

Práctico Nº 6

Tema: Lenguaje de Diseño. Estructuración de Datos (2° Parte)

Nota: para los ejercicios que pidan el desarrollo de un algoritmo, se debe utilizar la técnica de refinamiento sucesivo. (al menos dos versiones de desagregación además de la versión final escrita en lenguaje de diseño).

4. Diseñar un algoritmo que permita ingresar una cadena de no más de 20 caracteres. Posteriormente, invierta dicha cadena. La cadena original y la cadena invertida deben almacenarse en arreglos diferentes. Finalmente, el algoritmo debe mostrar ambas cadenas por pantalla. Se pide además, realizar el diagrama de flujo del algoritmo y ejecutarlo utilizando como entrada la cadena "NEUQUEN".

Ejemplo: cadena ingresada: "**TERREMOTO**" cadena resultante: "**OTOMERRET**"

5. Diseñar un algoritmo que permita ingresar una cadena de no más de 30 caracteres. Luego, mostrar la cantidad de veces que aparece cada una de las vocales mayúsculas o minúsculas (*sin mostrar las que no aparecen*). Posteriormente, realizar el diagrama de flujo del algoritmo y ejecutarlo utilizando como datos de entrada la cadena del ejemplo siguiente.

Ejemplo: cadena ingresada: "**#biologia\$\$**"

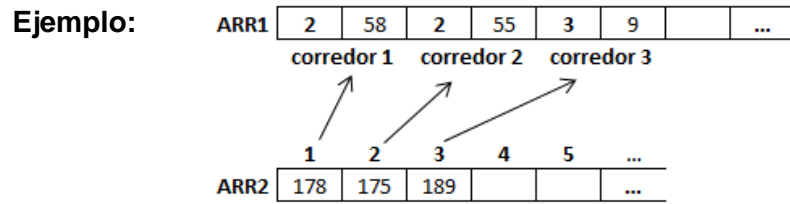
cantidad letra **a**: 1 cantidad letra **i**: 2 cantidad letra **o**: 2

6. Diseñar un algoritmo que permita ingresar una palabra de hasta 10 caracteres y mostrarla por pantalla. Luego, inserte un carácter en dicha palabra. El usuario ingresará el carácter que desea insertar y la posición donde debe insertarse. Finalmente, mostrar por pantalla la palabra modificada.
7. Diseñar un algoritmo que permita ingresar no más de 50 números enteros positivos (la cantidad de números la decide el usuario). Los números pares se almacenarán a partir de la posición 1 y los números impares a partir de la última posición del arreglo. Posteriormente, mostrar por pantalla los números ingresados de la siguiente manera: en primer lugar se mostrarán los números impares y luego los pares, en ambos casos en el orden que fueron ingresados. Finalmente, mostrar por pantalla la cantidad de números pares e impares.
8. Se necesita almacenar el número de alumnos del nivel secundario de la Esc. No 48 Gral. San Martín. El nivel secundario posee 6 cursos mixtos (varones y mujeres). Por cada curso existen 2 divisiones (A y B) y se necesita diferenciar el número de varones y mujeres existentes en cada división de cada curso. Se pide diseñar y dibujar de que manera estructurar los datos utilizando:
 - a. La cantidad de estructuras que considere necesarias.
 - b. Una única estructura.

Nota: NO se pide diseñar el algoritmo. Sólo hay que decidir y especificar claramente cómo almacenar los datos.

9. Diseñar un algoritmo que permita ingresar y almacenar adecuadamente en un único arreglo, la altura y el sexo de hasta 100 personas. Luego, se debe informar cuántas mujeres tienen una altura entre 1,70 y 1,80 y el sexo de la/s persona/s de mayor altura.
10. Diseñar un algoritmo que permita ingresar en una única estructura, los siguientes datos de las habitaciones de un hotel: número de habitación, capacidad (2, 4 o 6) y disponibilidad (SI/NO). El hotel cuenta con 100 habitaciones. Se pide, informar el número de las habitaciones que se encuentren desocupadas, con cierta capacidad (dicha capacidad debe ser ingresada por el usuario). Luego, se debe informar el porcentaje de habitaciones ocupadas y desocupadas del hotel.

11. Diseñar un algoritmo que permita ingresar en un único arreglo, los tiempos de carrera (en segundos) de no más de 25 corredores de 1.500 metros. Posteriormente, el algoritmo deberá almacenar en otra estructura adecuada, el tiempo expresado en minutos y segundos por cada corredor. Finalmente, informar cuantos corredores corrieron en menos de 3 minutos.



Ejercicios adicionales

- Diseñar un algoritmo que permita ingresar en un arreglo una cadena de hasta 40 caracteres. Luego, el algoritmo debe chequear la posición en donde se encuentran las vocales minúsculas. Para ello construir un nuevo arreglo de tipo lógico en el cual se almacene un valor VERDADERO en las posiciones donde se encuentra cada vocal minúscula y un FALSO donde no se encuentran. El algoritmo deberá mostrar por un lado las posiciones donde se encuentran las vocales en el arreglo y por otro las posiciones donde se encuentran los demás caracteres.
- Diseñar un algoritmo que permita concatenar dos cadenas de caracteres ingresadas por el usuario. La longitud de cada cadena ingresada no debe superar los 10 caracteres. Se debe utilizar una única estructura para almacenar las tres cadenas. Posteriormente, mostrar por pantalla las cadenas ingresadas y la resultante de la concatenación. Realizar el diagrama de flujo correspondiente y ejecutar el algoritmo codificado utilizando como datos de entrada: "voley" y "ball".