

EJERCICIOS RESUELTOS CON ESTRUCTURAS REPETITIVAS

ESTRUCTURA PARA (for)



Ing. Angel Moreano Terrazas

<https://www.facebook.com/amoreanoterrazas>



Creative Commons



Perú - 2013

Ejercicios Tipo Resueltos

1. **Mostrar los múltiplos de 8 hasta el valor 190. Debe aparecer en pantalla 8 - 16 - 24 - 32, etc. (no necesitas ingresar ningún valor desde el teclado)**

Análisis:

- Primero identifica en cuanto comienzan los números, comienza con 8 verdad.
- También debes fijarte el valor más alto que debe tomar en este caso no exceder de 190, por tanto **el valor máximo es de 184 que es 23x8, el valor de 23 lo obtenemos dividiendo 190 entre 8.**
- Ahora debemos darnos cuenta que necesitamos un bucle **PARA que va desde 1 hasta 23, al contador lo llamaremos k.**
- También debemos darnos cuenta que si tenemos el contador k podemos hallar lo que buscamos multiplicando por 8, es decir: **termino = k*8**, no olvides que k tiene valores que varían desde 1 hasta 23.

Diseño: Ahora podemos armar el diseño:

Algoritmo SerieDeOcho

Inicio

```
Entero k, termino
Para k = 1 Hasta 23
    termino = k*8
    Escribir(termino)
FinPara
```

Fin

Codificación en CSharp:

```
public static void Main(string[] args)
{
    int k, termino;
    for(k = 1; k <= 23; k=k+1)
    {
        termino = k*8;
        Console.WriteLine("♣" + termino);
    }
    Console.WriteLine("Enter para Salir . . .");
    Console.ReadLine();
}
```

2. Realizar un programa que imprima varios términos de la serie 11 - 22 - 33 - 44, etc. (Se ingresan sólo en valor de n para indicar la cantidad de términos)

Análisis:

- Como vemos, si observas bien los numeros, son multiplicados por 11, por ejemplo 1x11, 2x11, 3x11, 4x11, ... 4 x n, así obtenemos los términos.
- También necesitamos un bucle de tipo PARA, que inicie en 1 hasta n, también debes tener en cuenta que n es cualquier valor leído.
- Recuerda que lo que se repite es: dato = k * 11, por supuesto que k es el contador que cambia de valores desde 1 hasta n.

Diseño: Ahora podemos armar el diseño:

Algoritmo LaSerieOnce

Inicio

Entero k, dato, n

Leer(n)

Para k = 1 Hasta n

dato = k*11

Escribir(dato)

FinPara

Fin

Nota: Cuando en un ejercicio te dicen que imprimas, leas o hagas una operación varias veces, significa que debes leer un valor de n que te indica la cantidad de veces que haras la operación, luego en el bucle PARA deberás poner n como límite de las repeticiones.

Codificación en CSharp:

```
public static void Main(string[] args)
{
    int k, dato, n;
    Console.WriteLine("♣♣♣ Ingresa el valor de N ♣♣♣");
    n = int.Parse(Console.ReadLine());
    for(k = 1; k <= n; k = k + 1)
    {
        dato = k * 11;
        Console.WriteLine("Elemento de la serie:" + dato);
    }
    Console.WriteLine("Enter para Salir . . . ");
    Console.ReadLine();
}
```

3. Se tienen la base y altura de 10 triángulos, se desea calcular el área de cada uno de ellos.

Análisis:

- Ahora veamos, si nos dicen 10 triángulos entonces usamos un bucle PARA que servirá para repetir el calculo de area desde 1 hasta 10, usando el contador j.
- Entonces también sabemos que para calcular el área de un triángulo tenemos la formula: $\text{area} = \text{base} * \text{altura} / 2$, dicha fórmula deberá repetirse, por supuesto ingresando como datos la base y la altura, es decir, leer base y altura antes de calcular el área.
- Como respuesta tenemos que escribir el área calculada.

