

PROGRAMACIÓN

Programador Universitario - Licenciatura en Informática - Ingeniería en Informática
Facultad de Ciencias Exactas y Tecnología - UNT

Trabajo Práctico N° 6

TEMA: Punteros. Mecanismos de Pasaje de Parámetros. Asignación dinámica de memoria.

“Un puntero es una variable que contiene la dirección de una variable”. Kernighan & Ritchie

CONSIGNA DE TRABAJO



Para los desplazamientos en los arreglos, use notación de punteros. Siempre que sea posible declare los arreglos de forma dinámica.

1. Calentando motores

- Escriba un programa que lea dos números enteros y los almacene en las variables a y b. Muestre en pantalla los valores leídos en a y b.
- Cree dos punteros enteros llamados p y q. Asigne en los punteros p y q la dirección de memoria de a y b respectivamente.
- Muestre el contenido de p y q.
- Muestre el contenido apuntado por p y q.
- Muestre las direcciones de memoria de p y q.
- Lea un número y guárdelo en una variable llamada c. Muestre el contenido de c y la dirección de memoria de la variable c.
- Asigne a la variable c el valor de a utilizando el puntero p. Muestre las variables a y c.
- En la variable c guarde la suma de a+b utilizando los punteros p y q. Muestre las variables a, b y c.
- ¿Qué sucede si ejecutamos $*p=*p+1$?
- ¿Qué sucede si ejecutamos $p=p+1$?
- Dada la siguiente asignación: $p=&a$:
- ¿Las sentencias $*p=*p+1$ y $*p=a+1$ son equivalentes?
- ¿Las sentencias $p=p+1$ y $p=&a+1$ son equivalentes?

2. Calentando motores con arreglos

Lea y almacene una línea de texto en minúscula en un arreglo de nombre frase[100]. Cuente las consonantes y conviértelas a mayúsculas.

Use variables punteros para manejar los contadores.

Modifique el código de su programa para sumar un puntero que apunte al arreglo frase. Siga las instrucciones paso a paso:

- Declare en forma local a main una variable puntero **punt**, que apunte a un objeto de tipo char.
- Asigne la dirección inicial del arreglo frase a la variable puntero **punt**. Realice esta tarea de dos maneras distintas.

PROGRAMACIÓN

Programador Universitario - Licenciatura en Informática - Ingeniería en Informática
Facultad de Ciencias Exactas y Tecnología - UNT

03. Muestre en pantalla la dirección de la variable puntero **punt**, use el indicador de formato apropiado. Muestre también la dirección de la primera componente del arreglo **frase** y su contenido.
04. Muestre en pantalla el contenido de la cuarta celda del arreglo **frase**. Refiérase a ella utilizando: notación de subíndice del arreglo y notación de puntero con desplazamiento mediante **punt**.
05. Suponiendo que **punt** apunta al principio del arreglo **frase**, ¿cuál es la dirección referenciada por **punt+5**? ¿Cuál es el valor almacenado en esa dirección?. Al mostrar el valor almacenado en **punt+5**, ¿el puntero se movió del inicio del arreglo?
06. Escriba las instrucciones necesarias para visualizar en la pantalla cada una de las componentes del vector **frase**,
 - a. Utilizando notación de punteros con el puntero **punt**.
 - b. Mediante subíndices del puntero **punt**.
 - c. Para los casos anteriores, ¿en qué posición quedó el puntero **punt**? ¿se ha movido de la componente 0 del arreglo **frase**?

3. Mezclando Cadenas

Lea dos cadenas de caracteres alfabéticos y un número. La tarea consiste en insertar **la segunda cadena en la primera**, a partir del carácter que está en la posición indicada por el número. Muestre la primera cadena modificada.

Use el siguiente prototipo, considerando una cadena auxiliar para formar la mezcla de salida:

```
void mezclarCadenas(char *cad1, char *cad2, int pos);
```

Nota: Realizar los controles que sean necesarios, realice una reserva de memoria para las cadenas de entrada con un tamaño por defecto y ajuste una vez leídas. No use la función concatenar de la biblioteca `string.h`



Sugerencia: Si para resolver el ejercicio usó un arreglo auxiliar de apoyo, intente llevar a cabo la tarea de mezcla sin el uso del mismo. Implemente mediante otra función.

Ejemplo:

Cadena 1: "Primavera" - Cadena 2: "Florecer" - Número: 4

Cadena 1 de Salida: **PriFlorecermavera**

4. Promedio de un curso

Realice un programa que permita calcular el promedio de notas de los alumnos de un curso.

