**SOLUCIÓN ALGORÍTMICA PARA LAS PASOS 2019**

**DISTRITO CABA, NACONALES**

Se debe desarrollar una solución algorítmica para el análisis de los resultados de las elecciones PASO Nacionales en la Ciudad Autónoma de Buenos Aires correspondientes al año 2019, donde se deben renovar trece bancas, correspondientes a igual número de diputados.

Se presentan 5 listas con 25 aspirantes cada una, los primeros 13 corresponden a candidatos” titulares” mientras que los restantes son candidatos “suplentes”.

Una vez realizada la elección, para el escrutinio (conteo de votos) se utiliza el Sistema D´Hont

[[1]](#endnote-1), mecanismo que se utiliza para la asignación de las bancas.

En el siguiente ejemplo podemos observar cómo se asignan las bancas, suponiendo, que se deben distribuir 5 bancas entre 7 listas:

1. Para obtener una banca, al menos se deben obtener el 3% de los votos:   
   Una vez contados todos los votos válidos (no se consideran votos en blanco ni votos nulos) de cada lista votada se descartan aquellos que obtuvieron menos del 3%. Supongamos en el ejemplo que en 4 listas no se alcanzó el 3%, por lo tanto las bancas se van a repartir en las 4 listas restantes.
2. Se confecciona una grilla en un papel, donde las filas representan las listas y una columna por cada banca votada (en el ejemplo son 5 columnas). El orden de la lista es de mayor a menor en función de la cantidad de votos obtenidos por cada lista

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Listas** | **Cantidad de votos** | **% votos válidos** | **Primera banca** | **Segunda banca** | **Tercera banca** | **Cuarta banca** | **Quinta banca** |
| **1 BLANCA** |  |  |  |  |  |  |  |
| **2 VERDE** |  |  |  |  |  |  |  |
| **3. ROJA** |  |  |  |  |  |  |  |
| **4. AZUL** |  |  |  |  |  |  |  |
| **5. ROJA Y BLANCA** |  |  |  |  |  |  |  |
| **6. VERDE Y BLANCA** |  |  |  |  |  |  |  |
| **7. AZUL Y BLANCA** |  |  |  |  |  |  |  |
| **VOTOS EN BLANCO** |  |  |  |  |  |  |  |
| **VOTOS NULOS** |  |  |  |  |  |  |  |

1. El resultado de cada partido se debe dividir en 1, 2, 3, 4 y 5 (dado que son 5 las bancas en juego). Por ejemplo si el partido 1 obtuvo 100 votos, se deben obtener los siguientes valores: 100, 50 33, 25 y 20.
2. Luego se consideran los cinco valores más elevados, y de acuerdo a la cantidad de valores de cada lista serán los diputados que obtengan bancas.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Listas** | **Cantidad de votos** | **% votos válidos** | **Primera banca** | **Segunda banca** | **Tercera banca** | **Cuarta banca** | **Quinta banca** | **Ganan:** |
| **1 BLANCA** | 100 | 37% | **100** | **50** | 33 | 25 | 20 | OBTIENE 2 BANCAS |
| **2 VERDE** | 80 | 29% | **80** | **40** | 27 | 20 | 16 | OBTIENE 1 BANCA |
| **3. ROJA** | 70 | 26% | **70** | 35 | 23 | 18 | 14 | OBTIENE 1 BANCA |
| **4. AZUL** | 5 | 2% |  |  |  |  |  | OBTUVO MENOS DEL 3%, QUEDA DESCARTADA |
| **5. ROJA Y BLANCA** | 5 | 2% |  |  |  |  |  | OBTUVO MENOS DEL 3%, QUEDA DESCARTADA |
| **6. VERDE Y BLANCA** | 5 | 2% |  |  |  |  |  | OBTUVO MENOS DEL 3%, QUEDA DESCARTADA |
| **7. AZUL Y BLANCA** | 5 | 2% |  |  |  |  |  | OBTUVO MENOS DEL 3%, QUEDA DESCARTADA |
| **VOTOS EN BLANCO** | 10 |  |  |  |  |  |  |  |
| **VOTOS NULOS** | 5 |  |  |  |  |  |  |  |

1. Finalmente la lista BLANCA obtiene 2 bancas, la lista VERDE obtiene 2 bancas y la lista ROJA obtiene 1 banca. El resto de las listas no obtiene bancas por haber obtenido menos del 3% de los votos válidos.

