

### Ejercicio Resueltos

1. Realizar una función llamada `par`, que toma un número entero como parámetro, y devuelve 1 si es par o devuelve 0 si es impar.

```
#include<stdio.h>
int par(int);
int main()
{
    int numero, resultado;
    printf("Introduzca un número:\n");
    scanf("%d",&numero);

    resultado=par(numero);

    if (resultado==1)
        printf("Es par.\n");
    else
        printf("Es impar.\n");
    return 0;
}

int par(int numero)
{
    if((numero%2)==0)
        return(1);
    else
        return 0;
}
```

2. Realizar una función llamada `media2`, que toma dos números reales como parámetros, y devuelve un número real que es la media de los dos números pasados como parámetros.

```
#include<stdio.h>
float media2(float n1, float n2);

int main()
{
    float n1, n2, resultado;
    printf("Introduzca un número real:\n");
    scanf("%f",&n1);
    printf("Introduzca otro número real:\n");
    scanf("%f",&n2);

    resultado=media2(n1,n2);
    printf("La media es: %f.\n",resultado);

    return 0;
}

float media2(float n1, float n2)
{
    float resultado;
    resultado=(n1+n2)/2;
    return resultado ;
}
```

3. Realizar un programa que calcule el área de un rectángulo. Observar que el prototipo sólo tiene el tipo de datos, sin nombre de variables, está permitido.

```
#include<stdio.h>

int arearect(int, int);

int main()
{
    int suprect, mayor, menor;
    printf("Ingrese en centímetros el lado mayor: ");
    scanf("%d", &mayor);
    printf("Ingrese en centímetros el lado menor: ");
    scanf("%d", &menor);
    suprect = arearect(mayor, menor);
    printf("El area del rectangulo es: %d cm2", suprect);
    return 0;
}
```

	<pre> }  int arearect(int a, int b) {     int suprect;     suprect = a * b;     return suprect; } </pre>
<p>4. Realizar una función llamada media3, que toma tres números reales como parámetros, y no devuelve nada. Esa función debe calcular la media de los tres números pasados como parámetros y mostrar con un mensaje cuál es la media calculada.</p>	<pre> #include&lt;stdio.h&gt; void media3(float n1,float n2,float n3);  int main() {     float n1, n2, n3;     printf("Introduzca un número real:\n");     scanf("%f",&amp;n1);     printf("Introduzca otro número real:\n");     scanf("%f",&amp;n2);     printf("Introduzca otro número real:\n");     scanf("%f",&amp;n3);      media3(n1,n2,n3);     return 0; }  void media3(float n1, float n2, float n3) {     float resultado;     resultado=(n1+n2+n3)/3;     printf("La media es:%f\n",resultado); } </pre>
<p>5. Realizar una función llamada negativo, que toma un número entero como parámetro, y devuelve 1 si es negativo o 0 si no lo es.</p>	<pre> #include&lt;stdio.h&gt; int negativo(int numero);  int main() {     int n,resultado;      printf("Introduzca un número:\n");     scanf("%i",&amp;n);      resultado=negativo(n);      if (resultado==1)         printf("Es un número negativo.\n");     else         printf("No es negativo.\n");      return 0; }  int negativo(int numero) {     int res;     if (numero&lt;0)         res=1;     else         res=0;     return res; } </pre>
<p>6. Realizar dos funciones: una llamada pedir, que no toma parámetros, y devuelve un</p>	<pre> #include&lt;stdio.h&gt;  int pedir(); </pre>

