Ejercicio de Funciones con 2 números

**Código:**

#include <stdio.h>  
#include <stdlib.h>

// PROTOTIPADO: ----------------------------------------------------------------------------------  
int sumarNumeros(int pnum1, int pnum2);  
int multiplicarNumeros(int pnum1, int pnum2);  
double dividirNumeros(pnum1, pnum2);  
int restarNumeros(pnum1, pnum2);  
void imprimirResultados(int psuma,int pmult,double pdiv, int presta);  
//PROGRAMA PRINCIPAL: ---------------------------------------------------------------------------  
int main(int argc, char \*argv[])  
{  
int num1, num2;  
  
printf("Ingrese numero 1:\n");  
scanf("%d",&num1);  
printf("Ingrese numero 2:\n");  
scanf("%d",&num2);  
  
int suma = sumarNumeros(num1, num2);  
int mult= multiplicarNumeros(num1, num2);  
double div = dividirNumeros(num1, num2);  
int resta = restarNumeros(num1, num2);  
  
imprimirResultados(suma,mult,div,resta);

return 0;  
}  
//FUNCIONES: ---------------------------------------------------------------------------------------  
int sumarNumeros(int pnum1, int pnum2)  
{  
int suma = 0;  
suma = pnum1 + pnum2;  
//printf("La suma es: %d \n",suma);  
  
return suma;  
}

int multiplicarNumeros(int pnum1, int pnum2)  
{  
int mult = 0;  
mult = pnum1 \* pnum2;  
//printf("La mult es: %d \n",mult);  
  
return mult;  
}

double dividirNumeros(int pnum1, int pnum2)  
{  
if(pnum2==0)//Para validar que no de error la division  
{  
printf("Error, no se puede dividir por 0");  
return 0;  
}  
double div = 0;  
div = (double)pnum1 / pnum2;  
  
//printf("La div es: %d \n",div);  
return div;  
}

int restarNumeros(pnum1, pnum2)  
{  
int resta = 0;  
resta = pnum1 - pnum2;  
//printf("La resta es: %d \n",resta);  
  
return resta;  
}

void imprimirResultados(int psuma,int pmult,double pdiv, int presta)  
{  
printf("La suma es: %d \n",psuma);  
printf("La mult es: %d \n",pmult);  
printf("La div es: %lf \n",pdiv);  
printf("La resta es: %d \n",presta);  
}

## Ejemplo Printf y Scanf

//Declaración de variables:

 int edad = 0;  
char nombre[20]=" ";  
double altura = 0;

printf("Ingrese su edad:\n");//Imprime en pantalla ese mensaje  
scanf("%d",&edad); //Solicita el ingreso de dato por teclado y lo asigna a dicha variable  
  
printf("Ingrese su nombre:\n");  
scanf("%s",&nombre);  
  
printf("Ingrese su altura:\n");  
scanf("%lf",&altura);  
  
printf("\nUsted tiene %d anios, se llama %s y mide %lf\n",edad,nombre,altura);

//Imprime las variables con los datos guardados

## Ejemplo de código de implementación de funciones con Array/Arreglos

//PROTOTIPADO: -------------------------------------------------------------------------------------------------------  
void imprimirVector(int pvector[],int pcant);  
int sumarComponentesVector(int pvector[],int pcant);  
int obtenerComponenteMayor(int pvector[],int pcant);  
//PROGRAMA PRINCIPAL: ---------------------------------------------------------------------------------------------------  
int main(int argc, char \*argv[])  
{  
int vector[4];  
srand(getpid());//Sirve para generar numeros diferentes tomando el numero de proceso del sistema operativo  
for(int i=0; i<4;i++)  
{  
vector[i] = rand() % 51;  
}  
int n = sizeof(vector)/sizeof(vector[0]); //obtener el tamanio del vector  
imprimirVector(vector,4);  
printf("\n-La suma de los componentes es %d \n-Tamanio del vector %d\n-El componente mayor es %d",sumarComponentesVector(vector,4),n,obtenerComponenteMayor(vector,4));  
return 0;  
}  
//FUNCIONES: -----------------------------------------------------------------------------------------------------------------  
void imprimirVector(int pvector[],int pcant)  
{  
for(int i=0; i<pcant;i++)  
{  
printf("%d |",pvector[i]);  
}  
}

int sumarComponentesVector(int pvector[],int pcant)  
{  
int suma=0;  
for(int i=0; i<4;i++)  
{  
suma=suma + pvector[i];  
}  
//printf("%d",suma);  
  
return suma;  
}

int obtenerComponenteMayor(int pvector[],int pcant)  
{  
int mayor = -999999999;  
for(int i=0; i<4;i++)  
{  
if (pvector[i]>mayor)  
{  
mayor = pvector[i];  
}  
}  
return mayor;  
}

## Ejemplo de código de ordenamiento método de la burbuja

int main(int argc, char \*argv[])  
{  
int vector[6];  
int aux = 0;  
  
srand(getpid());//Sirve para generar numeros diferentes tomando el numero de proceso del sistema operativo  
printf("Vector desordenado:");  
for(int i=0; i<6;i++)  
{  
vector[i] = rand() % 51;  
printf("%d|",vector[i]);  
}  
//Ordenamiento Ascendente:  
for(int i=0 ; i<6 ; i++)  
{  
for(int j=0 ; j<6 ; j++)  
{  
if(vector[j]>vector[j+1])//j es la posicion siguiente  
{  
aux = vector[j];  
vector[j] = vector[j+1];  
vector[j+1] = aux;  
}  
}  
}  
  
printf("\n\nVector ordenado:");  
for(int i=0; i<6;i++)  
{  
printf("%d|",vector[i]);  
}  
  
/\*Otro metodo  
for(int j=i+1 ; j<6 ; j++)  
{  
if(vector[i]>vector[j])//j es la posicion siguiente  
{  
aux=vector[i];  
vector[i]=vector[j];  
vector[j]=aux;  
}  
\*/

return 0;  
}

## Punteros en C

**Punteros en C:**

Mirar el video completo y practicarlo en Dev C++.

