type questions.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| [View Feedback](javascript://)   |  |  | | --- | --- | | [Hide Feedback](javascript://) | | |  |  | | **public** **class** HashAlgoEx {  **public** **static** **void** main(String[] args) {  System.***out***.println ("For numbers 145623 to 145624");  System.***out***.println ("Number           last two digits               middle two digits          last digits of sum");  **int** lastTwo;  **int** middleTwo;  **int** computedTwo;  **int** [] lastTwoArray = **new** **int**[100];  **int** [] midTwoArray = **new** **int**[100];  **int** [] sumArray = **new** **int**[100];  **for** (**int** i=145623; i < 145724; i++ ) {  lastTwo = i%100;  middleTwo = i/100%100;  computedTwo = ((i/100000) + (i/10000%10) + (i/1000%10) + (i/100%10) + (i/10%10) + (i%10))%100;  System.***out***.println (i +"        " + lastTwo + "         " +middleTwo + "       " + computedTwo);  }  System.***out***.println ("For 100 random numbers");  Random random = **new** Random();  System.***out***.println ("\n\nNumber           last two digits               middle two digits          last digits of sum");  **for** (**int** j=0; j < 100; j++ ) {  **int** i = random.nextInt(1000000) + 1;  lastTwo = i%100;  lastTwoArray[lastTwo]++;  middleTwo = i/100%100;  midTwoArray[middleTwo]++;  computedTwo = ((i/100000) + (i/10000%10) + (i/1000%10) + (i/100%10) + (i/10%10) + (i%10))%100;  sumArray[computedTwo]++;  System.***out***.println (i +"        " + lastTwo + "         " +middleTwo + "       " + computedTwo);  }  System.***out***.println("Distribution of array indexes");  **for** (**int** i = 0; i < 100; i++) {  System.***out***.println (i +"        " + lastTwoArray[i] + "         " +midTwoArray[i] + "       " + sumArray[i]);  }  } } | | |
| **Question 5** | |  | 0 / 1 point |

Now run the above program with 100 random numbers and copy the output below.

number              last two digits     middle two digits   last digits of sum    
427286              86                  72                  29                    
382508              8                   25                  26                    
277657              57                  76                  34                    
265486              86                  54                  31                    
349104              4                   91                  21                    
668595              95                  85                  39                    
998625              25                  86                  39                    
159129              29                  91                  27                    
381306              6                   13                  21                    
579699              99                  96                  45                    
159987              87                  99                  39                    
567362              62                  73                  29                    
661006              6                   10                  19                    
575264              64                  52                  29                    
616602              2                   66                  21                    
851373              73                  13                  27                    
932436              36                  24                  27                    
242532              32                  25                  18                    
104052              52                  40                  12                    
579405              5                   94                  30                    
439197              97                  91                  33                    
635539              39                  55                  31                    
422986              86                  29                  31                    
919871              71                  98                  35                    
953431              31                  34                  25                    
161505              5                   15                  18                    
528191              91                  81                  26                    
734680              80                  46                  28                    
546105              5                   61                  21                    
683530              30                  35                  25                    
104720              20                  47                  14                    
174935              35                  49                  29                    
297155              55                  71                  29                    
494583              83                  45                  33                    
797092              92                  70                  34                    
218783              83                  87                  29                    
442057              57                  20                  22                    
831875              75                  18                  32                    
794397              97                  43                  39                    
506922              22                  69                  24                    
252215              15                  22                  17                    
943041              41                  30                  21                    
764099              99                  40                  35                    
525093              93                  50                  24                    
450607              7                   6                   22                    
485929              29                  59                  37                    
470354              54                  3                   23                    
654881              81                  48                  32                    
271215              15                  12                  18                    
742364              64                  23                  26                    
808895              95                  88                  38                    
564245              45                  42                  26                    
481060              60                  10                  19                    
352875              75                  28                  30                    
439838              38                  98                  35                    
137402              2                   74                  17                    
397010              10                  70                  20                    
834203              3                   42                  20                    
402179              79                  21                  23                    
150060              60                  0                   12                    
246057              57                  60                  24                    
924277              77                  42                  31                    
549373              73                  93                  31                    
480450              50                  4                   21                    
137359              59                  73                  28                    
910407              7                   4                   21                    
854007              7                   40                  24                    
109218              18                  92                  21                    
837138              38                  71                  30                    
135597              97                  55                  30                    
591472              72                  14                  28                    
201944              44                  19                  20                    
963358              58                  33                  34                    
445151              51                  51                  20                    
127410              10                  74                  15                    
986931              31                  69                  36                    
546151              51                  61                  22                    
331185              85                  11                  21                    
401831              31                  18                  17                    
937841              41                  78                  32                    
932057              57                  20                  26                    
456164              64                  61                  26                    
223951              51                  39                  22                    
745888              88                  58                  40                    
545415              15                  54                  24                    
688257              57                  82                  36                    
366148              48                  61                  28                    
919502              2                   95                  26                    
758632              32                  86                  31                    
261090              90                  10                  18                    
172845              45                  28                  27                    
180727              27                  7                   25                    
264511              11                  45                  19                    
978382              82                  83                  37                    
680716              16                  7                   28                    
837095              95                  70                  32                    
388527              27                  85                  33                    
819981              81                  99                  36                    
980298              98                  2                   36                    
695312              12                  53                  26

**This question has not been graded.**

**The correct answer is not displayed for Written Response type questions.**

|  |
| --- |
| [View Feedback](javascript://) |
| **Question 6** | |  | 3 / 3 points |

Which statements are correct?

|  |  |
| --- | --- |
|  | Hashing is the process of converting a given key into another value. |
|  | A good hash function uses a **two-way** hashing algorithm, or in other words, the hash can be converted back into the original key. |
|  | A **hash function** is used to generate the new value according to a mathematical algorithm. |
|  | The result of a hash function is known as a **hash value** or simply, a **hash**. |
|  | **Hash Collision** is a phenomenon when two keys can generate the same hash. |