Diseño: Ahora podemos armar el diseño:

Algoritmo AreaDeDiezTriangulos

Inicio

```

Entero j
Real area, basee, altura
Para j = 1 Hasta 10
    Leer(basee, altura)
    area = basee * altura / 2
    Escribir(area)
FinPara

```

Fin

Nota: puedes cambiar el valor de 10 por una variable n por ejemplo y leer n para que puedas calcular la cantidad de triángulos que quieras.

Codificación en CSharp:

[illegible]

4. Escribir un programa que solicite ingresar 8 notas de alumnos y nos informe cuántos tienen notas mayores o iguales a 13 y cuántos menores a esta nota.

Análisis:

- Primero deberás tener en cuenta que si son 8 notas para ingresar deberás tener un bucle PARA que permita repetir 8 veces, k va desde 1 hasta 8.
- Ahora dentro del bucle deberás leer las notas, de forma que en cada lectura se verifique que la nota es mayor o igual a 13, si esto ocurre contamos los mayores.
- En caso que la nota no cumpla la condición contamos los menores.
- Al final debemos mostrar como resultado la cantidad de notas mayores y menores.

Diseño: Ahora podemos armar el diseño:

Algoritmo OchoNotas

Inicio

```
Entero nota, mayores, menores, k
mayores = 0
menores = 0
Para k = 1 Hasta 8
    Leer(nota)
    Si nota >= 13 Entonces
        mayores = mayores + 1
    Sino
        menores = menores + 1
    FinSi
FinPara
Escribir(mayores, menores)
```

Fin

Nota: Tenemos dos contadores, mayores y también menores, estos contadores son inicializados en CERO, luego se actualizan en 1 para poder realizar en conteo.

Codificación en CSharp:

```
public static void Main(string[] args)
{
    int nota, mayores, menores, k;
    mayores = 0;
    menores = 0;
    for(k = 1; k <= 8; k=k+1)
    {
        Console.WriteLine("Ingresa la nota "+k+ " de 8: ");
        nota = int.Parse(Console.ReadLine());
        if (nota >= 13)
            mayores = mayores + 1;
        else
            menores = menores + 1;
    }
    Console.WriteLine("■■■■■■ INFORME ■■■■■■");
    Console.WriteLine("La notas de 13 a más son: "+mayores);
    Console.WriteLine("La notas menores a 13 son: "+menores);
    Console.WriteLine("Enter para Salir . . .");
    Console.ReadLine();
}
```

5. Desarrollar un programa que permita cargar n números enteros y luego nos informe cuántos valores fueron pares y cuántos impares.

Análisis:

- Para cargar n números, primero debemos leer n, luego deberemos hacer un bucle PARA que vaya desde 1 hasta n, para poder leer los números.
- Ahora dentro del bucle para deberás preguntar si el número es par o impar.
- Cuando los números son pares deberás usar un contador llamado a para contar los pares, es decir: $a = a + 1$.
- Cuando los números son impares, deberás usar un contador llamado b para contar los impares, es decir: $b = b + 1$.
- Finalmente fuera del bucle deberás escribir las respuestas a y b.
- Importante, no olvidar que a y b deben ser inicializadas en cero antes del bucle PARA.

Diseño: Ahora podemos armar el diseño:

Algoritmo CuentaParesImpares

Inicio

Entero a, b, dato, k, n

a = 0

b = 0

Leer (n)

Para k = 1 Hasta n

Leer(dato)

Si dato MOD 2 == 0 Entonces

a = a + 1

Sino

b = b + 1

FinSi

FinPara

Escribir(a, b)

Fin

Nota: Aquí usas dos contadores a y b; a, es para contar pares. b, es para contar impares, dato son los números que ingresas para el conteo.

