

Algoritmos y estructura de datos 2020 K1091

Trabajo práctico anual 2020

Para realizar en grupo

SOLUCIÓN ALGORÍTMICA PARA LAS PASOS 2021

DISTRITO CABA, NACONALES

El trabajo práctico consiste en el desarrollo de programas que abordan tres ejes: arreglos, archivos y estructuras enlazadas.

Fecha de entrega	Eje
Agosto 2020, después receso	primero: arreglos
Fin de septiembre 2020	segundo: archivos
Fin de noviembre 2020	tercero: estructuras enlazadas

El trabajo práctico consiste en desarrollar algoritmos que cumplan con las especificaciones que se detallan en las siguientes filminas, utilizando únicamente las herramientas y estructura de datos trabajadas durante la cursada.

Por cada entrega deben subir un archivo en formato comprimido “rar” que contenga:

- portada con la composición del grupo
- especificaciones del programa
- 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.

Descripción del problema

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 2021, donde se deberán renovar trece bancas, correspondientes a igual número de diputados nacionales.

Se presentan 5 listas con 25 candidatos 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 utilizará el Sistema D'Hont, mecanismo que se utiliza para la asignación de las bancas.

Para una correcta comprensión del problema planteado es importante que vean documentación acerca de la distribución de bancas en una elección, para lo cual les proponemos varios links (no son los únicos publicados).

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

1. Por cada lista: número de lista; nombre de la lista y nombre de cada uno de los candidatos
2. 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.

Ver fuentes explicativas 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=kXbUXugLnIc>;
- <https://prezi.com/v5hhzlo266fe/sistema-dont/>;
- <http://www.estadisticaparatodos.es/taller/electoral/electoral.html>

Breve síntesis del sistema D'Hont a través de un ejemplo

Se deben asignar 5 bancas luego de una elección para lo cual se presentaron 7 listas

MÉTODO DE ASIGNACIÓN D'HONT

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 ALCANZARON** el 3%, por lo tanto las bancas se van a **REPARTIR** en las 3 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
3. El resultado de cada lista se debe dividir en 1, 2, 3, 4 y 5 (dado que son 5 las bancas en juego). Por ejemplo si la lista 1 obtuvo 100 votos, se deben obtener los siguientes valores: 100, 50 33, 25 y 20.
4. 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.
5. 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.

Distribución de las bancas luego de aplicar el sistema D’Hont

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 2 BANCAS
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							

[Temas trabajados primera entrega: arreglos]

Se debe desarrollar un algoritmo para distribuir las bancas de acuerdo al conteo de votos.

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

[Temas trabajados segunda entrega: 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.

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 por 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

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

[Temas trabajados tercera entrega: estructuras enlazadas, memoria dinámica]

Se debe desarrollar un algoritmo que permita 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.

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 descritos más abajo
3. Desarrollar un algoritmo que “levante” a memoria en pantalla cuántas mujeres y cuántos varones votaron cada lista
4. 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

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