El trabajo práctico consiste en desarrollar algoritmos que cumplan con las especificaciones de más abajo, utilizando únicamente las herramientas y estructura de datos trabajadas durante la cursada. Por cada entrega deben enviar mail indicando el grupo (entre 3 y 4 estudiantes), número de entrega, y con un archivo adjunto en formato comprimido “rar” que contenga: diagrama de la solución en formato digital, código fuente con la solución, texto con un detalle del uso del/los programas, imagen de las pantallas, lote de prueba con datos de entrada y resultados esperados.

|  |
| --- |
| El trabajo práctico consiste en tres ejes: arreglos; archivos; estructuras enlazadas.   * El primer eje se debe entregar la primera semana después del receso de julio. * El segundo eje se debe entregar antes de la finalización del mes de septiembre de 2018. * El tercer eje antes de la finalización de la cursada hacia finales de noviembre de 2018. |

Al finalizar la cursada, los estudiantes que estén en condiciones de acceder a la APROBACIÓN DIRECTA deberán entregar una carpeta con todo el material entregado durante la misma.

**[Temas trabajados: arreglos]**

Se debe desarrollar un algoritmo para:

1. Describir la estructura necesaria para la resolución del problema
2. Mostrar por pantalla una tabla indicando claramente los números de lista, nombre de la lista, la cantidad de votos de cada lista y porcentaje entre los votos válidos, ordenado de mayor a menor, la cantidad de votos nulos y de votos en blanco.
3. Mostrar por pantalla una tabla como la del ejemplo de más arriba, indicando claramente para cada lista los nombres de los candidatos que obtienen banca
4. Mostrar por pantalla, para cada lista la cantidad de votos por rango de edades: hasta 18 años; hasta 30 años; hasta 50 años; más de 50 años.

Para desarrollar el algoritmo se cuenta con las siguientes fuentes de datos:

1. Ingresar por el teclado los números de lista; el nombre de la lista y el nombre de cada uno de los candidatos
2. Ingresar por teclado la información de cada uno de los votos (número entre 1 y 7 o algún otro valor; sexo del votante; edad del votante en años). Los números entre 1 y 7 son votos válidos, mientras que cualquier otro valor es un voto nulo. El valor 0 representa un voto en blanco.

**[Temas trabajados: archivos]**

Se debe desarrollar algoritmos que permitan que los datos a procesar (número y nombre de las listas; votos; resultados) permanezcan en archivos para próximas corridas. Se deben desarrollar algoritmos para:

1. Describir la estructura necesaria para la resolución del problema
2. Pedir por pantalla y grabar en un archivo los números; nombres de las listas y nombres de cada uno de los candidatos
3. Pedir por pantalla, ordenar y guardar en un archivo los votos emitidos: número de lista
4. Modificar los algoritmos (2, 3 y 4) de la primer entrega teniendo en cuenta que los datos a procesar están en archivos y los resultados se deben mostrar por pantalla y guardar en un archivo

**[Temas trabajados: estructuras enlazadas, memoria dinámica]**

Se debe desarrollar algoritmos que permitan que, a partir de los datos de los archivos desarrollados en el punto anterior, se armen listas con sublistas, donde las listas representan las listas votadas y las sublistas representan cada voto. Se pide:

1. Describir la estructura de listas y sublistas que “soporten” los datos de la estructura
2. Definir claramente el orden de las listas y sublistas teniendo en cuenta los requerimientos descriptos más abajo
3. Desarrollar un algoritmo que “levante” a memoria en las estructuras definidas toda la información de la elección.
4. Desarrollar un algoritmo que muestre por pantalla cuántas mujeres y cuántos varones votaron cada lista
5. Desarrollar un algoritmo que muestre por pantalla la secuencia de votos ordenados por edad en años, indicando la edad y si es varón o mujer

**NOTA: para el desarrollo de todos los temas se deben utilizar las funciones de biblioteca que se trabajan en la clase, las cuales deben implementar en sus programas.**

1. Ver fuentes explicativos publicados: para las elecciones 2013 por La Nación S.A. en:

   * <https://www.youtube.com/watch?v=0-HkiOiPa1U>;
   * <https://www.youtube.com/watch?v=kXbUXugLnlc>;
   * <https://prezi.com/v5hhzlo266fe/sistema-dont/>;
   * <http://www.estadisticaparatodos.es/taller/electoral/electoral.html>

   [↑](#endnote-ref-1)