Codificación en CSharp:

```
public static void Main(string[] args)
{
    int a, b, dato, k, n;
    a = 0;
    b = 0;
    Console.WriteLine("PROGRAMA PARA CONTAR PARES E IMPARES");
    Console.Write("Ingresa el valor de N: ");
    n = int.Parse(Console.ReadLine());
    for(k = 1; k <= n; k=k+1)
    {
        Console.Write("Ingresa un numero: ");
        dato = int.Parse(Console.ReadLine());
        if (dato % 2 == 0)
            a = a + 1;
        else
            b = b + 1;
    }
    Console.WriteLine("INFORME");
    Console.WriteLine("La cantidad de pares ingresada es: " + a);
    Console.WriteLine("La cantidad de impares ingresada es: " + b);
}
```

```

    Console.WriteLine("Enter para Salir . . .");
    Console.ReadLine();
}

```

6. Hacer un programa que calcule el promedio de varias notas (n).

Análisis:

- Los primero que debemos hacer es tener la cantidad de notas que vamos a calcular, para ello leeremos n.
- El segundo paso es hacer la suma de las notas que deberemos ingresar, para ellos usamos el bucle PARA desde 1 hasta n, en el interior del bucle debemos leer cada nota y comenzar a sumar, es decir, acumular las notas.
- Después del bucle al terminar de ingresar todas las notas debemos proceder a calcular el promedio, para ello la suma deberás dividirla entre la cantidad de notas, cola la fórmula: $\text{promedio} = \text{suma} / n$.
- Finalmente deberás mostrar como resultado el promedio.

Diseño: Ahora podemos armar el diseño:

Algoritmo PromedioNotas

Inicio

Real nota, suma, promedio, k, n

Leer(n)

suma = 0

Para k = 1 Hasta n

Leer(nota)

suma = suma + nota

FinPara

promedio = suma / n

Escribir(promedio)

Fin

Nota: la variable **suma es un acumulador**, por regla sabes que los acumuladores se inician en cero antes del bucle y luego mediante un bucle comenzamos a acumular los datos, a esto se le llama actualizar el acumulador.

Codificación en CSharp:

```

public static void Main(string[] args)
{
    double nota, suma, promedio, k, n;
    Console.WriteLine("PROMEDIO DE N NOTAS");
    Console.Write("Ingresa la cantidad de notas:");
}

```

```
n = double.Parse(Console.ReadLine());
suma = 0;
for(k = 1; k<= n; k=k+1)
{
    Console.WriteLine("Ingresa nota" + k + ": ");
    nota = double.Parse(Console.ReadLine());
    suma = suma + nota;
}
promedio = suma / n;
Console.WriteLine("El promedio de las notas es:" + promedio);
Console.WriteLine("Enter para Salir . . . ");
Console.ReadLine();
}
```


7. Hacer un programa que ingrese varios números positivos y nos indique el mayor de estos números. (recuerda al hablar de varios números podemos asumir que se trata de n números)

Análisis:

- Lo primero que debemos hacer es saber cuántos números ingresaremos, para ellos leeremos n primero.
- Después de saber el valor de n, ahora procedemos formar un bucle PARA que nos permitirá ingresar la cantidad de números que deseamos desde 1 hasta n, para esto usaremos un contador llamado k.
- Ahora dentro del bucle debemos leer el número y comenzar a capturar el mayor de estos, la variable a leer le llamaremos dato.
- Para hacer la captura tendremos que tener una variable llamada mayor, la cual inicializamos en cero antes de comenzar el bucle, para poder así comparar con los números positivos ingresado ingresados.

Diseño: Ahora podemos armar el diseño:

Algoritmo NumeroMayor

Inicio

```
Entero n, k, mayor, dato
mayor = 0
Leer(n)
Para k=1 Hasta n
    Leer(dato)
    Si dato > mayor Entonces
        mayor = dato
    FinSi
FinPara
Escribir(mayor)
```

Fin

Codificación en CSharp:

```
public static void Main(string[] args)
{
    int n, k, mayor, dato;
    mayor = 0;
    Console.Write("Ingresa la cantidad de numeros: ");
    n = int.Parse(Console.ReadLine());
    for(k=1; k <= n ;k=k+1)
    {
        Console.Write(" ♣♣ " + k + ". Ingresa numero: ");
        dato = int.Parse(Console.ReadLine());
        if (dato > mayor)
        {
            mayor = dato;
        }
    }
    Console.WriteLine("El mayor de todos los numeros es: " + mayor);
    Console.WriteLine("Enter para Salir . . . ");
    Console.ReadLine();
}
```

8. Los autos de la ciudad tienen placas que son solo números, el último dígito del número indica el color del auto según los siguientes valores:

DÍGITO	COLOR
1 o 2	amarillo
3 o 4	negro
5 o 6	rojo
7 o 8	verde
9 o 0	azul

El programa debe pedir la placa de cierta cantidad de autos (quiere decir n autos), y en cada caso debe mostrar el color de cada auto ingresado.

