# Parte A:

1A- Obtener el ranking de las N franjas horarias que tengan más infracciones. El valor N es un dato de entrada.

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 1A- obtenerRankingFranjas(N) |
| **Funcionalidad** | Obtener las N franjas horarias en las que haya más accidentalidad. |
| **Especificación** | |
| **Entrada** | Franja horaria específica. |
| **Proceso** | Recorrer la cola de prioridad obteniendo las infracciones que están en la franja especificada. |
| **Salida** | Cantidad total de infracciones en la franja, el porcentaje de infracciones sin accidente, el porcentaje de infracciones con accidente y el valor total a pagar por las infracciones. |
| **Complejidad** | O(n^2) |

2A- Realizar el ordenamiento de las infracciones por Localización Geográfica (Xcoord, Ycoord).

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 2A- ordenarPorLocation(xCoord, yCoord) |
| **Funcionalidad** | Ordenar las infracciones según su loclización geográfica. |
| **Especificación** | |
| **Entrada** | Coordenada x y y de la localización geográfica de la infracción. |
| **Proceso** | Buscar las infracciones mediante la llave (coordenadas) en la tabla de Hash que tengan la localización dada y ordenarlas según el criterio. |
| **Salida** | Cantidad total de infracciones, porcentaje de infracciones sin accidente, porcentaje de infracciones con accidente, valor total a pagar por las infracciones, location, el Id de la dirección (AddressID) y el Id de la calle (StreetSegId). |
| **Complejidad** | O(N) |

3A- Buscar las infracciones por rango de fechas [Fecha Inicial (Año/Mes/Día), Fecha Final (Año/Mes/Día)].

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 3A- buscarPorFecha(fInicial, fFinal) |
| **Funcionalidad** | Dadas un par de fechas (inicial-final) se buscan las infracciones que se hicieron en ese rango de fechas. |
| **Especificación** | |
| **Entrada** | Fecha final (Año/Mes/Día) y Fecha final (Año/Mes/Día) |
| **Proceso** | Se hace un recorrido sobre la tabla de infracciones y se obtienen las que estén dentro del rango especificado para luego representarlas en un árbol balanceado por TicketIssueDate. |
| **Salida** | Árbol balanceado que muestra la fecha (Año/Mes/Día), el total de infracciones, su porcentaje de infracciones sin accidente, su porcentaje de infracciones con accidente y el valor total a pagar por las infracciones. |
| **Complejidad** | O(N) |

# Parte B:

1B- Obtener el ranking de las N tipos de infracción (ViolationCode) que tengan más infracciones. El valor N es un dato de entrada.

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 1B- rankingTiposInfraccion(N) |
| **Funcionalidad** | Dado un tipo de infracción (ViolationCode), obtener el ranking de la cantidad de infracciones que tienen ese código. |
| **Especificación** | |
| **Entrada** | Tipo de infracción (ViolationCode). |
| **Proceso** | Se recorre la tabla de infracciones y se toman las que tengan el código dado, para luego colocarlas en una Cola de Prioridad orientada a mayor por el número de infracciones de cada tipo de infracción. |
| **Salida** | Cola de Prioridad con las infracciones en donde se muestra para cada tipo de infracción su código, total de infracciones, su porcentaje de infracciones sin accidente, su porcentaje de infracciones con accidente y el valor total a pagar por las infracciones. |
| **Complejidad** | O(N+(ordenamiento)) |

2B- Realizar el ordenamiento de las infracciones por Localización Geográfica (Xcoord, Ycoord).

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 2B- ordenarPorLocation(xCoord, yCoord) |
| **Funcionalidad** | Ordenar las infracciones según su localización. |
| **Especificación** | |
| **Entrada** | Coordenada x y y de la localización por la que se quiere ordenar. |
| **Proceso** | Se recorre la tabla de infracciones usando las coordenadas como llave de búsqueda y el criterio de ordenamiento y se agrupan en un árbol balanceado por localización geográfica. |
| **Salida** | Árbol balanceado en donde se informa el total de infracciones, su porcentaje de infracciones sin accidente, su porcentaje de infracciones con accidente, el valor total a pagar por las infracciones, el location, el Id de la dirección (AddressID) y el Id de la calle (StreetSegId). |
| **Complejidad** | O(N+(ordenamiento)) |

3B- Buscar las franjas de fecha-hora donde se tiene un valor acumulado de infracciones en un rango dado [US$ valor inicial, US$ valor final].