<p>número entero; y otra llamada triple, que toma un número entero como parámetro y devuelve un número entero.</p> <p>La función pedir debe pedir por teclado un número entero, y devolverlo.</p> <p>La función triple, debe calcular el triple del número que recibe como parámetro y devolver el resultado.</p>	<pre> int triple(int);  int main() {     int numero, total;      numero=pedir();      total =triple(numero);      printf("El triple de %i es: %i\n",numero, total);     return 0; }  int pedir() {     int n;     printf("Introduzca un número entero:\n");     scanf("%i",&amp;n);     return n; }  int triple(int num) {int n; n=3*num; return n; } </pre>
<p>7. Muestra el uso de funciones. Emplea cinco funciones además de main()</p> <p>main() llama a una de ellas y esta a otra, enviando un parámetro que se altera localmente y se envía a la siguiente función.</p> <p>Finaliza emitiendo el valor devuelto por la última función, en main();</p>	<pre> #include &lt;stdio.h&gt;  int *f1(int *a); int *f2(int *b); int *f3(int *c); int *f4(int *a); int *f5(int bx); //parametro pasado por valor  int main() {     int x = 33, *res;     printf("En main() x = %d\n\n", x);     printf("\nSaliendo de main:\n\n");     res = f1(&amp;x);     printf("\n\nVolviendo a main() res = %d", *res);     return 0; }  int * f1(int *n) {     int *res;     puts("\nEn f1():");     printf("\n\tAl entrar: %d", *n);     *n += *n;     res = f2(n);     return res; //devuelve una dirección de memoria }  int *f2(int *ppp) {     int *res;     puts("\nEn f2():");     printf("\n\tAl entrar: %d", *ppp);     *ppp += *ppp;     res = f3(ppp);     return res; //devuelve una dirección de memoria }  int * f3(int *asd) </pre>

	<pre> {     int *res;     puts("\nEn f3():");     printf("\n\tAl entrar: %d", *asd);     *asd += *asd;     res = f4(asd);     return res;    //devuelve una dirección de memoria } int * f4(int *valor) {     int *res;     puts("\nEn f4():");     printf("\n\tAl entrar: %d", *valor);     *valor += *valor;     res = f5(*valor);     return res;    //devuelve una dirección de memoria } int *f5(int ggg) //ggg no es puntero {     puts("\nEn f5():");     printf("\n\tAl entrar: %d", ggg);     ggg += ggg;     return &amp;ggg; //devuelve una dirección de memoria: la de ggg } </pre>
<p>8. En este ejemplo se usa un pasaje por valor que es un dato de entrada y un pasaje por dirección que funciona como dato de salida ya que la función es void.</p>	<pre> #include &lt;stdio.h&gt;  void promecal(int nota, int *notajus);  int main(){     int nota , notafinal;      printf("ingrese nota \n");     scanf ("%d", &amp;nota);     printf ("variable nota antes %d\n", nota);      promecal(nota,&amp;notafinal);     printf ("variable nota despues %d\n", nota);     printf ("Su nota ajustada es %d\n", notafinal);      return 0; }  void promecal(int nota, int *notajus) {     int notaajustada;     if (nota == 4)         notaajustada = nota + 1;     else if (nota &gt;= 8)         notaajustada = 10;     else notaajustada=nota;      *notajus=notaajustada; } </pre>
<p>9. Dadas la base y el exponente, calcular la potencia. Utilizar una función para el cálculo</p>	<pre> #include&lt;stdio.h&gt;  float calcpotencia(float, int);  int main() {     float base, pot;     int exp;     printf("Ingresa la base: "); </pre>

	<pre> scanf("%f", &amp;base); printf("Ingrese el exponente: "); scanf("%d", &amp;exp); pot = calcpotencia(base, exp); printf("La potencia es %.2f", pot); return 0; }  float calcpotencia(float base, int exp) {     int ciclo;     float pot;     pot = base;     if (exp == 0)         pot = 1;     else         for(ciclo=1;ciclo&lt;exp;ciclo++)             pot = pot * base;     return pot; } </pre>
10. Ejemplo de variable global	<pre> #include &lt;stdio.h&gt;  int vglobal = 1000;  int main(){ int vlocal=20; printf("\nLa variable vglobal en main vale = %d\n", vglobal); printf("\nLa variable vlocal en main vale = %d\n", vlocal); vlocal++; vglobal++; printf("\nLa variable vglobal despues de incrementar vale = %d\n", vglobal); printf("\nLa variable vlocal despues de incrementar vale = %d\n", vlocal); return 0; } </pre>