



## Taller Final - Conceptos Básicos en JavaScript

### 1. Divergente – Análisis de resultados de la Ceremonia de Selección

Ha llegado el momento de una nueva Ceremonia de Selección y Jeanine Matthews, líder de la facción Erudición, ha puesto en marcha su plan para erradicar las facciones que se le oponen en su plan por la dominación total. Su justificación para los ataques dependerá de la proporción de nuevos miembros en la facción Osadía, así como la aparición de una cantidad representativa de Divergentes.

Jeanine ha identificado que, si la cantidad de miembros de Osadía más los Divergentes supera el 40% del resultado de selección, puede dar marcha a su maléfico plan. En una ceremonia de selección, 20 nuevos miembros escogerán una facción: Abnegación (A), Cordialidad (C), Erudición (E), Osadía (O) y Verdad (V). Cualquier otra letra será entendida como que ha aparecido un Divergente.

Implemente una aplicación que lea las 20 asignaciones de facciones realizadas en la Ceremonia de Selección y calcule los porcentajes para cada una de las facciones incluyendo el porcentaje de Divergentes. Para cada facción o divergentes, debe indicar el total de miembros y el porcentaje obtenido.

- Si el porcentaje de miembros de Osadía más el porcentaje de Divergentes supera el 40% del total, la aplicación debe decir “Jeanine, puedes proceder con la dominación total!”.
- Si el porcentaje de miembros de Erudición es el menor de todos y no hay Divergentes, la aplicación debe decir “Janine, ite tocó aplazar tu maléfico plan!”



## **2. Harry Potter - Análisis de resultados del sombrero seleccionador**

Preocupados por los acontecimientos que llevaron a la subida al poder del señor oscuro, el Ministerio de Magia ha decidido enviar a su mejor estadista, Yallow Vaddlesprout, para que analice estadísticamente los resultados de las sesiones de clasificación que El Sombrero Seleccionador hace en Howgarts. Se quiere identificar en etapas tempranas cuando una casa estará en posición dominante respecto a otra.

La posición dominante se da cuando el porcentaje de nuevos miembros de una casa es superior al de los miembros de otra casa. Por ejemplo, si Ravenclaw obtiene 24% y Hufflepuff obtiene 12%, Ravenclaw tiene posición dominante sobre Hufflepuff.

En una sesión de clasificación, 16 nuevos estudiantes recibirán asignación a una casa: Gryffindor (G), Ravenclaw (R), Slytherin (S) y Hufflepuff (H). Cualquier otra letra se entenderá como un Hatstall (indecisión del sombrero) y se preguntará nuevamente. Los 16 valores deben ser solamente H, G, R y S.

Implemente una aplicación que lea las 16 asignaciones de casas realizadas por el sombrero seleccionador, calcule los porcentajes de cada una de las casas e identifique cuales son las posiciones dominantes que se puedan presentar. Si no hay posición dominante, la aplicación debe indicar que fue una clasificación normal.



### **3. Nueva administración en La Cabaña del Misterio en Gravity Falls**

Decepcionada por la forma como se maneja la Cabaña del Misterio, Mabel Pines le hace una apuesta al tío Stan creyendo que en un fin de semana puede recoger más dinero utilizando técnicas modernas de administración. El tío Stan acepta y mientras él se retira a participar en un concurso de televisión, Mabel se convierte en La Jefa.

Mabel define una meta de dinero a recoger y les asigna a todos, la responsabilidad de recoger una cuota correspondiente a la cuarta parte del meta. Coloca cuatro frascos de vidrio vacíos, con los nombres de los responsables: 1. Wendy, 2: Dipper, 3: Soos y 4: Mabel.

Al final de la jornada, Mabel utilizará una aplicación para contabilizar el dinero recolectado por todos y generará indicadores para cada responsable, especificando si cumplieron, sobrepasan o se quedaron cortos en la cuota asignada. La aplicación indicará qué porcentaje del total recolectado fue conseguido por cada responsable.

La aplicación entonces:

- Recolectarán el valor tipo int de la meta establecida por Mabel.
- Recolectaron cuatro valores tipo int correspondientes a lo recogido por cada responsable y los almacenará en un arreglo.
- Implementará una función que recibe un dato tipo int correspondiente al código del responsable y devolverá el nombre del responsable en un dato tipo string, según el contexto presentado.
- Implementará una función que recibe un parámetro tipo int correspondiente a lo recogido por el responsable y un parámetro tipo float con el valor de la cuota asignada y devolverá un resultado tipo string indicando si cumplió, superó o se quedaron cortos en la cuota asignada.
- Implementará una función que recibe un arreglo tipo int correspondiente a lo recogido por los responsables y devolverá un arreglo tipo float con los porcentajes correspondientes a cada responsable.

La aplicación al final visualizará como resultados, el total recolectado por todos, el valor individual de cada responsable y su correspondiente porcentaje del total. Si el total recolectado es igual o superior a la meta establecida por Mabel, debe escribir la frase “Meta Cumplida, ¡eres buena Jefe Mabel!”.



#### **4. Gestión Estadística de la Batalla de los Cinco Ejércitos**

Es la noche después de que Bardo El Arquero con sorprendente puntería mató a Smaug El Dragón utilizando La Flecha Negra. Ya no hay limitaciones para que se acceda a la riqueza que se encuentra en las entrañas de Erabor, La Montaña Solitaria.

Gandalf, conociendo las habilidades tecnológicas de los Hobbits, le encarga Bilbo una aplicación en JAVA que le permita llevar registro del enfrentamiento que se llevará a cabo entre Enanos, Elfos, Humanos, Orcos y Huargos. La idea es que en cualquier momento Gandalf le pregunte a Bilbo como va la situación y éste le responda diferentes indicadores de gestión sobre la batalla.

Bilbo hace una aplicación que utiliza un arreglo de 5 posiciones para almacenar los muertos por raza, utilizando el siguiente orden: 1: Enanos, 2: Elfos, 3: Humanos, 4: Orcos y 5: Huargos. La aplicación pedirá la cantidad de muertos por raza y lo almacenará en el arreglo.

La aplicación utilizará:

- Una función que recibe un dato tipo int correspondiente al código de la raza y devolverá el nombre de la raza en un dato tipo string, según el contexto presentado.
- Una función que recibe el arreglo de muertos y devuelve un dato tipo string con el nombre de la raza que más muertos lleva en la batalla.
- Una función que recibe el arreglo de muertos y devuelve un dato tipo string con el nombre de la raza que menos muertos lleva en la batalla.

La aplicación al final visualizará como resultados, el total global de muertos en combate, qué porcentaje corresponde a cada raza y cuáles son las razas con más y menos muertos.

En este escenario, tu eres Bilbo. Realiza una aplicación que cumpla con las especificaciones solicitadas.



## 5. Evaluación de Traumas de Morty causados por Rick

Dentro del juicio que la Federación Galáctica le está haciendo a Rick Sánchez por los crímenes cometidos, ha llamado la atención a los jueces el maltrato que le ha causado a su nieto Morty Smith. Se ha identificado que, en repetidas ocasiones, Morty ha sido sometido a situaciones que le han causado pánico, trauma mental e inclusive daño físico.

La corte ha decidido contratar a un especialista en maltrato infantil, el cual revisará 20 situaciones en las que los fiscales basan las acusaciones contra Morty. El especialista calificará la gravedad de cada situación, asignándole una calificación en una escala de 1 a 10.

Para ello, el especialista utilizará una aplicación desarrollada en JAVA, la cual le preguntará qué tipo de maltrato es, utilizando una letra de la siguiente forma: **(P)** para Pánico, **(T)** para

Trauma mental y **(F)** para daño físico. Para cualquier otra letra, la aplicación indicará que la clasificación es errónea y volverá a solicitar el tipo de maltrato. Luego, la aplicación le preguntará al especialista cuál es la calificación de la gravedad de la situación, en un número que va desde 0 hasta 10, con posibilidad de dígitos decimales.

Luego de recolectar las 20 calificaciones, el programa informará cuantas situaciones corresponden a cada tipo y cuál es la calificación promedio de cada tipo. Si el promedio de la calificación de gravedad de las situaciones de Trauma mental **(T)** o de Daño físico **(F)** superan el valor de 7.0, la aplicación debe recomendar máximo castigo para el acusado.

### Datos de entrada:

20 calificaciones de situación para **(P)** para Pánico, **(T)** para Trauma mental y **(F)** para daño físico. Cada una de ellas con valor numérico con dígitos decimales.

### Datos de Salida:

Cuántas situaciones por cada tipo de maltrato, el promedio de calificación de cada tipo y la recomendación de castigo en caso de que supere el valor de 7.0 para las situaciones tipo T o F.



**Implemente una función que reciba como parámetros:**

- Un arreglo que contiene en cada posición la cantidad de eventos por tipo de maltrato.
- Un arreglo que contiene en cada posición la suma de las calificaciones de gravedad por cada tipo de maltrato.

La función devuelve un arreglo con las calificaciones promedio de cada tipo de maltrato.

## **6. Manos de Poker**

Una baraja de naipes francesa está conformada por cartas identificadas por una pinta y una figura. Las pintas son **(T)**réboles, **(D)**iamantes, **(P)**icas y **(C)**orazones. Las figuras son 2, 3, 4, 5, 6, 7, 8, 9, J, Q, K y A.

Entonces el Tres de trébol se representa como 3T, el cinco de diamantes como 5D y el ocho de picas como 8P.

Realice un programa que reciba la información correspondiente a 3 cartas y escriba la frase **“trio”** si las tres cartas son de la misma figura, pero de distinta pinta; que escriba la frase **“escalera”** si las figuras son consecutivas de la misma pinta o que escriba **“tres cartas sin relación”** cuando no cumple ninguno de los dos criterios anteriores.



## 7. Procesamiento de cadenas de ADN

En un laboratorio de genética están analizando muestras de ADN para la identificación de las proteínas que la conforman. Cada muestra está conformada por las iniciales de las proteínas, así: **(A)**denina, **(C)**itosina, **(G)**uanina y **(T)**imina.

### Requerimientos:

- Realice un programa en JAVA, que utilizando un ciclo FOR y un arreglo, pregunte y almacene los 20 valores para las proteínas que conforman la secuencia. Los valores pueden ser tipo string.
- En el programa, implementa una función llamada ValidaSecuencia, que reciba el arreglo de valores y devuelva un número tipo entero correspondiente la cantidad de valores correctos. El programa debe visualizar en la función principal, el valor devuelto por la función.
- Solo si los 20 valores son correctos, el programa debe proceder a contar cuántos valores hay por cada proteína. Para ello, implemente una función llamada CuentaProteinas que reciba el arreglo de valores y devuelva un arreglo de resultados. Dicho arreglo tendrá 4 posiciones: [0]: Adenina, [1]: Citosina, [2]: Guanina, [3]: Timina. El programa debe visualizar en la función principal, los valores devueltos por la función en el arreglo de resultados.
- Si los valores correctos no son 20, el programa debe indicar el mensaje “Error en la secuencia. Hay valores que no corresponden a las proteínas válidas”.