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 3B- buscarPorRangoDinero(vInicial, vFinal) |
| **Funcionalidad** | Buscar las franjas de fecha-hora en donde hay un valor acumulado que esté dentro del rango de valores dados. |
| **Especificación** | |
| **Entrada** | Rango de valores acumulados (valorInicial, valorFinal). |
| **Proceso** | Se recorre la tabla sacando las infracciones que cumplen el rango de valor acumulado dado, y de éstas se toma su fecha y hora para informar. |
| **Salida** | Fechas – hora que tienen un valor acumulado de infracciones (US$) en el rango de consulta suministrado en donde se muestre también su valor acumulado de infracciones (US$), el total de infracciones, su porcentaje de infracciones sin accidente y su porcentaje de infracciones con accidente. |
| **Complejidad** | O(N) |

# Parte C:

1C- Obtener la información de una localización dada (AddressID).

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 1C-obtenerInfoLocation (AddressId). |
| **Funcionalidad** | Consultar la información de una localización dada por parámetro. |
| **Especificación** | |
| **Entrada** | Localización (AddresId) de la cual se quiere obtener la información. |
| **Proceso** | Se recorre la tabla sacando las infracciones que tengan el AddressId dado y de éstas se toma la información de salida. |
| **Salida** | Cantidad total de infracciones en la localización, el porcentaje de infracciones sin accidente, el porcentaje de infracciones con accidente, el valor total a pagar por las infracciones, y el Id de la calle (StreetSegId). |
| **Complejidad** | O(N+(ordenamiento)) |

2C- Obtener las infracciones en un rango de horas [HH:MM:SS inicial, HH:MM:SS final]

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 2C-obtenerInfraccionesRangoHoras (hInicial, hFinal) |
| **Funcionalidad** | Dado un rango de horas (inicial-final) mostrar la información general y por código de las infracciones que están dentro del rango dado. |
| **Especificación** | |
| **Entrada** | Rango de horas (HH:MM:SS inicial, HH:MM:SS final). |
| **Proceso** | Se recorre la tabla tomando las infracciones que están dentro del rango dado y saca de ellas su información general. |
| **Salida** | Información general: el total de infracciones, el porcentaje de infracciones sin accidente, el porcentaje de infracciones con accidente, y el valor total a pagar por las infracciones. Información por código (ViolationCode): Agrupar las infracciones obtenidas por código, informar el detalle de código y el total de infracciones. |
| **Complejidad** | O(N) |

3C – Obtener el ranking de las N localizaciones geográficas (Xcoord, Ycoord) con la mayor cantidad de infracciones.

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 3C-obtenerRankingLocation (xCoord, yCoord) |
| **Funcionalidad** | Obtiene las localizaciones en donde hay mayor cantidad de infracciones. |
| **Especificación** | |
| **Entrada** | Coordenadas de la localización (xCoord, yCoord). |
| **Proceso** | Se recorre la tabla tomando las infracciones que se encuentran en las coordenadas dadas por el cliente para ver cuáles localizaciones tienen más infracciones. |
| **Salida** | Reporte con el total de infracciones, su porcentaje de infracciones sin accidente, su porcentaje de infracciones con accidente, el location, el Id de la dirección (AddressID) y el Id de la calle (StreetSegId) |
| **Complejidad** | O(N) |

4C– Mostrar una gráfica ASCII con la información de las infracciones por código (ViolationCode).

|  |  |
| --- | --- |
| **Nombre del Requerimiento** | 4C- graficaInfraccionesPorCodigo |
| **Funcionalidad** | Muestra al cliente una gráfica ASCII con el porcentaje de infracciones que cumplen con un ViolationCode específico. |
| **Especificación** | |
| **Entrada** | Tabla de infracciones a comprobar. |
| **Proceso** | Se recorre la tabla almacenando/clasificando las infracciones con tipo de código.  Se introducen en la gráfica los datos. |
| **Salida** | Gráfica ASCII con la representación de los datos. |
| **Complejidad** | O(N) |

La gráfica debe ser de la forma:

TXXX | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TYYY | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TZZZ | \*\*\*\*\*\*\*\*\*\*\*\*\*

…

En donde cada \* representa un porcentaje K% (a definir). En la gráfica debe indicarse el porcentaje (%) al que corresponde cada \*.