

Práctica Funciones con pase de parámetros por referencia y Arreglos

1. Corra en frio los siguientes programas

<pre>void pr(int x,int y,int *z); main() { int a=5,b=8,c=3; pr(a,b,&c); pr(7,a+b+c,&a); pr(a*b,b/a,&c); } void pr(int x,int y,int *z) { *z=x+y+*z; printf(“%d %d %d ”,x,y,*z); }</pre>	
<pre>void p1(int *i,int j,int k); void p2(int h,int *j); void p3(int *i); main() { int i,j,k; i=0; j=1; k=2; p2(0,&k); p2(1,&i); p2(2,&j); } void p1(int *i,int j,int k) { (*i)++; printf(“%d %d %d ”,*i,j,k); } void p2(int h,int *j) { int i; i=*j; if (h==0) p1(j,i,8); else p3(&h); printf(“%d %d %d ”,i,*j,8); } void p3(int *i) { (*i)++; }</pre>	

```
void r1(int *a, int b);
int r2(int i,int a,int k);
void r3(int *a, int i, int k);
```

```
main()
{
    int k=1,x=2;
    r1(&x,k);
    printf("%d",x);}

```

```
void r1(int *a, int b)
{
    int i=0;

    *a=2*b;
    r3(&i,*a,b);
    printf("%d %d ",*a,b);
}

```

```
int r2(int i,int a,int k)
{
    i+=3;
    if (k <= 2)
        printf("%d %d ",i,k);
    return(i+k+a);
}

```

```
void r3(int *a, int i, int k)
{
    int x,y;
    *a=i+k;
    x=r2(*a,i,k);
    y=r2(2*(*a),i,k);
    printf("%d %d ",x,y);
}

```

2. Escriba una función que dado un arreglo A de enteros, el número N de elementos almacenados en dicho arreglo y un entero X, retorne si el elemento está en dicho arreglo. Para ello defina un tipo boolean que devuelva verdadero o falso. Escriba el programa principal que invoca esta función, el cual debe leer los elementos del arreglo y el tamaño antes de invocarla. Suponga que existe una constante MAX para el tamaño máximo del arreglo. El programa principal también debe indicar al usuario si el elemento está o no en el arreglo.
3. Escriba una función que dado un arreglo de caracteres retorne el tamaño del mismo. Escriba dos funciones adicionales que calculen el número de vocales y el número de consonantes de un arreglo.

4. Escriba un programa que utilice una función que recibe dos enteros y un apuntador a un vector (una matriz de una sola fila), intercambie los dos elementos del vector que están en las posiciones dadas.

```
void mueve(int i,int j, int *vector);
```

5. Escriba un programa que dadas las coordenadas de tres puntos $A=(x_1, x_2)$, $B=(y_1, y_2)$ y $C=(x_3, y_3)$, calcule los coeficientes b y m de las ecuaciones de la rectas ($y= b+ m*x$), que pasa por los puntos A,B y A,C . Determine si las rectas pueden ser paralelas o perpendiculares.

Ayuda: Dos rectas son paralelas si sus pendientes m_1 y m_2 son iguales, son perpendiculares si $m_1 = 1/m_2$.

6. Escriba una función que reciba una secuencia de enteros por teclado hasta que se inserte el número 0, y retorne:
- La cantidad de enteros positivos
 - La cantidad de enteros negativos
 - El promedio de los positivos
 - El promedio de los negativos.

7. Describa que hace la función abajo descrita

```
void funcion(int a[], int n )
{
int t ,aux,i;
for(t=0; t<n; t++)
  for(i=n-1; i<t;i--)
    if (a[i] < a[i-1])
      {
        aux=a[i];
        a[i]= a[i-1];
        a[i-1]=aux;
      }
}
```

```
int main ()
{
  int a [] = {5, 2, 8, 3, 1};
  int i;
  for (i = 0; i < 5; i++)
```

```

        printf("%d ", a[i]);
    printf("\n ");

    function(a);

    for (i = 0; i < 5; i++)
        printf("%d ", a[i]);
    printf("\n ");
}

```

8. Escriba un programa que utilice la función “mueve” dada en el ejercicio 4 para ordenar de menor a mayor los elementos de un vector

¿se podría usar la función “mueve” para ordenarlos de mayor a menor?

9. Escriba un programa que use funciones para calcular el cubo de un valor; producir una versión que pase parámetros por valor y otra que pase parámetros por referencia:

//por valor:

```
int cuboValor(int a);
```

//por referencia:

```
void cuboRef(int *a);
```

10. Escriba una función que dado un arreglo de caracteres retorne el tamaño del mismo en un entero, por referencia modifique un contador de vocales y un contador de numeros que tenga la secuencia.

```
int cuenta_cadena(char cad[], int *cont_letras, int *cont_numeros);
```