Análisis:

- Lo primero es saber la cantidad de autos que se van a ingresar, en este caso serán n, para ello debemos leer n primero.
- Ahora debemos hacer un bucle PARA que repita desde 1 hasta n.
- Las operaciones que repetimos dentro del bucle son: La lectura de la placa, luego evaluamos con varias estructuras si el último dígito del número y en estos mostrar el mensaje que corresponde al color.
- Vamos a tener en cuenta que usaremos una variable llamada digito que tendrá el valor del último dígito de la placa con la fórmula: $\text{digito} = \text{placa} \text{ MOD } 10$.

Diseño: Ahora podemos armar el diseño:

Algoritmo ColorDeAuto

Inicio

```
    Entero n, digito, placa, k
    Leer(n)
    Para k = 1 hasta n
        Leer(placa)
        digito = placa MOD 10
        Si digito == 1 ó digito == 2 Entonces
            Escribir("amarillo")
        FinSi
        Si digito == 3 ó digito == 4 Entonces
            Escribir("negro")
        FinSi
        Si digito == 5 ó digito == 6 Entonces
            Escribir("rojo")
        FinSi
        Si digito == 7 ó digito == 8 Entonces
            Escribir("verde")
        FinSi
        Si digito == 9 ó digito == 0 Entonces
            Escribir("azul")
        FinSi
    FinPara
```

Fin

Codificación en CSharp:

```
public static void Main(string[] args)
```

[illegible]

9. Se tienen 10 personas, de las cuales se conocen sus edades, hacer un programa que permita ingresar las edades de todas las personas y luego nos indique cuantas son mayores de edad y cuántos son menores de edad (Consideramos mayores de edad a todos aquellos que tienen de 18 años a más).

Análisis:

- Como vemos ya tenemos la cantidad exacta de personas, por ello entonces solo debemos hacer un bucle PARA que permita ir desde 1 hasta 10.
- Dentro del bucle debemos leer la edad de cada persona, luego debemos evaluar si son mayores o menores, si es mayor debemos actualizar un contador que llamaremos may, en caso contrario si es menor actualizamos un contador llamado men.

Diseño: Ahora podemos armar el diseño:

Algoritmo EdadesMayoresMenores

Inicio

Entero edad, may, men, k

```

may = 0
men = 0
Para k = 1 hasta 10
    Leer(edad)
    Si edad >= 18 Entonces
        may = may + 1
    Sino
        men = men + 1
    FinSi
FinPara
Escribir(may, men)

```

Fin

Nota: tenemos dos contadores uno llamado may, el otro se llama men, los contadores se suman siempre en uno y se inicializan en cero.

Codificación en CSharp:

```

public static void Main(string[] args)
{
    int edad, may, men, k;
    may = 0;
    men = 0;
    Console.WriteLine("***** PROGRAMA DE EDADES *****");
    for(k = 1; k <= 10; k=k+1)
    {
        Console.Write("Persona Nº " + k + " de 10. Ingrese su edad: ");
        edad = int.Parse(Console.ReadLine());
        if (edad >= 18)
            may = may + 1;
        else
            men = men + 1;
    }
    Console.WriteLine("\n***** RESULTADOS *****");
    Console.WriteLine("La cantidad de mayores de edad es: " + may);
    Console.WriteLine("La cantidad de menores de edad es: " + men);
    Console.WriteLine("\nEnter para Salir . . .");
    Console.ReadLine();
}

```

10. Se tienen tres modelos de celulares, el modelo 1, 2, 3; el modelo 1 cuesta 60um, el modelo 2 cuesta 120um, el modelo 3 cuesta 200um, el programa debe ingresar la cantidad de celulares que se han vendido, luego debe contar cuantos celulares de del modelo 1, 2, 3 se han vendido, también debe mostrar el total en dinero que se obtuvo de de todas las ventas.