La cantidad de alumnos dependerá del curso (según la cantidad de alumnos que posea).

Simule las notas con la función `rand()` con números aleatorios de 0 a 10 y con 2 cifras decimales.

Calcule y muestre el promedio del curso y la mejor nota.

Utilice los siguientes prototipos:

PROGRAMACIÓN

Programador Universitario - Licenciatura en Informática - Ingeniería en Informática
Facultad de Ciencias Exactas y Tecnología - UNT

```
// Función para simular las notas
void cargarNotas(float *notas, int alumnos);
// Función que calcula y guarda el promedio, además retorna la mejor nota.
float calcularPromedio(float *notas, int alumnos, float *promedio);
```

5. Tiro al blanco

En una Kermés se presenta el juego de tiro al blanco con dardos. El costo de un tiro es de \$350. Los premios según la puntería son:

- El círculo rojo con valor 10 tiene un premio de \$1000.
- El círculo de valor 9 tiene un premio de \$850.
- El círculo de valor 8 recupera los \$350.
- El círculo de valor 7 recupera \$300.
- El resto de círculos no tiene premio pero si el dardo cae en 1 o 2 se descuenta \$100.



Realice un programa que permita ingresar la cantidad de tiros que desea realizar el jugador. Luego, simule los tiros. Por último, muestre el resumen de tiros, cuánta plata gastó o perdió, cuánta plata ganó y el saldo final del jugador.

Utilice las siguientes funciones:

```
//Función para cargar los puntajes de los tiros.
void cargarTiros(int *puntos, int tiros);
//Función que calcula cuánto ganó el jugador.
float calcularGanancia(int *puntos, int tiros, float *perdida);
```



Sugerencia: Modifique el programa anterior para que funcione con un número indeterminado de usuarios. Además de mostrar los resultados de cada jugador, al finalizar el juego para todos los usuarios, muestre el monto total de gasto/perdida por los mismos, el monto total en ganancias y el saldo final. Elija la marca final que considere apropiada.

Para esto modifique la función **calcularGanancia** con el siguiente prototipo:

```
float calcularGanancia(int *puntos, int tiros, float *perdida, float
*totalPerdida, float *totalGanancia)
```

donde:

- **totalPerdida:** acumula el pago/descuento efectuado por los jugadores.
- **totalGanancia:** acumula la cantidad de premios ganados por los jugadores.

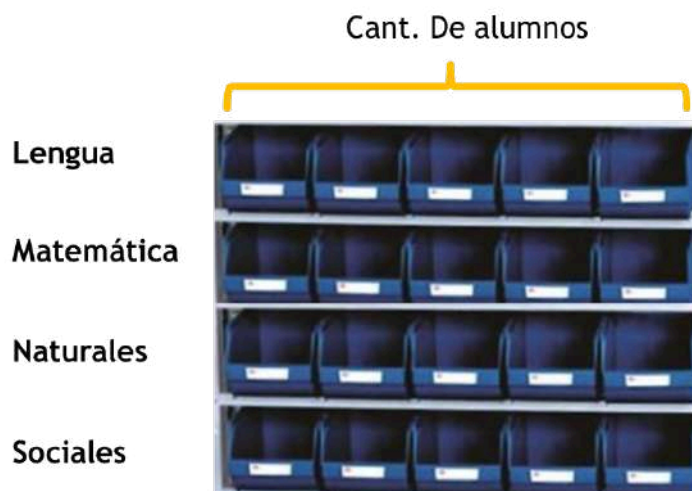
PROGRAMACIÓN

Programador Universitario - Licenciatura en Informática - Ingeniería en Informática
Facultad de Ciencias Exactas y Tecnología - UNT

EJERCICIO EXTRA

Modifique el programa del punto 4 de manera tal que permita cargar las notas de los alumnos de un curso para 4 materias: (0)Lengua, (1)Matemática, (2)Naturales, (3) Sociales. Muestre el promedio de notas y la nota más alta de cada materia.

Nota: Trabaje con una matriz para guardar las notas donde las filas corresponden a la cantidad de materias y las columnas a la cantidad de alumnos de un curso. Guarde en un arreglo las notas promedio y en otro arreglo la mayor nota de cada materia.
Reserve memoria de forma dinámica.



Se sugiere el uso de las siguientes funciones:

```
void cargarNotas(float **notas, int alumnos);
```

```
void calcularPromedio(float **notas, int alumnos, float *promedios, float  
*mayorNotas);
```