**RETO 1.1**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | F | matemáticas | 4.1 | | Nicolás | M | matemáticas | 4.6 | | Alexandra | F | idiomas | 3.6 | | Nicolás | M | idiomas | 3.1 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Cuántos exámenes tienen una calificación Excelente?  ¿Cuál es la materia con el mejor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia historia?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | Historia | 1 | | Literatura | 2 | | Biología | 3 |  |  |  | | --- | --- | | Género | Identificador | | M | 0 | | F | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 2.3 1.0 0.0 2.0 0.3 1.0 0.0 3.0 3.8 2.0 0.0 1.0 1.4 2.0 0.0 2.0 0.1 2.0 0.0 3.0 3.3 3.0 0.0 1.0 2.9 3.0 0.0 2.0 3.7 3.0 0.0 3.0 3.2 4.0 1.0 1.0 4.5 4.0 1.0 2.0 2.5 4.0 1.0 3.0 5.0 5.0 1.0 1.0 3.7 5.0 1.0 2.0 3.0 5.0 1.0 3.0 0.5 6.0 1.0 1.0 1.8 6.0 1.0 2.0 4.4 6.0 1.0 3.0 2.6 | | Salida del programa | | 0.61 1 biologia María |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.2**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.2 | | Armando | m | matemáticas | 9.7 | | Marcela | f | idiomas | 8.4 | | Armando | m | idiomas | 6.0 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Cuántos exámenes tienen una calificación Sobresaliente?  ¿Cuál es la materia con el peor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia literatura?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | literatura | 1 | | biologia | 2 | | Geografía | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 4.9 1.0 0.0 2.0 2.0 1.0 0.0 3.0 4.6 2.0 0.0 1.0 2.2 2.0 0.0 2.0 5.6 2.0 0.0 3.0 6.3 3.0 0.0 1.0 3.5 3.0 0.0 2.0 2.0 3.0 0.0 3.0 0.9 4.0 1.0 1.0 6.2 4.0 1.0 2.0 9.7 4.0 1.0 3.0 5.1 5.0 1.0 1.0 8.7 5.0 1.0 2.0 5.9 5.0 1.0 3.0 7.8 6.0 1.0 1.0 6.3 6.0 1.0 2.0 5.9 6.0 1.0 3.0 2.7 | | Salida del programa | | 0.33 1 Geografía Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.3**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 82.7 | | Armando | m | matemáticas | 92.7 | | Alexandra | f | idiomas | 82.7 | | Armando | m | idiomas | 74.3 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Cuántos exámenes tienen una calificación Regular?  ¿Cuál es la materia con el mejor desempeño promedio para el género femenino?  ¿Cuál es el estudiante con el mejor desempeño para la materia biología?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | biologia | 1 | | Geografía | 2 | | matemáticas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 13.3 1.0 0.0 2.0 22.0 1.0 0.0 3.0 22.2 2.0 0.0 1.0 78.6 2.0 0.0 2.0 2.6 2.0 0.0 3.0 91.8 3.0 0.0 1.0 16.3 3.0 0.0 2.0 57.1 3.0 0.0 3.0 74.9 4.0 1.0 1.0 4.0 4.0 1.0 2.0 36.6 4.0 1.0 3.0 19.2 5.0 1.0 1.0 18.2 5.0 1.0 2.0 2.9 5.0 1.0 3.0 84.6 6.0 1.0 1.0 84.3 6.0 1.0 2.0 63.6 6.0 1.0 3.0 58.5 | | Salida del programa | | 0.33 3 matemáticas Alexandra |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.4**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 4.2 | | Armando | m | matemáticas | 4.9 | | María | f | idiomas | 3.7 | | Armando | m | idiomas | 2.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Cuántos exámenes tienen una calificación Insuficiente?  ¿Cuál es la materia con el peor desempeño promedio para el género masculino?  ¿Cuál es el estudiante con el mejor desempeño para la materia geografía?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | Geografía | 1 | | matemáticas | 2 | | informática | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.6 1.0 0.0 2.0 3.5 1.0 0.0 3.0 1.3 2.0 0.0 1.0 0.3 2.0 0.0 2.0 4.4 2.0 0.0 3.0 2.0 3.0 0.0 1.0 4.2 3.0 0.0 2.0 3.9 3.0 0.0 3.0 3.2 4.0 1.0 1.0 4.3 4.0 1.0 2.0 2.7 4.0 1.0 3.0 3.9 5.0 1.0 1.0 1.8 5.0 1.0 2.0 1.8 5.0 1.0 3.0 0.2 6.0 1.0 1.0 1.6 6.0 1.0 2.0 0.4 6.0 1.0 3.0 2.4 | | Salida del programa | | 0.44 7 Geografía María |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.5**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.8 | | Armando | m | matemáticas | 9.5 | | Marcela | f | idiomas | 9.0 | | Armando | m | idiomas | 6.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Cuántos exámenes tienen una calificación Deficiente?  ¿Cuál género tiene un mejor desempeño promedio?  ¿Cuál es el estudiante con el mejor desempeño para la materia matemáticas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | matemáticas | 1 | | informática | 2 | | física | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 4.6 1.0 0.0 2.0 9.5 1.0 0.0 3.0 9.9 2.0 0.0 1.0 9.8 2.0 0.0 2.0 1.6 2.0 0.0 3.0 7.2 3.0 0.0 1.0 1.8 3.0 0.0 2.0 2.3 3.0 0.0 3.0 7.9 4.0 1.0 1.0 3.4 4.0 1.0 2.0 1.0 4.0 1.0 3.0 4.4 5.0 1.0 1.0 1.8 5.0 1.0 2.0 9.4 5.0 1.0 3.0 0.2 6.0 1.0 1.0 2.0 6.0 1.0 2.0 7.8 6.0 1.0 3.0 5.3 | | Salida del programa | | 0.39 7 m Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.6**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 84.6 | | Nicolás | m | matemáticas | 98.3 | | Marcela | f | idiomas | 83.5 | | Nicolás | m | idiomas | 68.2 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Qué porcentaje de los exámenes fueron Excelente?  ¿Cuál es la materia con el mayor número de exámenes aprobados?  ¿Cuál es el estudiante con el mejor desempeño para la materia informática?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | informática | 1 | | física | 2 | | química | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 62.9 1.0 0.0 2.0 62.4 1.0 0.0 3.0 95.7 2.0 0.0 1.0 38.7 2.0 0.0 2.0 53.9 2.0 0.0 3.0 75.7 3.0 0.0 1.0 71.0 3.0 0.0 2.0 71.4 3.0 0.0 3.0 24.2 4.0 1.0 1.0 48.3 4.0 1.0 2.0 14.7 4.0 1.0 3.0 30.0 5.0 1.0 1.0 100.0 5.0 1.0 2.0 0.8 5.0 1.0 3.0 3.7 6.0 1.0 1.0 80.5 6.0 1.0 2.0 75.3 6.0 1.0 3.0 55.2 | | Salida del programa | | 0.50 0.11 química Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.7**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de l siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 4.4 | | Nicolás | m | matemáticas | 4.9 | | María | f | idiomas | 4.0 | | Nicolás | m | idiomas | 2.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el porcentaje de aprobación para todos los exámenes presentados por el grupo?  ¿Qué porcentaje de los exámenes fueron Sobresalientes?  ¿Cuál es la materia con el mayor número de exámenes reprobados  ¿Cuál es el estudiante con el mejor desempeño para la materia física?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | física | 1 | | química | 2 | | idiomas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.3 1.0 0.0 2.0 0.9 1.0 0.0 3.0 1.1 2.0 0.0 1.0 4.3 2.0 0.0 2.0 2.6 2.0 0.0 3.0 4.8 3.0 0.0 1.0 0.2 3.0 0.0 2.0 0.1 3.0 0.0 3.0 3.8 4.0 1.0 1.0 1.5 4.0 1.0 2.0 3.8 4.0 1.0 3.0 2.1 5.0 1.0 1.0 1.4 5.0 1.0 2.0 0.8 5.0 1.0 3.0 2.7 6.0 1.0 1.0 0.9 6.0 1.0 2.0 3.6 6.0 1.0 3.0 1.9 | | Salida del programa | | 0.39 0.22 idiomas Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.8**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.6 | | Daniel | m | matemáticas | 9.5 | | Marcela | f | idiomas | 8.6 | | Daniel | m | idiomas | 7.6 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Cuántos exámenes tienen una calificación Excelente?  ¿Cuál es la materia con el peor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia biología?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | biologia | 1 | | Geografía | 2 | | matemáticas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.4 1.0 0.0 2.0 0.0 1.0 0.0 3.0 4.3 2.0 0.0 1.0 5.5 2.0 0.0 2.0 5.5 2.0 0.0 3.0 9.2 3.0 0.0 1.0 9.3 3.0 0.0 2.0 6.3 3.0 0.0 3.0 8.9 4.0 1.0 1.0 7.2 4.0 1.0 2.0 3.7 4.0 1.0 3.0 9.7 5.0 1.0 1.0 2.3 5.0 1.0 2.0 8.1 5.0 1.0 3.0 9.1 6.0 1.0 1.0 2.2 6.0 1.0 2.0 8.3 6.0 1.0 3.0 9.8 | | Salida del programa | | 6.16 5 biologia Daniel |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.9**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 88.4 | | Nicolás | m | matemáticas | 99.2 | | María | f | idiomas | 81.7 | | Nicolás | m | idiomas | 70.2 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Cuántos exámenes tienen una calificación Sobresaliente?  ¿Cuál es la materia con el mejor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia geografía?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | Geografía | 1 | | matemáticas | 2 | | informática | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 81.8 1.0 0.0 2.0 18.7 1.0 0.0 3.0 95.9 2.0 0.0 1.0 97.5 2.0 0.0 2.0 66.3 2.0 0.0 3.0 41.1 3.0 0.0 1.0 77.4 3.0 0.0 2.0 45.1 3.0 0.0 3.0 74.6 4.0 1.0 1.0 41.8 4.0 1.0 2.0 33.3 4.0 1.0 3.0 51.9 5.0 1.0 1.0 37.0 5.0 1.0 2.0 32.6 5.0 1.0 3.0 46.8 6.0 1.0 1.0 0.6 6.0 1.0 2.0 89.1 6.0 1.0 3.0 67.2 | | Salida del programa | | 55.48 2 informática Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.10**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 4.1 | | Daniel | m | matemáticas | 4.6 | | Alexandra | f | idiomas | 3.5 | | Daniel | m | idiomas | 3.4 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Cuántos exámenes tienen una calificación Regular?  ¿Cuál es la materia con el peor desempeño promedio para el género masculino?  ¿Cuál es el estudiante con el mejor desempeño para la materia historia?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | historia | 1 | | literatura | 2 | | biologia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 4.8 1.0 0.0 2.0 1.8 1.0 0.0 3.0 3.9 2.0 0.0 1.0 2.8 2.0 0.0 2.0 2.4 2.0 0.0 3.0 0.0 3.0 0.0 1.0 1.1 3.0 0.0 2.0 0.4 3.0 0.0 3.0 2.1 4.0 1.0 1.0 0.3 4.0 1.0 2.0 3.0 4.0 1.0 3.0 1.5 5.0 1.0 1.0 1.1 5.0 1.0 2.0 4.9 5.0 1.0 3.0 0.5 6.0 1.0 1.0 0.7 6.0 1.0 2.0 2.0 6.0 1.0 3.0 0.4 | | Salida del programa | | 1.87 2 literatura Armando |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.11**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 8.9 | | Nicolás | m | matemáticas | 9.1 | | María | f | idiomas | 8.5 | | Nicolás | m | idiomas | 6.2 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Cuántos exámenes tienen una calificación Insuficiente?  ¿Cuál es la materia con el mejor desempeño promedio para el género femenino?  ¿Cuál es el estudiante con el mejor desempeño para la materia literatura?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | literatura | 1 | | biologia | 2 | | Geografía | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 5.6 1.0 0.0 2.0 0.4 1.0 0.0 3.0 7.8 2.0 0.0 1.0 8.9 2.0 0.0 2.0 0.5 2.0 0.0 3.0 1.5 3.0 0.0 1.0 8.1 3.0 0.0 2.0 4.0 3.0 0.0 3.0 7.9 4.0 1.0 1.0 4.8 4.0 1.0 2.0 9.3 4.0 1.0 3.0 4.8 5.0 1.0 1.0 5.8 5.0 1.0 2.0 3.3 5.0 1.0 3.0 4.7 6.0 1.0 1.0 6.3 6.0 1.0 2.0 5.5 6.0 1.0 3.0 8.4 | | Salida del programa | | 5.42 8 biologia Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.12**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 83.5 | | Nicolás | m | matemáticas | 99.3 | | Alexandra | f | idiomas | 84.2 | | Nicolás | m | idiomas | 76.4 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Cuántos exámenes tienen una calificación Deficiente?  ¿Cuál es la materia con el mayor número de exámenes aprobados?  ¿Cuál es el estudiante con el mejor desempeño para la materia física?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | física | 1 | | química | 2 | | idiomas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 51.3 1.0 0.0 2.0 7.2 1.0 0.0 3.0 67.3 2.0 0.0 1.0 25.6 2.0 0.0 2.0 63.7 2.0 0.0 3.0 80.9 3.0 0.0 1.0 64.5 3.0 0.0 2.0 86.5 3.0 0.0 3.0 82.3 4.0 1.0 1.0 10.2 4.0 1.0 2.0 11.7 4.0 1.0 3.0 39.5 5.0 1.0 1.0 68.0 5.0 1.0 2.0 44.1 5.0 1.0 3.0 66.8 6.0 1.0 1.0 20.2 6.0 1.0 2.0 71.7 6.0 1.0 3.0 75.0 | | Salida del programa | | 52.03 5 idiomas Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.13**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela priMaría desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 3.9 | | Daniel | m | matemáticas | 5.0 | | Alexandra | f | idiomas | 3.8 | | Daniel | m | idiomas | 2.9 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Qué porcentaje de los exámenes fueron Excelente?  ¿Cuál género tiene un mejor desempeño promedio?  ¿Cuál es el estudiante con el mejor desempeño para la materia química?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | química | 1 | | idiomas | 2 | | historia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.3 1.0 0.0 2.0 1.6 1.0 0.0 3.0 0.6 2.0 0.0 1.0 4.7 2.0 0.0 2.0 0.1 2.0 0.0 3.0 4.7 3.0 0.0 1.0 1.2 3.0 0.0 2.0 1.4 3.0 0.0 3.0 1.0 4.0 1.0 1.0 0.4 4.0 1.0 2.0 0.8 4.0 1.0 3.0 1.7 5.0 1.0 1.0 0.2 5.0 1.0 2.0 2.6 5.0 1.0 3.0 0.2 6.0 1.0 1.0 0.5 6.0 1.0 2.0 2.1 6.0 1.0 3.0 1.0 | | Salida del programa | | 1.45 0.11 m Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.14**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.8 | | Armando | m | matemáticas | 9.3 | | Marcela | f | idiomas | 8.8 | | Armando | m | idiomas | 6.8 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es el desempeño promedio de todo el grupo?  ¿Qué porcentaje de los exámenes fueron Regulares?  ¿Cuál es la materia con el mayor número de exámenes reprobados  ¿Cuál es el estudiante con el mejor desempeño para la materia matemáticas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | matemáticas | 1 | | informática | 2 | | física | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 4.3 1.0 0.0 2.0 1.6 1.0 0.0 3.0 7.0 2.0 0.0 1.0 3.7 2.0 0.0 2.0 7.0 2.0 0.0 3.0 7.5 3.0 0.0 1.0 4.2 3.0 0.0 2.0 2.8 3.0 0.0 3.0 0.6 4.0 1.0 1.0 1.8 4.0 1.0 2.0 9.2 4.0 1.0 3.0 4.2 5.0 1.0 1.0 7.2 5.0 1.0 2.0 9.9 5.0 1.0 3.0 8.9 6.0 1.0 1.0 1.3 6.0 1.0 2.0 8.6 6.0 1.0 3.0 0.2 | | Salida del programa | | 5.00 0.22 física Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.15**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 85.2 | | Armando | m | matemáticas | 98.8 | | María | f | idiomas | 89.1 | | Armando | m | idiomas | 73.4 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Cuántos exámenes tienen una calificación Excelente?  ¿Cuál es la materia con el mejor desempeño promedio para el genero femenino?  ¿Cuál es el estudiante con el mejor desempeño para la materia Geografía?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | Geografía | 1 | | matemáticas | 2 | | informática | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 92.4 1.0 0.0 2.0 45.6 1.0 0.0 3.0 53.6 2.0 0.0 1.0 83.2 2.0 0.0 2.0 36.6 2.0 0.0 3.0 99.0 3.0 0.0 1.0 80.3 3.0 0.0 2.0 88.1 3.0 0.0 3.0 35.9 4.0 1.0 1.0 66.8 4.0 1.0 2.0 52.0 4.0 1.0 3.0 91.4 5.0 1.0 1.0 34.1 5.0 1.0 2.0 90.5 5.0 1.0 3.0 75.5 6.0 1.0 1.0 62.6 6.0 1.0 2.0 87.9 6.0 1.0 3.0 87.6 | | Salida del programa | | 461.63 4 informática Armando |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.16**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 3.6 | | Nicolás | m | matemáticas | 4.7 | | María | f | idiomas | 4.1 | | Nicolás | m | idiomas | 2.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Cuántos exámenes tienen una calificación Sobresaliente?  ¿Cuál es la materia con el peor desempeño promedio para el género masculino?  ¿Cuál es el estudiante con el mejor desempeño para la materia biología?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | biologia | 1 | | Geografía | 2 | | matemáticas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 2.1 1.0 0.0 2.0 2.7 1.0 0.0 3.0 2.2 2.0 0.0 1.0 0.6 2.0 0.0 2.0 3.1 2.0 0.0 3.0 4.9 3.0 0.0 1.0 5.0 3.0 0.0 2.0 1.2 3.0 0.0 3.0 3.9 4.0 1.0 1.0 3.7 4.0 1.0 2.0 4.6 4.0 1.0 3.0 0.4 5.0 1.0 1.0 2.7 5.0 1.0 2.0 3.6 5.0 1.0 3.0 4.4 6.0 1.0 1.0 0.1 6.0 1.0 2.0 0.9 6.0 1.0 3.0 3.6 | | Salida del programa | | 2.41 5 Geografía Daniel |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.17**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.4 | | Armando | m | matemáticas | 9.3 | | Marcela | f | idiomas | 8.3 | | Armando | m | idiomas | 6.2 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Cuántos exámenes tienen una calificación Regular?  ¿Cuál es la materia con el mejor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia literatura?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | literatura | 1 | | biologia | 2 | | Geografía | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 9.2 1.0 0.0 2.0 2.3 1.0 0.0 3.0 4.3 2.0 0.0 1.0 2.5 2.0 0.0 2.0 8.9 2.0 0.0 3.0 7.2 3.0 0.0 1.0 1.9 3.0 0.0 2.0 7.6 3.0 0.0 3.0 2.3 4.0 1.0 1.0 4.8 4.0 1.0 2.0 3.6 4.0 1.0 3.0 1.9 5.0 1.0 1.0 8.0 5.0 1.0 2.0 4.9 5.0 1.0 3.0 9.7 6.0 1.0 1.0 4.1 6.0 1.0 2.0 1.3 6.0 1.0 3.0 0.1 | | Salida del programa | | 8.61 3 literatura Armando |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.18**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 85.3 | | Daniel | m | matemáticas | 93.4 | | María | f | idiomas | 87.7 | | Daniel | m | idiomas | 75.6 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Cuántos exámenes tienen una calificación Insuficiente?  ¿Cuál es la materia con el peor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia historia?