EJERCICIO DE LA TIENDA DE CELULARES

Diseño: Ahora podemos armar el diseño:

Algoritmo LosCelulares

Inicio

Entero n, k, cel1, cel2, cel3, suma, tipo

cel1 = 0

cel2 = 0

cel3 = 0

suma = 0

Leer(n)

Para k = 1 Hasta n

Leer(tipo)

Si tipo = 1 Entonces

cel1 = cel1+1

suma = suma + 60

FinSi

Si tipo = 2 Entonces

cel2 = cel2+1

suma = suma + 120

FinSi

Si tipo = 3 Entonces

cel3 = cel3+1

suma = suma + 200

FinSi

FinPara

Escribir(cel1, cel2, cel3, suma)

Fin

Nota:

tipo. es el tipo de celular, puede ser 1, 2 ó 3.

cel1, cel2, cel3. son los contadores para cada tipo de celular

suma. es el total que se paga por todos los celulares

n. es la cantidad de celulares

/*SOLUCIÓN EN CSHARP*/

public static void Main(string[] args)

{

int n, k, cel1, cel2, cel3, suma, tipo;

cel1 = 0;

cel2 = 0;

cel3 = 0;

suma = 0;

Console.**WriteLine**("Ingresa la cantidad de celulares");

```

n = int.Parse(Console.ReadLine());
Console.WriteLine();
Console.WriteLine("♠♠♠♠ INGRESO DE CELULARES ♠♠♠♠");
for(k = 1; k <= n; k=k+1)
{
    Console.Write("Ingresa el tipo de celular: ");
    tipo = int.Parse(Console.ReadLine());
    if (tipo == 1)
    {
        cel1 = cel1+1;
        suma = suma + 60;
    }
    if (tipo == 2)
    {
        cel2 = cel2+1;
        suma = suma + 120;
    }
    if(tipo == 3)
    {
        cel3 = cel3+1;
        suma = suma + 200;
    }
}
Console.WriteLine();
Console.WriteLine("♠♠♠♠ RESULTADOS ♠♠♠♠");
Console.WriteLine("La cantidad de celulares del tipo UNO es: "+cel1);
Console.WriteLine("La cantidad de celulares del tipo DOS es: "+cel2);
Console.WriteLine("La cantidad de celulares del tipo TRES es: "+cel3);
Console.WriteLine("El monto por las ventas es: "+suma);
Console.WriteLine("Presiona Enter para salir");
Console.ReadLine();
}

```

11. Ayuda con el programa para encriptar mensajes (esto no tiene que ver nada con el examen, si quieres que modificar la codificación deberás cambiar tanto en el codificador como en el decodificador el 3, así obtendrás diferentes tipos de codificación, te recomiendo usar valores desde el 5 hasta el 80, ver la tabla ASCII es muy importante ver los valores y entender como funciona esto).

Codificador de mensajes:

- Este programa permite que un mensaje sea convertido en otros símbolos, todo tiene que ver con la tabla de códigos ASCII, para codificar solo sumamos 3, observa una tabla ASCII y analiza lo que has escrito y lo que te sale.

- Como usarlo: Escribes un texto y luego presionas ENTER y el texto aparecerá codificado, no podrás entender lo que dice.

En CSharp:

```
public static void Main(string[] args)
{
    Console.WriteLine("♠♠♠♠♠♠ ENCRIPADOR DE MENSAJES ♠♠♠♠♠♠");
    Console.WriteLine("Escribe el mensaje el a codificar solo minuscula: ");
    string msg = Console.ReadLine();
    char[] msgCod = new char[msg.Length];

    //Codificador
    for (int i = 0; i < msg.Length; i++)
    {
        int asciiCod = msg[i] + 3;
        msgCod[i] = (char)asciiCod;
    }
    Console.WriteLine("♣♣♣♣ Mensaje Codificado: ");
    //Imprimo
    for (int i = 0; i < msgCod.Length; i++)
    {
        Console.Write(msgCod[i]);
    }
    Console.WriteLine();
    Console.WriteLine("Presiona enter para Salir.....");
    Console.ReadLine();
}
```