**Código para practicar:**

int main(int argc, char \*argv[])  
{  
//EJEMPLO 1  
int numero=7; //Declaración de variable entera de tipo entero  
int \*puntero; //Declaración de variable puntero de tipo entero  
puntero = &numero; //Asignación de la dirección memoria de a  
printf("El valor del numero es: %d. \nEl valor de \*puntero es: %d. \n",numero,\*puntero);  
printf("La direccion de memoria de \*puntero es: %p",puntero);//Para obtener la direccion de memoria  
cambioDeValor(&numero);  
printf("El valor del numero es: %d. \nEl valor de \*puntero es: %d. \n",numero,\*puntero);  
  
//EJEMPLO 2 (comentar el ejemplo 1 antes de ejecutarlo)  
int num1=2, num2=4;  
printf("numero1 es %d, numero2 es %d",num1,num2);  
  
intercambiaConPunteros(&num1,&num2);  
printf("\n Luego de ejecutar funcion el numero1 es %d, numero2 es %d",num1,num2);

return 0;  
}  
void intercambiaConPunteros(int \*pnum1,int \*pnum2)  
{  
int aux;  
  
aux= \*pnum1;  
\*pnum1= \*pnum2;  
\*pnum2= aux;  
}

void cambioDeValor(int \*pnum)  
{  
\*pnum = 20;  
}

void intercambia(int pnum1,int pnum2)//No utiliza punteros  
{  
int aux;  
aux= pnum1;  
pnum1= pnum2;  
pnum2= aux;  
}

## Código en C de Generar e imprimir una Matriz

#include <stdio.h>  
#include <stdlib.h>

int main(int argc, char \*argv[])  
{  
int matriz[3][3];  
  
// Generar una matriz de 3x3 con numeros generados por el usuario:  
printf("Ingresa los elementos de la matriz de 3x3:\n");  
for (int i = 0; i < 3; ++i)  
{  
for (int j = 0; j < 3; ++j)  
{  
printf("Ingresa el elemento en la posicion [%d][%d]: ", i, j);  
scanf("%d", &matriz[i][j]);  
}  
}  
// Imprimir Matriz:  
printf("\nLa matriz ingresada es:\n");

for (int i = 0; i < 3; ++i)  
{  
for (int j = 0; j < 3; ++j)  
{  
printf("%d\t", matriz[i][j]);  
}  
printf("\n");  
}  
return 0;  
}

## Ejemplo Generar, Obtener mayor, menor, sumar e imprimir Arreglo

PRESTEN ATENCIÓN A LOS NOMBRES DE LAS VARIABLES YA QUE VIENEN DE DIFERENTES EJERCICIOS:

int arregloA [4];

int arregloB [4];

int arregloC [4];

int arregloD [4];

int suma = 0;  
float promedio = 0;  
int mayor = -999;

int menor= 999;

//GENERAR A  
for(int i=0; i<4; i++)  
{  
printf("Ingresar componente %d para A: ",i+1);  
scanf("%d",&arregloA[i]);  
}

//GENERAR B  
for(int i=0; i<4; i++)  
{  
printf("Ingresar componente para B: ");  
scanf("%d",&arregloB[i]);  
}  
  
//GENERAR C  
for(int i=0; i<4; i++)  
{  
arregloC[i]=arregloA[i]+arregloB[i];  
}  
//GENERAR D  
int pos = 3;  
for(int i=0; i<4; i++)  
{  
arregloD[i]=arregloA[i] \* arregloB[pos];  
pos --;  
}

//SUMAR COMPONENTES DEL VECTOR:

for(int i=0; i<4;i++)  
{  
suma=suma + vectora[i];  
}

//OBTENER COMPONENTE MAYOR  
for(int i=0; i<4; i++)  
{  
if(arreglo[i] > mayor)  
{  
mayor = arreglo[i];  
}  
}  
printf("El componente mayor es: %d\n",mayor);

//OBTENER COMPONENTE MENOR  
for(int i=0; i<4; i++)  
{  
if(arreglo[i] < menor)  
{  
 menor = arreglo[i];  
}  
}  
printf("El componente mayor es: %d\n",mayor);

//IMPRIMIR TODOS LOS COMPONENTES DEL VECTOR/ARREGLO:  
for(int i=0; i<4; i++)  
{  
printf("%d|",arregloA[i],i);  
  }  
   
//IMPRIMIR COMPONENTES DE ATRAS PARA ADELANTE:  
printf("\nVector C de atras para adelante: ");  
for(int i=3; i>=0; i--)  
{  
printf("%d |",arregloC[i]);  
}

int tamanioVector = sizeof(arregloA) / sizeof(arregloA[0]);  
printf("\n El tamanio es: %d",tamanioVector);//ESTE CÓDIGO IMPRIME EL TAMAÑO DEL ARREGLO.