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | historia | 1 | | literatura | 2 | | biologia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 72.2 1.0 0.0 2.0 13.9 1.0 0.0 3.0 9.3 2.0 0.0 1.0 51.8 2.0 0.0 2.0 28.0 2.0 0.0 3.0 73.8 3.0 0.0 1.0 76.5 3.0 0.0 2.0 85.9 3.0 0.0 3.0 60.3 4.0 1.0 1.0 39.0 4.0 1.0 2.0 94.8 4.0 1.0 3.0 18.5 5.0 1.0 1.0 69.1 5.0 1.0 2.0 50.9 5.0 1.0 3.0 54.1 6.0 1.0 1.0 15.6 6.0 1.0 2.0 4.5 6.0 1.0 3.0 50.9 | | Salida del programa | | 739.30 5 biologia Daniel |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.19**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 3.8 | | Armando | m | matemáticas | 4.8 | | María | f | idiomas | 3.9 | | Armando | m | idiomas | 2.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Cuántos exámenes tienen una calificación Deficiente?  ¿Cuál es la materia con el mayor número de exámenes reprobados  ¿Cuál es el estudiante con el mejor desempeño para la materia informática?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | informática | 1 | | física | 2 | | química | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 4.7 1.0 0.0 2.0 2.5 1.0 0.0 3.0 0.5 2.0 0.0 1.0 0.2 2.0 0.0 2.0 4.9 2.0 0.0 3.0 4.7 3.0 0.0 1.0 3.5 3.0 0.0 2.0 4.4 3.0 0.0 3.0 0.9 4.0 1.0 1.0 1.0 4.0 1.0 2.0 2.5 4.0 1.0 3.0 3.2 5.0 1.0 1.0 1.1 5.0 1.0 2.0 1.8 5.0 1.0 3.0 0.1 6.0 1.0 1.0 0.6 6.0 1.0 2.0 2.9 6.0 1.0 3.0 0.3 | | Salida del programa | | 2.78 7 química Armando |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.20**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 8.3 | | Daniel | m | matemáticas | 9.7 | | María | f | idiomas | 8.6 | | Daniel | m | idiomas | 7.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Qué porcentaje de los exámenes fueron Sobresalientes?  ¿Cuál género tiene un mejor desempeño promedio?  ¿Cuál es el estudiante con el mejor desempeño para la materia idiomas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | idiomas | 1 | | historia | 2 | | literatura | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 8.3 1.0 0.0 2.0 0.3 1.0 0.0 3.0 4.2 2.0 0.0 1.0 7.3 2.0 0.0 2.0 8.9 2.0 0.0 3.0 1.5 3.0 0.0 1.0 5.5 3.0 0.0 2.0 4.1 3.0 0.0 3.0 8.1 4.0 1.0 1.0 4.7 4.0 1.0 2.0 2.7 4.0 1.0 3.0 1.8 5.0 1.0 1.0 10.0 5.0 1.0 2.0 3.7 5.0 1.0 3.0 3.3 6.0 1.0 1.0 0.4 6.0 1.0 2.0 5.6 6.0 1.0 3.0 7.2 | | Salida del programa | | 8.24 0.17 m Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.21**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 81.2 | | Nicolás | m | matemáticas | 92.9 | | María | f | idiomas | 86.1 | | Nicolás | m | idiomas | 68.9 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuál es la varianza de las calificaciones para todo el grupo?  ¿Qué porcentaje de los exámenes fueron Regulares?  ¿Cuál es la materia con el mayor número de exámenes aprobados?  ¿Cuál es el estudiante con el mejor desempeño para la materia química?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | química | 1 | | idiomas | 2 | | historia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 76.3 1.0 0.0 2.0 54.4 1.0 0.0 3.0 17.5 2.0 0.0 1.0 3.6 2.0 0.0 2.0 76.7 2.0 0.0 3.0 46.4 3.0 0.0 1.0 56.0 3.0 0.0 2.0 50.0 3.0 0.0 3.0 5.2 4.0 1.0 1.0 88.5 4.0 1.0 2.0 91.1 4.0 1.0 3.0 58.2 5.0 1.0 1.0 22.2 5.0 1.0 2.0 19.1 5.0 1.0 3.0 11.9 6.0 1.0 1.0 74.0 6.0 1.0 2.0 13.7 6.0 1.0 3.0 88.8 | | Salida del programa | | 910.37 0.17 historia María |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.22**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 4.0 | | Nicolás | m | matemáticas | 4.9 | | María | f | idiomas | 4.2 | | Nicolás | m | idiomas | 3.0 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Excelente?  ¿Cuál es la materia con el peor desempeño promedio para el género masculino?  ¿Cuál es el estudiante con el mejor desempeño para la materia literatura?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | literatura | 1 | | biologia | 2 | | Geografía | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 2.6 1.0 0.0 2.0 2.9 1.0 0.0 3.0 2.0 2.0 0.0 1.0 1.4 2.0 0.0 2.0 3.1 2.0 0.0 3.0 2.5 3.0 0.0 1.0 3.5 3.0 0.0 2.0 2.7 3.0 0.0 3.0 3.7 4.0 1.0 1.0 2.6 4.0 1.0 2.0 3.3 4.0 1.0 3.0 4.1 5.0 1.0 1.0 4.5 5.0 1.0 2.0 3.8 5.0 1.0 3.0 4.7 6.0 1.0 1.0 0.6 6.0 1.0 2.0 3.5 6.0 1.0 3.0 2.8 | | Salida del programa | | 9 1 literatura Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.23**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 8.3 | | Armando | m | matemáticas | 9.3 | | María | f | idiomas | 8.8 | | Armando | m | idiomas | 6.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Sobresaliente?  ¿Cuál es la materia con el mejor desempeño promedio para el género femenino?  ¿Cuál es el estudiante con el mejor desempeño para la materia historia?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | historia | 1 | | literatura | 2 | | biologia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 6.0 1.0 0.0 2.0 2.2 1.0 0.0 3.0 8.3 2.0 0.0 1.0 6.7 2.0 0.0 2.0 5.7 2.0 0.0 3.0 1.7 3.0 0.0 1.0 0.2 3.0 0.0 2.0 4.6 3.0 0.0 3.0 6.1 4.0 1.0 1.0 0.3 4.0 1.0 2.0 6.8 4.0 1.0 3.0 5.6 5.0 1.0 1.0 1.5 5.0 1.0 2.0 4.6 5.0 1.0 3.0 9.4 6.0 1.0 1.0 0.2 6.0 1.0 2.0 0.1 6.0 1.0 3.0 9.6 | | Salida del programa | | 7 1 biologia Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.24**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 86.9 | | Daniel | m | matemáticas | 91.0 | | María | f | idiomas | 80.7 | | Daniel | m | idiomas | 71.9 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Regular?  ¿Cuál es la materia con el peor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia geografía?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | Geografía | 1 | | matemáticas | 2 | | informática | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 21.7 1.0 0.0 2.0 44.8 1.0 0.0 3.0 86.8 2.0 0.0 1.0 73.6 2.0 0.0 2.0 23.3 2.0 0.0 3.0 59.5 3.0 0.0 1.0 10.7 3.0 0.0 2.0 39.7 3.0 0.0 3.0 19.4 4.0 1.0 1.0 19.1 4.0 1.0 2.0 55.4 4.0 1.0 3.0 37.5 5.0 1.0 1.0 69.7 5.0 1.0 2.0 31.7 5.0 1.0 3.0 67.1 6.0 1.0 1.0 14.3 6.0 1.0 2.0 7.4 6.0 1.0 3.0 36.1 | | Salida del programa | | 11 3 matemáticas Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.25**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 4.4 | | Armando | m | matemáticas | 4.6 | | María | f | idiomas | 4.1 | | Armando | m | idiomas | 3.1 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Insuficiente?  ¿Cuál es la materia con el mejor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia biología?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | biologia | 1 | | Geografía | 2 | | matemáticas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.9 1.0 0.0 2.0 2.7 1.0 0.0 3.0 4.7 2.0 0.0 1.0 4.2 2.0 0.0 2.0 2.3 2.0 0.0 3.0 1.9 3.0 0.0 1.0 1.2 3.0 0.0 2.0 2.9 3.0 0.0 3.0 0.2 4.0 1.0 1.0 0.6 4.0 1.0 2.0 1.6 4.0 1.0 3.0 4.2 5.0 1.0 1.0 3.7 5.0 1.0 2.0 2.5 5.0 1.0 3.0 0.3 6.0 1.0 1.0 5.0 6.0 1.0 2.0 3.1 6.0 1.0 3.0 2.0 | | Salida del programa | | 9 7 biologia Alexandra |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.26**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.6 | | Daniel | m | matemáticas | 9.0 | | Marcela | f | idiomas | 8.3 | | Daniel | m | idiomas | 6.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Qué porcentaje de los exámenes fueron Excelente?  ¿Cuál es la materia con el mayor número de exámenes reprobados  ¿Cuál es el estudiante con el mejor desempeño para la materia idiomas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | idiomas | 1 | | historia | 2 | | literatura | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 5.7 1.0 0.0 2.0 0.2 1.0 0.0 3.0 5.0 2.0 0.0 1.0 7.5 2.0 0.0 2.0 9.9 2.0 0.0 3.0 3.5 3.0 0.0 1.0 2.2 3.0 0.0 2.0 5.5 3.0 0.0 3.0 2.2 4.0 1.0 1.0 7.9 4.0 1.0 2.0 2.2 4.0 1.0 3.0 9.9 5.0 1.0 1.0 9.3 5.0 1.0 2.0 9.8 5.0 1.0 3.0 5.8 6.0 1.0 1.0 9.5 6.0 1.0 2.0 7.2 6.0 1.0 3.0 6.6 | | Salida del programa | | 9 0.28 literatura Alexandra |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.27**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 87.7 | | Nicolás | m | matemáticas | 90.1 | | Marcela | f | idiomas | 87.8 | | Nicolás | m | idiomas | 75.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Qué porcentaje de los exámenes fueron Sobresalientes?  ¿Cuál es la materia con el mayor número de exámenes aprobados?  ¿Cuál es el estudiante con el mejor desempeño para la materia matemáticas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | matemáticas | 1 | | informática | 2 | | física | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 41.1 1.0 0.0 2.0 87.9 1.0 0.0 3.0 51.3 2.0 0.0 1.0 24.4 2.0 0.0 2.0 97.4 2.0 0.0 3.0 2.4 3.0 0.0 1.0 51.7 3.0 0.0 2.0 85.9 3.0 0.0 3.0 92.1 4.0 1.0 1.0 0.3 4.0 1.0 2.0 44.6 4.0 1.0 3.0 26.5 5.0 1.0 1.0 64.9 5.0 1.0 2.0 66.6 5.0 1.0 3.0 57.6 6.0 1.0 1.0 28.7 6.0 1.0 2.0 92.5 6.0 1.0 3.0 40.6 | | Salida del programa | | 10 0.11 física Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.28**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 4.2 | | Armando | m | matemáticas | 4.6 | | Alexandra | f | idiomas | 4.4 | | Armando | m | idiomas | 3.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos exámenes tienen una calificación menor a la del promedio del grupo?  ¿Qué porcentaje de los exámenes fueron Regulares?  ¿Cuál género tiene un mejor desempeño promedio?  ¿Cuál es el estudiante con el mejor desempeño para la materia informática?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | informática | 1 | | física | 2 | | química | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.1 1.0 0.0 2.0 0.7 1.0 0.0 3.0 0.7 2.0 0.0 1.0 3.4 2.0 0.0 2.0 1.1 2.0 0.0 3.0 2.8 3.0 0.0 1.0 4.0 3.0 0.0 2.0 3.3 3.0 0.0 3.0 3.4 4.0 1.0 1.0 3.9 4.0 1.0 2.0 3.6 4.0 1.0 3.0 4.2 5.0 1.0 1.0 5.0 5.0 1.0 2.0 0.9 5.0 1.0 3.0 4.9 6.0 1.0 1.0 4.0 6.0 1.0 2.0 3.4 6.0 1.0 3.0 3.1 | | Salida del programa | | 6 0.06 f Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.29**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | maria | f | matemáticas | 8.2 | | Daniel | m | matemáticas | 9.8 | | María | f | idiomas | 8.8 | | Daniel | m | idiomas | 7.1 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Excelente?  ¿Cuál género tiene un mejor desempeño promedio?  ¿Cuál es el estudiante con el mejor desempeño para la materia física?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | física | 1 | | química | 2 | | idiomas | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 4.0 1.0 0.0 2.0 9.8 1.0 0.0 3.0 0.7 2.0 0.0 1.0 0.6 2.0 0.0 2.0 1.1 2.0 0.0 3.0 9.9 3.0 0.0 1.0 4.9 3.0 0.0 2.0 8.1 3.0 0.0 3.0 8.9 4.0 1.0 1.0 5.7 4.0 1.0 2.0 4.7 4.0 1.0 3.0 5.6 5.0 1.0 1.0 0.7 5.0 1.0 2.0 3.0 5.0 1.0 3.0 0.0 6.0 1.0 1.0 0.0 6.0 1.0 2.0 8.5 6.0 1.0 3.0 0.3 | | Salida del programa | | 9 2 m María |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.30**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 81.8 | | Nicolás | m | matemáticas | 98.8 | | María | f | idiomas | 87.8 | | Nicolás | m | idiomas | 66.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Sobresaliente?  ¿Cuál es la materia con el mayor número de exámenes aprobados?  ¿Cuál es el estudiante con el mejor desempeño para la materia idiomas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | idiomas | 1 | | historia | 2 | | literatura | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 29.1 1.0 0.0 2.0 92.0 1.0 0.0 3.0 31.5 2.0 0.0 1.0 64.9 2.0 0.0 2.0 53.1 2.0 0.0 3.0 37.9 3.0 0.0 1.0 16.0 3.0 0.0 2.0 95.5 3.0 0.0 3.0 54.0 4.0 1.0 1.0 59.2 4.0 1.0 2.0 26.2 4.0 1.0 3.0 50.7 5.0 1.0 1.0 80.7 5.0 1.0 2.0 69.7 5.0 1.0 3.0 54.7 6.0 1.0 1.0 46.1 6.0 1.0 2.0 51.0 6.0 1.0 3.0 46.1 | | Salida del programa | | 8 1 historia Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.31**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 4.2 | | Armando | m | matemáticas | 4.5 | | Marcela | f | idiomas | 3.8 | | Armando | m | idiomas | 2.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Regular?  ¿Cuál es la materia con el mayor número de exámenes reprobados  ¿Cuál es el estudiante con el mejor desempeño para la materia química?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | química | 1 | | idiomas | 2 | | historia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.1 1.0 0.0 2.0 1.8 1.0 0.0 3.0 3.3 2.0 0.0 1.0 4.3 2.0 0.0 2.0 1.6 2.0 0.0 3.0 4.3 3.0 0.0 1.0 3.8 3.0 0.0 2.0 0.7 3.0 0.0 3.0 1.8 4.0 1.0 1.0 0.6 4.0 1.0 2.0 2.5 4.0 1.0 3.0 4.0 5.0 1.0 1.0 0.5 5.0 1.0 2.0 3.8 5.0 1.0 3.0 1.2 6.0 1.0 1.0 2.6 6.0 1.0 2.0 4.6 6.0 1.0 3.0 0.1 | | Salida del programa | | 9 1 historia Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.32**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 8.3 | | Daniel | m | matemáticas | 9.2 | | Marcela | f | idiomas | 8.1 | | Daniel | m | idiomas | 6.8 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Cuántos exámenes tienen una calificación Deficiente?  ¿Cuál es la materia con el mejor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el peor desempeño para la materia historia?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | historia | 1 | | literatura | 2 | | biologia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 5.4 1.0 0.0 2.0 9.0 1.0 0.0 3.0 2.6 2.0 0.0 1.0 9.7 2.0 0.0 2.0 2.3 2.0 0.0 3.0 0.3 3.0 0.0 1.0 5.4 3.0 0.0 2.0 1.3 3.0 0.0 3.0 1.3 4.0 1.0 1.0 2.6 4.0 1.0 2.0 5.7 4.0 1.0 3.0 2.5 5.0 1.0 1.0 2.1 5.0 1.0 2.0 5.7 5.0 1.0 3.0 6.3 6.0 1.0 1.0 7.0 6.0 1.0 2.0 2.9 6.0 1.0 3.0 2.3 | | Salida del programa | | 8 10 historia Marcela |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.33**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | María | f | matemáticas | 85.3 | | Daniel | m | matemáticas | 99.9 | | María | f | idiomas | 86.0 | | Daniel | m | idiomas | 78.5 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Qué porcentaje de los exámenes fueron Excelente?  ¿Cuál es la materia con el peor desempeño promedio para todo el grupo?  ¿Cuál es el estudiante con el mejor desempeño para la materia matemáticas?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | matemáticas | 1 | | informática | 2 | | física | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 59.0 1.0 0.0 2.0 17.4 1.0 0.0 3.0 12.7 2.0 0.0 1.0 20.9 2.0 0.0 2.0 98.2 2.0 0.0 3.0 97.9 3.0 0.0 1.0 36.8 3.0 0.0 2.0 38.8 3.0 0.0 3.0 8.9 4.0 1.0 1.0 96.8 4.0 1.0 2.0 5.0 4.0 1.0 3.0 46.3 5.0 1.0 1.0 35.1 5.0 1.0 2.0 40.9 5.0 1.0 3.0 88.6 6.0 1.0 1.0 66.4 6.0 1.0 2.0 25.9 6.0 1.0 3.0 71.5 | | Salida del programa | | 7 0.17 informática María |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.34**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 5, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 4.3 | | Armando | m | matemáticas | 4.8 | | Alexandra | f | idiomas | 3.7 | | Armando | m | idiomas | 2.6 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (4.5 - 5] | Excelente | | (3.5 - 4.5] | Sobresaliente | | (2.5 - 3.5] | Regular | | (1 - 2.5] | Insuficiente | | [0 - 1] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Qué porcentaje de los exámenes fueron Sobresalientes?  ¿Cuál es la materia con el mejor desempeño promedio para el género femenino?  ¿Cuál es el estudiante con el mejor desempeño para la materia informática?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | informática | 1 | | física | 2 | | química | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.2 1.0 0.0 2.0 3.0 1.0 0.0 3.0 3.9 2.0 0.0 1.0 4.9 2.0 0.0 2.0 3.7 2.0 0.0 3.0 2.2 3.0 0.0 1.0 2.5 3.0 0.0 2.0 0.6 3.0 0.0 3.0 5.0 4.0 1.0 1.0 2.9 4.0 1.0 2.0 4.9 4.0 1.0 3.0 3.9 5.0 1.0 1.0 1.1 5.0 1.0 2.0 1.4 5.0 1.0 3.0 4.3 6.0 1.0 1.0 3.0 6.0 1.0 2.0 4.7 6.0 1.0 3.0 2.2 | | Salida del programa | | 8 0.22 física Nicolás |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.35**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 10, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Alexandra | f | matemáticas | 8.3 | | Daniel | m | matemáticas | 9.2 | | Alexandra | f | idiomas | 8.3 | | Daniel | m | idiomas | 6.7 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (9 - 10] | Excelente | | (8 - 9] | Sobresaliente | | (6 - 8] | Regular | | (3 - 6] | Insuficiente | | [0 - 3] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes tienen una calificación mayor a la del promedio del grupo?  ¿Qué porcentaje de los exámenes fueron Regulares?  ¿Cuál es la materia con el peor desempeño promedio para el género masculino?  ¿Cuál es el estudiante con el peor desempeño para la materia literatura?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | literatura | 1 | | biologia | 2 | | Geografía | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 1.5 1.0 0.0 2.0 2.4 1.0 0.0 3.0 6.2 2.0 0.0 1.0 2.4 2.0 0.0 2.0 8.1 2.0 0.0 3.0 8.3 3.0 0.0 1.0 0.0 3.0 0.0 2.0 3.3 3.0 0.0 3.0 2.1 4.0 1.0 1.0 0.6 4.0 1.0 2.0 6.7 4.0 1.0 3.0 7.9 5.0 1.0 1.0 0.9 5.0 1.0 2.0 6.1 5.0 1.0 3.0 8.1 6.0 1.0 1.0 3.3 6.0 1.0 2.0 3.0 6.0 1.0 3.0 7.8 | | Salida del programa | | 8 0.28 literatura Daniel |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.36**

|  |  |
| --- | --- |
| Nombre del reto: | Diseño de sistema estadístico para escuela |
| Autor reto: | Rocio Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Una escuela primaria desea implementar un sistema estadístico de clasificación de notas de los exámenes de sus estudiantes.  La escuela utiliza un sistema de notas basado en números que van desde el 0 hasta el 100, con diferentes rangos con labels que enfatizan el desempeño de los estudiantes.  El sistema debe generar datos estadísticos que puedan ayudar a los profesores a identificar grupos o estudiantes con dificultades. Actualmente se cuenta con la información de las calificaciones de los exámenes para cada uno de los estudiantes de la clase F, el sistema debe generar los datos estadísticos a partir de la siguiente información:   |  |  |  |  | | --- | --- | --- | --- | | **Nombre** | **Género** | **Materia** | **Nota** | | Marcela | f | matemáticas | 85.0 | | Nicolás | m | matemáticas | 93.7 | | Marcela | f | idiomas | 83.3 | | Nicolás | m | idiomas | 72.9 |   Las calificaciones en la escuela se asignan con la siguiente escala de rangos:   |  |  | | --- | --- | | Rango de notas | Calificación | | (90 - 100] | Excelente | | (80 - 90] | Sobresaliente | | (60 - 80] | Regular | | (30 - 60] | Insuficiente | | [0 - 30] | Deficiente |   El algoritmo debe ser capaz de responder a las siguientes preguntas:  ¿Cuántos de los exámenes fueron reprobados?  ¿Cuántos exámenes tienen una calificación Excelente?  ¿Cuál es la materia con el mayor número de exámenes aprobados?  ¿Cuál es el estudiante con el peor desempeño para la materia historia?    **EJEMPLO**  Para facilitar el proceso de ingreso y manipulación de los datos, los valores de tipo string se les asignará un identificador numérico único por categorías:   |  |  | | --- | --- | | Nombre | Identificador | | Armando | 1 | | Nicolás | 2 | | Daniel | 3 | | María | 4 | | Marcela | 5 | | Alexandra | 6 |  |  |  | | --- | --- | | Materia | Identificador | | historia | 1 | | literatura | 2 | | biologia | 3 |  |  |  | | --- | --- | | Género | Identificador | | m | 0 | | f | 1 |  |  | | --- | | Entrada del programa | | 18 1.0 0.0 1.0 84.3 1.0 0.0 2.0 1.6 1.0 0.0 3.0 0.8 2.0 0.0 1.0 92.6 2.0 0.0 2.0 45.1 2.0 0.0 3.0 74.7 3.0 0.0 1.0 28.1 3.0 0.0 2.0 85.0 3.0 0.0 3.0 26.5 4.0 1.0 1.0 27.3 4.0 1.0 2.0 30.1 4.0 1.0 3.0 91.1 5.0 1.0 1.0 36.2 5.0 1.0 2.0 4.7 5.0 1.0 3.0 40.3 6.0 1.0 1.0 77.4 6.0 1.0 2.0 19.6 6.0 1.0 3.0 39.9 | | Salida del programa | | 12 2 biologia María |     Otro aspecto importante es el formato de entrada de los datos, la primera línea de la entrada se trata del número de registros que se deben leer. Las líneas de los registros están tienen un formato de tabla, el orden de las columnas es: nombre, género, materia y calificación, las columnas están separadas por un espacio. Se recomienda copiar y pegar este ejemplo en la terminal para realizar pruebas.  Notas:  Prestar especial cuidado a las notaciones de los rangos.  Se considera aprobado a una calificación igual o mayor a Regular.  Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución ya que estos pueden representar errores en la calificación automática de Codegrade.  Al imprimir números decimales deben ser redondeados a dos cifras significativas, ejemplo: 3.666666667 → 3.67.  Los elementos con un identificador menor tienen prioridad, al ejecutar un proceso y este arroja varios posibles resultados, se debe imprimir el que tenga menor identificador.  El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.37**

|  |  |
| --- | --- |
| Nombre del reto: | Detección de Enfermedades Tempranas |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Un centro de salud desea iniciar una investigación relacionada con la detección temprana de enfermedades asociadas a funciones renales utilizando como indicador el nivel de creatinina en el organismo.  La creatinina se puede medir a través de muestras de laboratorio tanto en sangre como en orina, y sus valores normales son los siguientes:   |  |  |  | | --- | --- | --- | |  | Masculino | Femenino | | Rango normal de creatinina en sangre. Miligramos por decilitro de sangre (mg/dL) | de 0,74 a 1,35 mg/dL | de 0,59 a 1,04 mg/dL | | Rango normal de creatinina en orina. Miligramos por kilogramo de masa corporal por día (mg/kg/día). | De 14 a 26 mg/kg/día | De 11 a 20 mg/kg/día |   El diagnóstico preliminar de una enfermedad crónica renal (ECR) inicia con la toma de 3 muestras del nivel de creatinina de un paciente, las primeras dos muestras corresponden al nivel de creatinina en sangre, y la última al nivel de creatinina en la orina. En cada prueba realizada se le otorga un puntaje de 5 al paciente si el resultado es anormal, y de 0 si no lo es. Al final de las 3 pruebas de acuerdo al puntaje total obtenido se procede con la categorización del riesgo del paciente a padecer una ECR de acuerdo a la siguiente tabla:   |  |  | | --- | --- | | Puntaje | Riesgo | | 0 | Sin riesgo | | 5 | Bajo | | 10 | Medio | | 15 | Alto |   Como apoyo a la fase inicial de la investigación se le ha contratado a usted para realizar un programa que deberá entonces:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, y género (M para masculino y F para femenino) de cada uno de los pacientes. * Leer y almacenar el valor de las muestras de laboratorio de cada uno de los pacientes.   Una vez leídos los datos:   * Por paciente indicar el puntaje total obtenido al evaluar las 3 muestras. * Por paciente indicar la categorización del riesgo. * Por paciente ¿cuántas muestras se encuentran por debajo de los niveles normales? * Por paciente ¿cuáles son los números de muestras (separadas por espacio) que se encuentran por debajo de los niveles normales? * ¿Cuál es el nombre del paciente con el valor más alto en el nivel de creatinina en la orina?   Ejemplo:  En la siguiente tabla se muestran los datos para dos pacientes.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Nombre | Cedula | Género | Muestra 1 | Muestra 2 | Muestra 3 | | Pedro González | 1098462547 | M | 0.7 | 0.6 | 13 | | Andrea Pérez | 72645468 | F | 0.4 | 1 | 10 |   Para el paciente 1:  Puntaje obtenido: 15  Categorización riesgo: Alto  Muestras por debajo de los niveles normales: 3  Número de las muestras que se encuentran por debajo de los niveles normales:1 2 3  Para el paciente 2:  Puntaje obtenido: 10  Categorización riesgo: Medio  Muestras por debajo de los niveles normales: 2  Número de las muestras por debajo de los niveles normales: 1 3  ¿Cuál es el nombre del paciente con el valor más alto en el nivel de creatinina en la orina?: Pedro González | |

**RETO 1.38**

|  |  |
| --- | --- |
| Nombre del reto: | Detección de Enfermedades Tempranas |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Un centro de salud desea iniciar una investigación relacionada con la detección temprana de enfermedades asociadas a funciones renales utilizando como indicador el nivel de creatinina en el organismo.  La creatinina se puede medir a través de muestras de laboratorio tanto en sangre como en orina, y sus valores normales son los siguientes:   |  |  |  | | --- | --- | --- | |  | Masculino | Femenino | | Rango normal de creatinina en sangre. Miligramos por decilitro de sangre (mg/dL) | de 0,74 a 1,35 mg/dL | de 0,59 a 1,04 mg/dL | | Rango normal de creatinina en orina. Miligramos por kilogramo de masa corporal por día (mg/kg/día). | De 14 a 26 mg/kg/día | De 11 a 20 mg/kg/día |   El diagnóstico preliminar de una enfermedad crónica renal (ECR) inicia con la toma de 3 muestras del nivel de creatinina de un paciente, las primeras dos muestras corresponden al nivel de creatinina en sangre, y la última al nivel de creatinina en la orina. En cada prueba realizada se le otorga un puntaje de 10 al paciente si el resultado es anormal, y de 0 si no lo es. Al final de las 3 pruebas de acuerdo al puntaje total obtenido se procede con la categorización del riesgo del paciente a padecer una ECR de acuerdo a la siguiente tabla:   |  |  | | --- | --- | | Puntaje | Riesgo | | 0 | Sin riesgo | | 10 | Bajo | | 20 | Medio | | 30 | Alto |   Como apoyo a la fase inicial de la investigación se le ha contratado a usted para realizar un programa que:  El programa deberá entonces:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula y género (M para masculino y F para femenino) de cada uno de los pacientes. * Leer y almacenar el valor de las muestras de laboratorio de cada uno de los pacientes.   Una vez leídos los datos:   * Por paciente indicar el puntaje total obtenido al evaluar las 3 muestras. * Por paciente indicar la categorización del riesgo. * Por paciente indicar el número de la muestra más alta. * ¿Cuál es el nombre del paciente con el valor más alto en cada una de las muestras?   Ejemplo:  En la siguiente tabla se muestran los datos para dos pacientes.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Nombre | Cedula | Género | Muestra 1 | Muestra 2 | Muestra 3 | | Julián Arévalo | 1098462547 | M | 0.8 | 0.7 | 15 | | María Córdoba | 72645468 | F | 0.5 | 1.3 | 11 |   Para el paciente 1  Puntaje obtenido: 10  Categorización riesgo: Bajo  El número de la muestra más alta es el 3  Para el paciente 2  Puntaje obtenido: 20  Categorización riesgo: Medio  El número de la muestra más alta es el 3  El paciente que tiene la muestra 1 más alta es: Julián Arévalo  El paciente que tiene la muestra 2 más alta es: María Córdoba  El paciente que tiene la muestra 3 más alta es: Julián Arévalo | |

**RETO 1.39**

|  |  |
| --- | --- |
| Nombre del reto: | Detección de Enfermedades Tempranas |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Un centro de salud desea iniciar una investigación relacionada con la detección temprana de enfermedades asociadas a funciones renales utilizando como indicador el nivel de creatinina en el organismo.  La creatinina se puede medir a través de muestras de laboratorio tanto en sangre como en orina, y sus valores normales son los siguientes:   |  |  |  | | --- | --- | --- | |  | Hombres | Mujeres | | Rango normal de creatinina en sangre. Miligramos por decilitro de sangre (mg/dL) | de 0,74 a 1,35 mg/dL | de 0,59 a 1,04 mg/dL | | Rango normal de creatinina en orina. Miligramos por kilogramo de masa corporal por día (mg/kg/día). | De 14 a 26 mg/kg/día | De 11 a 20 mg/kg/día |   El diagnóstico preliminar de una enfermedad crónica renal (ECR) inicia con la toma de 3 muestras del nivel de creatinina de un paciente, las primeras dos muestras corresponden al nivel de creatinina en sangre, y la última al nivel de creatinina en la orina. En cada prueba realizada se le otorga un puntaje de 10 al paciente si el resultado es anormal, y de 0 si no lo es. Al final de las 4 pruebas de acuerdo al puntaje total obtenido se procede con la categorización del riesgo del paciente a padecer una ECR de acuerdo a la siguiente tabla:   |  |  | | --- | --- | | Puntaje | Riesgo | | 0 | Sin riesgo | | 10 | Bajo | | 20-30 | Medio | | 40 | Alto |   Como apoyo a la fase inicial de la investigación se le ha contratado a usted para realizar un programa que:  El programa deberá entonces:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, y género (M para masculino y F para femenino) de cada uno de los pacientes. * Leer y almacenar el valor de las muestras de laboratorio de cada uno de los pacientes.   Una vez leídos los datos:   * Por paciente indicar el puntaje total obtenido al evaluar las 4 muestras. * Por paciente indicar la categorización del riesgo. * Por paciente indicar el promedio de las muestras de la creatinina en sangre y en orina. * ¿Cuál es el número de la cédula del paciente con la primera muestra de creatinina más baja?   Ejemplo:  En la siguiente tabla se muestran los datos para dos pacientes.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Nombre | Cedula | Género | Muestra 1 | Muestra 2 | Muestra 3 | Muestra 4 | | Pablo Nieto | 1098462547 | M | 0.8 | 0.7 | 15 | 12 | | María Córdoba | 72645468 | F | 0.5 | 1.3 | 11 | 13 |   Para el paciente 1  Puntaje obtenido: 20  Categorización riesgo: Medio  Promedio en sangre: 0.75  Promedio en orina: 13.5  Para el paciente 2  Puntaje obtenido: 20  Categorización riesgo: Medio  Promedio en sangre: 0.9  Promedio en orina: 12.0  La cédula del paciente con la primera muestra de creatinina más baja es: 72645468 | |

**RETO 1.40**

|  |  |
| --- | --- |
| Nombre del reto: | Detección de Enfermedades De Transmisión Alimentaria |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Un centro de salud desea iniciar una investigación relacionada con las enfermedades de transmisión alimentaria.  En el proceso de diagnóstico el profesional de la salud realiza la valoración inicial del paciente en la cual deberá identificar si se padece o no de alguno o algunos de los siguientes síntomas: Náuseas, vómitos, dolor abdominal, diarrea y fiebre.  Una vez valorados los síntomas se verifica con la siguiente tabla cuál es la posible enfermedad de transmisión alimentaria que pudiera tener el paciente:   |  |  |  | | --- | --- | --- | | **Enfermedad** | **Clasificación** | **Síntomas** | | Staphylococcus aureus | Bacteriano | Náuseas, vómitos, dolor abdominal, fiebre, diarrea | | Bacillus cereus | Bacteriano | Náuseas, vómito | | Taenia saginata | Parasitaria | Dolor abdominal, fiebre | | Norovirus | Viral | Náuseas, vómitos, diarrea, fiebre | | Rotavirus | Viral | Vómitos, diarrea |   Para que el paciente sea diagnosticado de manera efectiva deberá presentar todos los síntomas de la enfermedad, ni más ni menos. En caso tal que el paciente no pueda ser diagnosticado con éxito, será valorado con exámenes adicionales que apoyen el proceso de diagnóstico.  Se han tomado los datos de un conjunto de pacientes con el fin de generar estadísticas que den paso a la investigación requerida.  Conforme a lo anterior, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, y síntomas padecidos de cada uno de los pacientes.   Una vez leídos los datos   * Calcular el diagnóstico de cada paciente., indicando número de cédula (separados por espacio). * ¿Cuál fue el diagnóstico que más se presentó?, si varios diagnósticos cumplen esta condición tomar el primero. * ¿Cuántos pacientes no pudieron ser diagnosticados?   Requerimiento: Los síntomas deben ser leídos en el siguiente orden:  Náuseas, vómitos, dolor abdominal, diarrea y fiebre.  Ejemplo  Los siguientes tres pacientes presentaron los síntomas detallados:  Pablo Nieto: dolor abdominal, diarrea, náuseas, vómitos, fiebre  María Córdoba: dolor abdominal, fiebre.  José Pérez: fiebre  El diagnostico de Pablo Nieto fue Salmonella sp, y el de María Córdoba fue Taenia saginata.  Diagnostico que más se repitió: Salmonella sp  Pacientes sin diagnóstico: 1 | |

**RETO 1.41**

|  |  |
| --- | --- |
| Nombre del reto: | Detección de Enfermedades De Transmisión Alimentaria |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Un centro de salud desea iniciar una investigación relacionada con las enfermedades de transmisión alimentaria.  En el proceso de diagnóstico el profesional de la salud realiza la valoración inicial del paciente en la cual deberá identificar si se padece o no de alguno o algunos de los siguientes síntomas: Náuseas, vómitos, dolor abdominal, diarrea y fiebre.  Una vez valorados los síntomas se verifica con la siguiente tabla cuál es la posible enfermedad de transmisión alimentaria que pudiera tener el paciente:   |  |  |  | | --- | --- | --- | | **Enfermedad** | **Clasificación** | **Síntomas** | | Staphylococcus aureus | Bacteriano | Náuseas, vómitos, dolor abdominal, fiebre, diarrea | | Bacillus cereus | Bacteriano | Náuseas, vómito | | Taenia saginata | Parasitaria | Dolor abdominal, fiebre | | Norovirus | Viral | Náuseas, vómitos, diarrea, fiebre | | Rotavirus | Viral | Vómitos, diarrea |   Para que el paciente sea diagnosticado de manera efectiva deberá presentar todos los síntomas de la enfermedad, ni más ni menos. En caso tal que el paciente no pueda ser diagnosticado con éxito, será valorado con exámenes adicionales que apoyen el proceso de diagnóstico.  Se han tomado los datos de un conjunto de pacientes con el fin de generar estadísticas que den paso a la investigación requerida.  Conforme a lo anterior, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, y síntomas padecidos de cada uno de los pacientes.   Una vez leídos los datos   * Calcular el diagnóstico de cada paciente., indicando número de cédula (separados por espacio). * ¿Cuál fue el síntoma que más se presentó?, si varios diagnósticos cumplen esta condición tomar el primero. * ¿Cuál fue el síntoma que menos se presentó?, si varios diagnósticos cumplen esta condición tomar el primero. * ¿Cuántos pacientes fueron diagnosticados con éxito?   Requerimiento: Los síntomas deben ser leídos en el siguiente orden:  Náuseas, vómitos, dolor abdominal, diarrea y fiebre.  Ejemplo  Los siguientes tres pacientes presentaron los síntomas detallados:  Pablo Nieto: dolor abdominal, diarrea, náuseas, vómitos, fiebre  María Córdoba: dolor abdominal  José Pérez: fiebre  El diagnóstico de Pablo Nieto fue Salmonella sp. María Córdoba fue Taenia saginata y José Pérez no tuvieron diagnóstico.  Síntoma que más se repitió: fiebre  Síntoma que menos se repitió: náuseas  Pacientes diagnosticados con éxito: 1 | |

**RETO 1.42**

|  |  |
| --- | --- |
| Nombre del reto: | Detección de Enfermedades De Transmisión Alimentaria |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| Un centro de salud desea iniciar una investigación relacionada con las enfermedades de transmisión alimentaria.  En el proceso de diagnóstico el profesional de la salud realiza la valoración inicial del paciente en la cual deberá identificar si se padece o no de alguno o algunos de los siguientes síntomas Náuseas, vómitos, dolor abdominal, diarrea y fiebre.  Una vez valorados los síntomas se verifica con la siguiente tabla cuál es la posible enfermedad de transmisión alimentaria que pudiera tener el paciente:   |  |  |  | | --- | --- | --- | | **Enfermedad** | **Clasificación** | **Síntomas** | | Staphylococcus aureus | Bacteriano | Náuseas, vómitos, dolor abdominal, fiebre, diarrea | | Bacillus cereus | Bacteriano | Náuseas, vómito | | Taenia saginata | Parasitaria | Dolor abdominal, fiebre | | Norovirus | Viral | Náuseas, vómitos, diarrea, fiebre | | Rotavirus | Viral | Vómitos, diarrea |   Para que el paciente sea diagnosticado de manera efectiva deberá presentar todos los síntomas de la enfermedad, ni más ni menos. En caso tal que el paciente no pueda ser diagnosticado con éxito, será valorado con exámenes adicionales que apoyen el proceso de diagnóstico.  Se han tomado los datos de un conjunto de pacientes con el fin de generar estadísticas que den paso a la investigación requerida.  Conforme a lo anterior, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, y síntomas padecidos de cada uno de los pacientes.   Una vez leídos los datos   * Calcular el diagnóstico de cada paciente, indicando número de cédula (separados por espacio). * De la clasificación bacteriana ¿Cuál fue el diagnóstico que más se presentó?, si varios diagnósticos cumplen esta condición tomar el primero. * De la clasificación viral ¿Cuál fue el diagnóstico que más se presentó?, si varios diagnósticos cumplen esta condición tomar el primero.   Requerimiento: Los síntomas deben ser leídos en el siguiente orden:  Náuseas, vómitos, dolor abdominal, diarrea y fiebre.  Ejemplo  Los siguientes tres pacientes presentaron los síntomas detallados:  Pablo Nieto: dolor abdominal, diarrea, náuseas, vómitos, fiebre  María Córdoba: dolor abdominal.  José Pérez: fiebre  El diagnóstico de Pablo Nieto fue Salmonella sp. María Córdoba fue Taenia saginata y José Pérez no tuvieron diagnóstico.  De la clasificación bacteriana el diagnostico que más se presento fue: Salmonella sp  De la clasificación viral el diagnostico que más se presento fue: Norovirus | |

**RETO 1.43**

|  |  |
| --- | --- |
| Nombre del reto: | Estudio De Prevención De Mortalidad |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En Colombia se desea iniciar una investigación relacionada con las causas de mortalidad en la población.  En estudios previos se ha detectado que las siguientes enfermedades causan gran parte de la mortalidad en los pacientes: cáncer, cardiovasculares, respiratorias, cerebrovasculares, hipertensión y diabetes.  El objetivo de la investigación es realizar un estudio estadístico de las enfermedades anteriormente mencionadas con el fin de mitigar la mortalidad en la población y tomar decisiones en cuanto a los recursos disponibles para atacar cada una de ellas.  Para lo anterior, se tienen los siguientes datos de un conjunto de pacientes:   * Nombre completo * Número de cédula * Edad * Ciudad * EPS * Enfermedad diagnosticada   Conforme a lo mencionado, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, edad, ciudad, EPS, y enfermedad diagnosticada de cada uno de los pacientes.   Una vez leídos los datos   * ¿Cuál es la edad promedio de los pacientes del estudio? * ¿Cuáles pacientes se encuentran por encima de la edad promedio?, indicar nombre y cédula separados por espacios.   Ejemplo:  Se tienen los datos de los siguientes pacientes:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Julián | 723456 | 45 | Barranquilla | Sura | cardiovasculares | | Andrés | 734673 | 76 | Barranquilla | Sanitas | Cáncer |   Edad promedio: 60,5  Pacientes por encima del promedio:  Andrés 734673 | |

**RETO 1.44**

|  |  |
| --- | --- |
| Nombre del reto: | Estudio De Prevención De Mortalidad |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En Colombia se desea iniciar una investigación relacionada con las causas de mortalidad en la población.  En estudios previos se ha detectado que las siguientes enfermedades causan gran parte de la mortalidad en los pacientes: cáncer, cardiovasculares, respiratorias, cerebrovasculares, hipertensión y diabetes.  El objetivo de la investigación es realizar un estudio estadístico de las enfermedades anteriormente mencionadas con el fin de mitigar la mortalidad en la población y tomar decisiones en cuanto a los recursos disponibles para atacar cada una de ellas.  Para lo anterior, se tienen los siguientes datos de un conjunto de pacientes:   * Nombre completo * Número de cédula * Edad * Ciudad * EPS * Enfermedad diagnosticada   Conforme a lo mencionado, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer cantidad y nombre de las diferentes EPS en el estudio. * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, edad, ciudad, EPS, y enfermedad diagnosticada de cada uno de los pacientes.   Una vez leídos los datos:   * ¿Cuál es la enfermedad que más se presenta?, si varios diagnósticos cumplen esta condición tomar el primero. * ¿Cuál es la enfermedad que menos se presenta?, si varios diagnósticos cumplen esta condición tomar el primero. * ¿Cuál es la EPS que registra mayor número de pacientes en el estudio?, si varios diagnósticos cumplen esta condición tomar el primero.   Ejemplo:  Se tienen los datos de los siguientes pacientes:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Julián | 723456 | 45 | Barranquilla | Sura | cardiovasculares | | Andrés | 734673 | 76 | Barranquilla | Sanitas | Cáncer | | Eva | 104312456 | 32 | Bogotá | Sura | Cáncer |   Enfermedad que más se presenta: Cáncer  Enfermedad que menos se presenta: cardiovasculares  EPS con mayor registro: Sura | |

**RETO 1.45**

|  |  |
| --- | --- |
| Nombre del reto: | Estudio De Prevención De Mortalidad |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En Colombia se desea iniciar una investigación relacionada con las causas de mortalidad en la población.  En estudios previos se ha detectado que las siguientes enfermedades causan gran parte de la mortalidad en los pacientes: cáncer, cardiovasculares, respiratorias, cerebrovasculares, hipertensión y diabetes.  El objetivo de la investigación es realizar un estudio estadístico de las enfermedades anteriormente mencionadas con el fin de mitigar la mortalidad en la población y tomar decisiones en cuanto a los recursos disponibles para atacar cada una de ellas.  Para lo anterior, se tienen los siguientes datos de un conjunto de pacientes:   * Nombre completo * Número de cédula * Edad * Ciudad * EPS * Enfermedad diagnosticada   Conforme a lo mencionado, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, edad, ciudad, EPS, y enfermedad diagnosticada de cada uno de los pacientes. * ¿Cuál es la ciudad en la cual se encuentra el mayor número de pacientes? * En la ciudad anterior ¿Cuál es la enfermedad que más se presenta y la que menos se presenta? | |

**RETO 1.46**

|  |  |
| --- | --- |
| Nombre del reto: | Estudio De Prevención De Mortalidad |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En Colombia se desea iniciar una investigación relacionada con las causas de mortalidad en la población.  En estudios previos se ha detectado que las siguientes enfermedades causan gran parte de la mortalidad en los pacientes: cáncer, cardiovasculares, respiratorias, cerebrovasculares, hipertensión y diabetes.  El objetivo de la investigación es realizar un estudio estadístico de las enfermedades anteriormente mencionadas con el fin de mitigar la mortalidad en la población y tomar decisiones en cuanto a los recursos disponibles para atacar cada una de ellas.  Para lo anterior, se tienen los siguientes datos de un conjunto de pacientes:   * Nombre completo * Número de cédula * Edad * Ciudad * EPS * Enfermedad diagnosticada   Conforme a lo mencionado, se le ha contratado a usted para apoyar el proceso de investigación desarrollando un programa que implemente los siguientes requerimientos:   * Leer la cantidad de pacientes del estudio. * Leer y almacenar el nombre, número de cédula, edad, ciudad, EPS, y enfermedad diagnosticada de cada uno de los pacientes.   Una vez leídos los datos:   * ¿Cuántos pacientes hay en cada ciudad? * ¿Cuál es la ciudad en la cual se encuentra el menor número de pacientes?   Ejemplo:  Se tienen los datos de los siguientes pacientes:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Julián | 723456 | 45 | Barranquilla | Sura | cardiovasculares | | Andrés | 734673 | 76 | Barranquilla | Sanitas | Cáncer | | Eva | 104312456 | 32 | Bogotá | Sura | Cáncer |   Pacientes en Barranquilla: 2  Pacientes en Bogotá: 1  Menor número de pacientes en: Bogotá | |

**RETO 1.47**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el nivel de riesgo de la calidad del agua más alto encontrado? * ¿Cuál es el nivel de riesgo de la calidad del agua más bajo encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplo:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [7.9 5 30 70 25 4] | MEDIO  ALTO  Continuar el control y la vigilancia |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.48**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el segundo nivel de riesgo de la calidad del agua más alto encontrado? * ¿Cuál es el nivel de riesgo de la calidad del agua más común encontrado? En caso de haber dos o más niveles igual de comunes tomar el más bajo.   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplo:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 7 35 70 50 8] | MEDIO  ALTO  BAJO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.49**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la clasificación IRCA más alta encontrada? * ¿Cuál es el nivel de riesgo de la calidad del agua más bajo encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 7 25 70 50 8] | MEDIO  70.00  BAJO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.50**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la clasificación IRCA más baja encontrada? * ¿Cuál es el nivel de riesgo de la calidad del agua más alta encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [7.9 12 30 70 25 14] | MEDIO  7.90  ALTO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.51**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuáles son las entidades que tendrían que tomar acciones si se decidiese solo tener en cuenta el promedio de la clasificación IRCA en los cuerpos de agua? * ¿Cuál es el nivel de riesgo de la calidad del agua más alto encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 10 25 70 50 27] | MEDIO  PERSONA PRESTADORA  ALTO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**RETO 1.52**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la clasificación IRCA promedio? * ¿Cuál es el nivel de riesgo de la calidad del agua más común encontrado? En caso de haber dos o más niveles igual de comunes tomar el más bajo.   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 36 25 70 55 18] | MEDIO  34.85  ALTO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.53**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el porcentaje de cuerpos que tienen el mismo nivel de riesgo que el promedio? * ¿Cuál es el nivel de riesgo de la calidad del agua más bajo encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 7 25 70 50 8] | MEDIO  16.67  BAJO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.54**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el porcentaje de cuerpos de agua que tienen un nivel de riesgo considerado INVIABLE SANITARIAMENTE? * ¿Cuál es el nivel de riesgo de la calidad del agua más alto encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 95 25 70 100 8] | ALTO  33.33  INVIABLE SANITARIAMENTE |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.55**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el porcentaje de cuerpos de agua que tienen un nivel de riesgo considerado ALTO? * ¿Cuál es el nivel de riesgo de la calidad del agua más común?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [7.9 12 30 70 25 14] | MEDIO  16.67  BAJO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.56**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el porcentaje de cuerpos de agua que tienen un nivel de riesgo considerado MEDIO? * ¿Cuál es el porcentaje de cuerpos de agua que tienen un nivel de riesgo considerado ALTO?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 7 25 70 50 8] | MEDIO  16.67  33.33 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.57**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es el porcentaje de cuerpos de agua que tienen un nivel de riesgo considerado BAJA? * ¿Cuál es el porcentaje de cuerpos de agua que tienen un nivel de riesgo considerado INVIABLE SANITARIAMENTE?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 7 25 70 50 90] | ALTO  33.33  16.67 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.58**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de los porcentajes de cuerpos de agua que tienen niveles de riesgo considerados INVIABLE SANITARIAMENTE y ALTO? * ¿Existe algún cuerpo de agua en el que solo sea necesario continuar la vigilancia?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7.5 25 70 77 81] | ALTO  50.00  SI |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.59**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de los porcentajes de cuerpos de agua que tienen niveles de riesgo considerados ALTO y MEDIO? * Verdadero o falso ¿Hay cuerpos de agua que no se encuentran en riesgo?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [3 7 23 60 50 45] | MEDIO  66.67  VERDADERO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.60**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de los porcentajes de cuerpos de agua que tienen niveles de riesgo considerados MEDIO y BAJO? * ¿Hay más de un cuerpo de agua cuyo nivel de riesgo sea INVIABLE SANITARIAMENTE?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5.1 7 25 70 50 8] | MEDIO  66.67  NO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.61**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Persona prestadora,  alcaldía, gobernación,  entidades de orden  territorial | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Persona prestadora,  alcaldía, gobernación | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de los porcentajes de cuerpos de agua que tienen niveles de riesgo considerados BAJO y SIN RIESGO? * ¿Existe algún cuerpo de agua en el que solo sea necesario continuar la vigilancia?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  50.00  SI |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.62**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuál es la entidad a la que más se recurre según los datos leídos? En caso de haber dos o más entidades a las que se recurra igual cantidad de veces tomar la que corresponda al nivel de riesgo más bajo. * ¿Fue necesario recurrir a la gobernación por lo menos una vez?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 90 3] | MEDIO  CONTINUAR VIGILANCIA  NO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.63**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuál es la entidad a la que más se recurre según los datos leídos? En caso de haber dos o más entidades a las que se recurra igual cantidad de veces tomar la que corresponda al nivel de riesgo más bajo. * ¿Se recurrió a “CONTINUAR VIGILANCIA”?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 80 90] | ALTO  ALCALDIA  SI |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.64**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuál es la entidad a la que más se recurre según los datos leídos? * ¿Fue necesario recurrir a la ALCALDIA más de una vez?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  PERSONA PRESTADORA  SI |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.65**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuántas veces fue necesario recurrir a “CONTINUAR VIGILANCIA”? * ¿Cuál fue la clasificación IRCA más alta encontrada?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  1.00  70.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.66**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuántas veces fue necesario recurrir a “PERSONA PRESTADORA”? * ¿Cuál fue la clasificación IRCA más baja encontrada?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  3.00  5.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.67**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuántas veces fue necesario recurrir a la ALCALDIA? * ¿Cuál fue el nivel de riesgo más alto encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  2.00  ALTO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.68**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | GOBERNACION | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | ALCALDIA | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | PERSONA PRESTADORA | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | CONTINUAR VIGILANCIA |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * En cuanto a las entidades a tomar acciones ¿Cuántas veces fue necesario recurrir a la GOBERNACION? * ¿Cuál fue el nivel de riesgo más bajo encontrado?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  0.00  Continuar el control y la vigilancia |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.69**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál fue la clasificación IRCA más baja encontrada que calificara como nivel de riesgo INVIABLE SANITARIAMENTE? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada que calificara como nivel de riesgo INVIABLE SANITARIAMENTE? En caso de no haber ninguna regresar “NA”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 85 70 97 8] | ALTO  85.00  97.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.70**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál fue la clasificación IRCA más baja encontrada que calificara como nivel de riesgo ALTO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada que calificara como nivel de riesgo ALTO? En caso de no haber ninguna regresar “NA”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  50.00  70.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.71**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál fue la clasificación IRCA más baja encontrada que calificara como nivel de riesgo MEDIO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada que calificara como nivel de riesgo MEDIO? En caso de no haber ninguna regresar “NA”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  25.00  25.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.72**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál fue la clasificación IRCA más baja encontrada que calificara como nivel de riesgo BAJO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada que calificara como nivel de riesgo BAJO? En caso de no haber ninguna regresar “NA”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  7.00  8.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.73**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál fue la clasificación IRCA más baja encontrada que calificara como nivel de riesgo SIN RIESGO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada que calificara como nivel de riesgo SIN RIESGO? En caso de no haber ninguna regresar “NA”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  5.00  5.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.74**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de todas las clasificaciones IRCA según los datos ingresados? * Si se suman todas las clasificaciones IRCA ¿Esa suma alcanza a llegar a 100, es decir, se pasa de INVIABLE SANITARIAMENTE?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  165.00  SI |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.75**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la resta de todas las clasificaciones IRCA según el orden de los datos ingresados? * Si se restan todas las clasificaciones IRCA ¿Esa resta alcanza a llegar a 0 o menos?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  -165.00  SI |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.76**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la multiplicación de todas las clasificaciones IRCA según los datos ingresados? * Si se multiplican todas las clasificaciones IRCA ¿Esa multiplicación alcanza a llegar a 50000000?   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  24500000.00  NO |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.77**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de la clasificación IRCA más baja encontrada que calificara como nivel de riesgo SIN RIESGO y la clasificación IRCA más alta encontrada que calificara como nivel de riesgo SIN RIESGO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada?”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  10.00  70.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.78**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de la clasificación IRCA más baja encontrada que calificara como nivel de riesgo BAJO y la clasificación IRCA más alta encontrada que calificara como nivel de riesgo BAJO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más baja encontrada?”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  15.00  5.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.79**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de la clasificación IRCA más baja encontrada que calificara como nivel de riesgo MEDIO y la clasificación IRCA más alta encontrada que calificara como nivel de riesgo MEDIO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada?”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  50.00  90.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.80**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de la clasificación IRCA más baja encontrada que calificara como nivel de riesgo ALTO y la clasificación IRCA más alta encontrada que calificara como nivel de riesgo ALTO? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más baja encontrada?”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 70 50 8] | MEDIO  100.00  5.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**RETO 1.81**

|  |  |
| --- | --- |
| Nombre del reto: | Lector del nivel de la calidad del agua en el Atlántico |
| Autora reto: | Rocío Ramos Rodríguez |
| Descripción de reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.  El departamento del Atlántico se ha comprometido con esta causa y por ello ha decidido adoptar estos retos, se lista uno de los principales relacionados con el agua potable:  De aquí a 2030, se busca lograr el acceso universal y equitativo al agua potable a un precio asequible para todos.  Algunas ONG’s se atribuyeron la tarea de poder diseñar un dispositivo para analizar la calidad  del agua de poblaciones apartadas. Para comenzar, requieren que el dispositivo cuente con un  lector de la calidad del agua. Después de la lectura, el dispositivo nos entrega el índice de riesgo  de la calidad del agua, IRCA, y según este resultado debe indicar el nivel de riesgo.   |  |  |  |  | | --- | --- | --- | --- | | **Clasificación IRCA (%)** | **Nivel de riesgo** | **Entidades a notificar** | **Entidades a tomar**  **acciones** | | 80.1 - 100 | INVIABLE  SANITARIAMENTE | Persona prestadora, COVE, Alcaldía, Gobernación, SSPD, MPS, INS, MAVDT, Contraloría General, Procuraduría General | Gobernación (Nivel 4) | | 35.1 - 80 | ALTO | Persona prestadora, COVE,  Alcaldía, Gobernación, SSPD | Alcaldía (Nivel 3) | | 14.1 - 35 | MEDIO | Persona prestadora, COVE, Alcaldía, Gobernación | Persona prestadora (Nivel 2) | | 5.1 - 14 | BAJO | Persona prestadora, COVE | | 0 - 5 | SIN RIESGO | Continuar el control y la  vigilancia | Continuar vigilancia (Nivel 1) |   Se requiere leer un vector de tamaño n de entrada en el que cada elemento indique el porcentaje de la clasificación IRCA.  **El algoritmo debe ser capaz de responder a las siguientes preguntas:**   * ¿Cuál es en promedio el nivel de riesgo de la calidad del agua según los datos ingresados? * ¿Cuál es la suma de la clasificación IRCA más baja encontrada que calificara como nivel de riesgo INVIABLE SANITARIAMENTE y la clasificación IRCA más alta encontrada que calificara como nivel de riesgo INVIABLE SANITARIAMENTE? En caso de no haber ninguna regresar “NA” * ¿Cuál fue la clasificación IRCA más alta encontrada?”   Para el caso donde el porcentaje del IRCA tenga una lectura entre 0 - 5, el programa debe devolver el siguiente mensaje: “Continuar el control y la vigilancia”.  Ejemplos:   |  |  | | --- | --- | | **Entrada esperada** | **Salida esperada** | | [5 7 25 90 81 8] | ALTO  171.00  90.00 |   **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas  y salidas del programa. Por favor NO use ningún signo dentro del desarrollo de su solución ya  que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** El archivo debe llamarse reto1.java, de lo contrario no podrá ser cargado en la plataforma de Codegrade.  **Nota:** Los resultados numéricos deben estar formateados a dos cifras decimales. | |

**Reto 1.82**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir cada elemento del vector de alertas en una línea nueva * Imprimir cada elemento del vector de tratamientos en una línea nueva   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | moderadamente sospechoso  levemente sospechoso  benigno  aaf  seguimiento  no aaf | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | levemente sospechoso  altamente sospechoso  seguimiento  aaf |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.83**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir el conteo de las alertas presentes en el vector. Primero imprimir la alerta y separado por un espacio el conteo. El orden de impresión debe ser: benigno, no sospechoso, levemente sospechoso, moderadamente sospechoso, altamente sospechoso * Imprimir el conteo de los tratamientos presentes en el vector. Primero imprimir el tratamiento y separado por un espacio el conteo. El orden de impresión debe ser: no aaf, seguimiento y aaf   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | benigno 1  levemente sospechoso 1  moderadamente sospechoso 1  no aaf 1  seguimiento 1  aaf 1 | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | levemente sospechoso 1  altamente sospechoso 1  seguimiento 1  aaf 1 |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.84**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir el promedio del tamaño de los nódulos con formato de strings a dos cifras decimales * Imprimir el mínimo de los tamaños de los nódulos   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | 1.67  1.0 | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | 1.50  1.0 |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.85**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir las desviaciones estándar muestral del tamaño de los nódulos con formato de string a dos cifras decimales * Imprimir el máximo de los tamaños de los nódulos   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | 0.58  2.0 | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | 0.71  2.0 |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.86**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir la varianza del tamaño de los nódulos con formato de string a dos cifras decimales * Imprimir la suma del tamaño de los nódulos   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | 0.33  5.0 | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | 0.50  3.0 |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.87**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir las modas de las alertas:   + Si es una sola moda: imprimirla   + Si son dos, tres o cuatro modas: imprimir las alertas en el siguiente orden en una línea nueva: benigno, no sospechoso, levemente sospechoso, moderadamente sospechoso, altamente sospechoso   + Si todas las alertas tienen la misma frecuencia, imprimir: no hay moda * Imprimir la moda de los tratamientos:   + Si es una sola moda: imprimirla   + Si son dos: imprimir los tratamientos en el siguiente orden en una línea nueva: no aaf, seguimiento, aaf   + Si todos los tratamientos tienen la misma frecuencia, imprimir: no hay moda   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | no hay moda  no hay moda | | **Entrada esperada** | | 2  C4  E1  F1  M1  1  1  1  1  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | altamente sospechoso  aaf |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.88**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir la alerta menos frecuente:   + Si es un solo dato: imprimirlo   + Si son dos, tres o cuatro: imprimir las alertas en el siguiente orden: benigno, no sospechoso, levemente sospechoso, moderadamente sospechoso, altamente sospechoso   + Si todas las alertas tienen la misma frecuencia, imprimir: todos los datos tienen la misma frecuencia * Imprimir el tratamiento menos frecuente:   + Si es un solo dato: imprimirlo   + Si son dos: imprimir los tratamientos en el siguiente orden: no aaf, seguimiento, aaf   + Si todos los tratamientos tienen la misma frecuencia, imprimir: todos los datos tienen la misma frecuencia   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | todos los datos tienen la misma frecuencia  todos los datos tienen la misma frecuencia | | **Entrada esperada** | | 4  C3  E4  F1  M3  0  1  1  0  1.63  C4  E2  F1  M1  0  0  0  1  0.54  C4  E3  F1  M3  1  1  0  0  2.27  C2  E3  F1  M3  0  1  0  1  2.73 | | **Salida esperada** | | moderadamente sospechoso  seguimiento |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.89**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Imprimir el rango del tamaño de los nódulos * Imprimir la mediana del tamaño de los nódulos   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | 1.0  2.0 | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | 1.0  **1.5** |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.90**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Imprimir las alertas calculadas, en el orden en el que aparecen en la tabla, en una línea nueva y separada por un espacio la proporción con formato de strings a dos cifras decimales * Imprimir los tratamientos calculados, en el orden en el que aparecen en la tabla, en una línea nueva y separado por un espacio la proporción con formato de strings a dos cifras decimales * Calcular la media geométrica de las alertas con las proporciones porcentuales * Calcular la media geométrica de los tratamientos con las proporciones porcentuales * Imprimir la media geométrica de las alertas con formato de strings a dos cifras decimales * Imprimir la media geométrica de los tratamientos con formato de strings a dos cifras decimales   Utilice la siguiente formula para el cálculo de la media geométrica:  **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | benigno 0.33  levemente sospechoso 0.33  moderadamente sospechoso 0.33  no aaf 0.33  seguimiento 0.33  aaf 0.33  0.33  0.33 | | **Entrada esperada** | | 4  C4  E4  F2  M2  0  1  1  1  0.52  C3  E4  F2  M2  0  0  1  0  0.61  C2  E4  F2  M1  0  1  1  0  1.94  C1  E2  F1  M2  0  0  0  1  2.82 | | **Salida esperada** | | moderadamente sospechoso 0.25  altamente sospechoso 0.75  seguimiento 0.25  aaf 0.75  0.43  0.43 |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |

**Reto 1.91**

|  |  |
| --- | --- |
| Nombre del reto: | Clasificación de nódulos tiroideos y acciones a tomar |
| Autor reto: | Rocío Ramos Rodríguez |
| Descripción del reto con su respectiva solución: | |
| En el año 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. La buena salud es esencial para el desarrollo sostenible, y la Agenda 2030 busca reforzar la capacidad de todos los países, en particular los países en desarrollo, en materia de la alerta temprana, reducción de riesgos y gestión de los riesgos para la salud nacional y mundial.  El TI-RADS (Thyroid Imaging Reporting and Data Systems) es una prueba para la clasificación de nódulos tiroideos. Basándose en cinco características de los nodos como lo son la composición, ecogenicidad, forma márgenes y focos ecogénicos. Cada característica tiene un puntaje y según el total de punto se describe si el nódulo es benigno o no y, además, si requiere seguimiento o intervención con aguja fina.  El Ministerio de Salud le pide que lo ayude en el desarrollo de un programa para la emisión de alertas tempranas de acuerdo a las características del nodo. Para ello debe tener en cuenta las tablas siguientes:   |  |  |  | | --- | --- | --- | | **Composición** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | C1 | Puntos quísticos o casi completamente quísticos | 0 | | C2 | Espongiforme | 0 | | C3 | Mixto quístico y sólido | 1 | | C4 | Sólido o casi completamente sólido | 2 |  |  |  |  | | --- | --- | --- | | **Ecogenicidad** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | E1 | Anecoico | 0 | | E2 | Hiperecoico o isoecoico | 1 | | E3 | Hipoecoico | 2 | | E4 | Muy hipoecoico | 3 |  |  |  |  | | --- | --- | --- | | **Forma** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | F1 | Más ancho que alto | 0 | | F2 | Más alto que ancho | 3 |  |  |  |  | | --- | --- | --- | | **Margen** | | | | **SE DEBE ESCOGER SOLO UNO** | | | | **Código** | **Clasificación** | **Puntaje** | | M1 | Suave | 0 | | M2 | Bien definido | 0 | | M3 | Lobulado o irregular | 2 | | M4 | Extensión extra-tiroidea | 3 |  |  |  |  | | --- | --- | --- | | **Focos ecogénicos** | | | | **SE DEBEN SELECCIONAR TODOS LOS QUE APLIQUEN** | | | | **Código** | **Clasificación** | **Puntaje** | | FE1 | Ninguno o grandes artefactos en cola de cometa | 0 | | FE2 | Macrocalcificaciones | 1 | | FE3 | Calcificaciones periféricas (borde) | 2 | | FE4 | Focos ecogénicos punteados | 3 |   Después de calculado el puntaje la clasificación de los nódulos es la siguiente:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 - 1 puntos | 2 puntos | 3 puntos | 4 - 6 puntos | 7 o más | | Benigno | No sospechoso | Levemente sospechoso | Moderadamente sospechoso | Altamente sospechoso | | No AAF | No AAF | AAF si tamaño ≥ 2.5  Seguimiento si tamaño < 2.5 | AAF si tamaño ≥ 1.5  Seguimiento si tamaño < 1.5 | AAF si tamaño ≥ 1  Seguimiento si tamaño < 1 |   \*\* AAF = aspiración con aguja final.  El programa deberá entonces:   * Leer una variable N que indique el número de pacientes que se analizarán * Leer los datos de los N pacientes. Siempre serán 9 características por pacientes. Es decir, la matriz será de N x 9. * Calcular un vector con las alertas de los nódulos de cada paciente * Calcular un vector con los tratamientos que requieren cada paciente * Calcular un nuevo vector donde los elementos sean los tamaños de los nódulos divididos entre el valor máximo * Imprimir cada elemento del vector de alertas en una línea nueva * Imprimir cada elemento del vector de tratamientos en una línea nueva * Imprimir cada elemento del nuevo vector calculado en una línea nueva con formato de strings a dos cifras decimales   **Ejemplos:**   |  | | --- | | **Entrada esperada** | | 3  C1  E1  F2  M3  0  1  0  0  2  C1  E1  F1  M1  0  1  1  0  1  C1  E1  F1  M1  1  0  0  0  2 | | **Salida esperada** | | moderadamente sospechoso  levemente sospechoso  benigno  aaf  seguimiento  no aaf  1.00  0.50  1.00 | | **Entrada esperada** | | 2  C1  E1  F1  M1  0  1  1  0  1  C4  E4  F2  M4  1  1  1  1  2 | | **Salida esperada** | | levemente sospechoso  altamente sospechoso  seguimiento  aaf  0.50  1.00 |   **Nota:** Por favor **NO** incluya mensajes en los inputs.  **Nota:** Las tildes y cualquier otro signo ortográfico han sido omitidos a propósito en las entradas y salidas del programa. **Por favor NO use ningún signo ortográfico dentro del desarrollo de su solución** ya que estos pueden representar errores en la calificación automática de Codegrade.  **Nota:** El archivo debe llamarse **reto1.java**, de lo contrario no podrá ser cargado en la plataforma de Codegrade. | |