Decodificador de mensajes:

- Este programa permite que un mensaje codificado se muestre como realmente es, todo tiene que ver con la tabla de codigos ascii como anteriormente dije, para decodificar restamos 3 ya que anteriormente habiamos sumado tres, observa una tabla ascii y analiza ya que todo tiene que ver con los valores de esta tabla, el desplazamiento que hacemos es 3.
- Deberás escribir el texto codificado, ese texto inentendible y luego ENTER, ahí podrás ver lo que significa.

En CSharp:

```
public static void Main(string[] args)
{
    Console.WriteLine("♥♥♥♥♥♥ DESENCRIPTADOR DE MENSAJES ♥♥♥♥♥♥");
    Console.WriteLine("Escribe el mensaje a decodificar: ");
    string msg = Console.ReadLine();
    char[] msgCod = new char[msg.Length];

    //Codificador
    for (int i = 0; i < msg.Length; i++)
    {
        int asciiCod = msg[i] - 3;
    }
}
```

```
        msgCod[i] = (char)asciiCod;
    }
    Console.WriteLine("♣♣♣♣ Mensaje Codificado: ");
    //Imprimo
    for (int i = 0; i < msgCod.Length; i++)
    {
        Console.Write(msgCod[i]);
    }
    Console.WriteLine();
    Console.WriteLine("Presiona enter para Salir.....");
    Console.ReadLine();
}
```


Ejercicios Tipo Propuestos

(De las 6 primeras preguntas de los propuestos vienen dos preguntas para el examen, recomiendo que resuelvas las 6 primeras preguntas y así estar listo para tu examen, revisa los programas resueltos, ahí esta la solución, estas son solo variantes)

12. Leer n números y mostrar estos números al cubo.
13. Hacer un algoritmo que lea n notas y nos indique cuantas con mayores a cero y cuantas son menores a cero, también cuántos son iguales a cero.
14. Calcular el promedio de un alumno que tiene 7 calificaciones en la materia de programación.
15. Realizar un programa que permita cargar dos listas de 5 valores cada una. Informar con un mensaje cuál de las dos listas tiene un valor acumulado mayor o igual (los mensajes son "Lista 1 mayor", "Lista 2 mayor", "Listas iguales"). Tener en cuenta que puede haber dos o más estructuras repetitivas en un algoritmo, usar dos acumuladores una para cada lista de números.
16. Leer n números y determinar el número mayor y el número menor.
17. Tiendas ACME desea un programa que se ingrese en monto de varias ventas, al ingresar cada una de las ventas debe mostrarse un descuento del 7% y el monto que se paga en dicha venta (al decir varias ventas nos referimos a n ventas).
18. Hacer un programa que lea el sexo de n varones y mujeres, los varones tienen el valor de 1 y la mujeres el valor de 2, el programa debe mostrar el porcentaje de varones y el porcentaje de mujeres.
19. Hacer un programa que lea los nombres y las edades de varias personas (n), al finalizar el programa debe mostrar el nombre y la edad de la persona mayor, también el nombre y la edad de la persona menor.
20. Una persona desea invertir su dinero en un banco, el cual le otorga un 2% de interés mensual. Cual será la cantidad de dinero que esta persona tendrá al cabo de un año si la ganancia de cada mes es reinvertida (interés acumulado)
21. Una pizzería tiene n ventas, vende pizzas de 30cm, 40cm y 50cm de diametro. Hacer un programa que indique la cantidad de pizzas por cada tamaño, también indique el total vendido de las pizzas en soles, sabiendo que una pizza de 30cm cuesta 15 um, una de 40cm cuesta 30 um, y una de 50cm cuesta 40 um.
22. Se ingresan un conjunto de n alturas de personas por teclado. Mostrar la altura promedio de las personas.

Plantilla de Solución:

Análisis: <ul style="list-style-type: none">•
Diseño: Ahora podemos armar el diseño:
Codificación en CSharp: