

5.4 Ejercicios



Ejercicio 1

Muestra los números múltiplos de 5 de 0 a 100 utilizando un bucle `for`.



Ejercicio 2

Muestra los números múltiplos de 5 de 0 a 100 utilizando un bucle `while`.



Ejercicio 3

Muestra los números múltiplos de 5 de 0 a 100 utilizando un bucle `do-while`.



Ejercicio 4

Muestra los números del 320 al 160, contando de 20 en 20 hacia atrás utilizando un bucle `for`.



Ejercicio 5

Muestra los números del 320 al 160, contando de 20 en 20 hacia atrás utilizando un bucle `while`.



Ejercicio 6

Muestra los números del 320 al 160, contando de 20 en 20 utilizando un bucle `do-while`.



Ejercicio 7

Realiza el control de acceso a una caja fuerte. La combinación será un número de 4 cifras. El programa nos pedirá la combinación para abrirla. Si no acertamos, se nos mostrará el mensaje “Lo siento, esa no es la combinación” y si acertamos se nos dirá “La caja fuerte se ha abierto satisfactoriamente”. Tendremos cuatro oportunidades para abrir la caja fuerte.



Ejercicio 8

Muestra la tabla de multiplicar de un número introducido por teclado.



Ejercicio 9

Realiza un programa que nos diga cuántos dígitos tiene un número introducido por teclado. Este ejercicio es equivalente a otro realizado anteriormente, con la salvedad de que el anterior estaba limitado a números de 5 dígitos como máximo. En esta ocasión, hay que realizar el ejercicio utilizando bucles; de esta manera, la única limitación en el número de dígitos la establece el tipo de dato que se utilice (`int` o `long`).



Ejercicio 10

Escribe un programa que calcule la media de un conjunto de números positivos introducidos por teclado. A priori, el programa no sabe cuántos números se introducirán. El usuario indicará que ha terminado de introducir los datos cuando meta un número negativo.



Ejercicio 11

Escribe un programa que muestre en tres columnas, el cuadrado y el cubo de los 5 primeros números enteros a partir de uno que se introduce por teclado.



Ejercicio 12

Escribe un programa que muestre los n primeros términos de la serie de Fibonacci. El primer término de la serie de Fibonacci es 0, el segundo es 1 y el resto se calcula sumando los dos anteriores, por lo que tendríamos que los términos son 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144... El número n se debe introducir por teclado.



Ejercicio 13

Escribe un programa que lea una lista de diez números y determine cuántos son positivos, y cuántos son negativos.



Ejercicio 14

Escribe un programa que pida una base y un exponente (entero positivo) y que calcule la potencia.

**Ejercicio 15**

Escribe un programa que dados dos números, uno real (base) y un entero positivo (exponente), saque por pantalla todas las potencias con base el número dado y exponentes entre uno y el exponente introducido. No se deben utilizar funciones de exponenciación. Por ejemplo, si introducimos el 2 y el 5, se deberán mostrar 2^1 , 2^2 , 2^3 , 2^4 y 2^5 .

**Ejercicio 16**

Escribe un programa que diga si un número introducido por teclado es o no primo. Un número primo es aquel que sólo es divisible entre él mismo y la unidad.

**Ejercicio 17**

Realiza un programa que sume los 100 números siguientes a un número entero y positivo introducido por teclado. Se debe comprobar que el dato introducido es correcto (que es un número positivo).

**Ejercicio 18**

Escribe un programa que obtenga los números enteros comprendidos entre dos números introducidos por teclado y validados como distintos, el programa debe empezar por el menor de los enteros introducidos e ir incrementando de 7 en 7.

**Ejercicio 19**

Realiza un programa que pinte una pirámide por pantalla. La altura se debe pedir por teclado. El carácter con el que se pinta la pirámide también se debe pedir por teclado.

**Ejercicio 20**

Igual que el ejercicio anterior pero esta vez se debe pintar una pirámide hueca.

**Ejercicio 21**

Realiza un programa que vaya pidiendo números hasta que se introduzca un número negativo y nos diga cuantos números se han introducido, la media de los impares y el mayor de los pares. El número negativo sólo se utiliza para indicar el final de la introducción de datos pero no se incluye en el cómputo.

**Ejercicio 22**

Muestra por pantalla todos los números primos entre 2 y 100, ambos incluidos.

**Ejercicio 23**

Escribe un programa que permita ir introduciendo una serie indeterminada de números mientras su suma no supere el valor 10000. Cuando esto último ocurra, se debe mostrar el total acumulado, el contador de los números introducidos y la media.

**Ejercicio 24**

Escribe un programa que lea un número n e imprima una pirámide de números con n filas como en la siguiente figura:

```
1
121
12321
1234321
```

**Ejercicio 25**

Realiza un programa que pida un número por teclado y que luego muestre ese número al revés.

**Ejercicio 26**

Realiza un programa que pida primero un número y a continuación un dígito. El programa nos debe dar la posición (o posiciones) contando de izquierda a derecha que ocupa ese dígito en el número introducido.

**Ejercicio 27**

Escribe un programa que muestre, cuente y sume los múltiplos de 3 que hay entre 1 y un número leído por teclado.



Ejercicio 28

Escribe un programa que calcule el factorial de un número entero leído por teclado.

Ejemplo:

```
Por favor, introduzca un número entero: 6
6! = 720
```



Ejercicio 29

Escribe un programa que muestre por pantalla todos los números enteros positivos menores a uno leído por teclado que no sean divisibles entre otro también leído de igual forma.



Ejercicio 30

Realiza una programa que calcule las horas transcurridas entre dos horas de dos días de la semana. No se tendrán en cuenta los minutos ni los segundos. El día de la semana se puede pedir como un número (del 1 al 7) o como una cadena (de "lunes" a "domingo"). Se debe comprobar que el usuario introduce los datos correctamente y que el segundo día es posterior al primero.

Ejemplo:

```
Por favor, introduzca la primera hora.
Día: lunes
Hora: 18
Por favor, introduzca la segunda hora.
Día: martes
Hora: 20
Entre las 18:00h del lunes y las 20:00h del martes hay 26 hora/s.
```



Ejercicio 31

Realiza un programa que pinte la letra **L** por pantalla hecha con asteriscos. El programa pedirá la altura. El palo horizontal de la **L** tendrá una longitud de la mitad (división entera entre 2) de la altura más uno.

Ejemplo:

```
Introduzca la altura de la L: 5
```

```
*
*
*
*
* * *
```



Ejercicio 32

Escribe un programa que, dado un número entero positivo, diga cuáles son y cuánto suman los dígitos pares. Los dígitos pares se deben mostrar en orden, de izquierda a derecha. Usa `long` en lugar de `int` donde sea necesario para admitir números largos.

Ejemplo 1:

```
Por favor, introduzca un número entero positivo: 94026782
Dígitos pares: 4 0 2 6 8 2
Suma de los dígitos pares: 22
```

Ejemplo 2:

```
Por favor, introduzca un número entero positivo: 31779
Dígitos pares:
Suma de los dígitos pares: 0
```

Ejemplo 3:

```
Por favor, introduzca un número entero positivo: 2404
Dígitos pares: 2 4 0 4
Suma de los dígitos pares: 10
```



Ejercicio 33

Realiza un programa que pinte la letra **U** por pantalla hecha con asteriscos. El programa pedirá la altura. Fíjate que el programa inserta un espacio y pinta dos asteriscos menos en la base para simular la curvatura de las esquinas inferiores.

Ejemplo 1:

Introduzca la altura de la U: 5

```
*      *  
*      *  
*      *  
*      *  
*      *  
  * * *
```

Ejemplo 2:

Introduzca la altura de la U: 4

```
*      *  
*      *  
*      *  
  * *
```



Ejercicio 34

Escribe un programa que pida dos números por teclado y que luego mezcle en dos números diferentes los dígitos pares y los impares. Se van comprobando los dígitos de la siguiente manera: primer dígito del primer número, primer dígito del segundo número, segundo dígito del primer número, segundo dígito del segundo número, tercer dígito del primer número... Para facilitar el ejercicio, podemos suponer que el usuario introducirá dos números de la misma longitud y que siempre habrá al menos un dígito par y uno impar. Usa `long` en lugar de `int` donde sea necesario para admitir números largos.

Ejemplo 1:

```
Por favor, introduzca un número: 9402
Introduzca otro número: 6782
El número formado por los dígitos pares es 640822
El número formado por los dígitos impares es 97
```

Ejemplo 2:

```
Por favor, introduzca un número: 137
Introduzca otro número: 909
El número formado por los dígitos pares es 0
El número formado por los dígitos impares es 19379
```



Ejercicio 35

Realiza un programa que pinte una X hecha de asteriscos. El programa debe pedir la altura. Se debe comprobar que la altura sea un número impar mayor o igual a 3, en caso contrario se debe mostrar un mensaje de error.

Ejemplo:

```
Por favor, introduzca la altura de la X: 5

*  *
* *
 *
* *
*  *
```




Ejercicio 36

Escribe un programa que diga si un número introducido por teclado es o no capicúa. Los números capicúa se leen igual hacia delante y hacia atrás. El programa debe aceptar números de cualquier longitud siempre que lo permita el tipo, en caso contrario el ejercicio no se dará por bueno. Se recomienda usar `long` en lugar de `int` ya que el primero admite números más largos.

Ejemplo 1:

```
Por favor, introduzca un número entero positivo: 678
El 678 no es capicúa.
```

Ejemplo 2:

```
Por favor, introduzca un número entero positivo: 2019102
El 2019102 es capicúa.
```



Ejercicio 37

Realiza un conversor del sistema decimal al sistema de “palotes”.

Ejemplo:

```
Por favor, introduzca un número entero positivo: 47021
El 47021 en decimal es el | | | | - | | | | | - - | | - | en el sistema de palotes.
```



Ejercicio 38

Realiza un programa que pinte un reloj de arena relleno hecho de asteriscos. El programa debe pedir la altura. Se debe comprobar que la altura sea un número impar mayor o igual a 3, en caso contrario se debe mostrar un mensaje de error.

Ejemplo:

```
Por favor, introduzca la altura del reloj de arena: 5
```

```
*****
***
*
***
*****
```



Ejercicio 39

Escribe un programa que pida un número entero positivo por teclado y que muestre a continuación los números desde el 1 al número introducido junto con su factorial.

Ejemplo:

```
Por favor, introduzca un número entero positivo: 7
```

```
1! = 1
2! = 2
3! = 6
4! = 24
5! = 120
6! = 720
7! = 5040
```



Ejercicio 40

Realiza un programa que pinte por pantalla un rombo hueco hecho con asteriscos. El programa debe pedir la altura. Se debe comprobar que la altura sea un número impar mayor o igual a 3, en caso contrario se debe mostrar un mensaje de error.

Ejemplo:

```
Por favor, introduzca la altura del rombo: 5
```

```
  *
 * *
*   *
 * *
  *
```



Ejercicio 41

Escribe un programa que diga cuántos dígitos pares y cuántos dígitos impares hay dentro de un número. Se recomienda usar `long` en lugar de `int` ya que el primero admite números más largos.

Ejemplo 1:

```
Por favor, introduzca un número entero positivo: 406783
El 406783 contiene 4 dígitos pares y 2 dígitos impares.
```

Ejemplo 2:

```
Por favor, introduzca un número entero positivo: 3177840
El 3177840 contiene 3 dígitos pares y 4 dígitos impares.
```



Ejercicio 42

Escribe un programa que pida un número entero positivo por teclado y que muestre a continuación los 5 números consecutivos a partir del número introducido. Al lado de cada número se debe indicar si se trata de un primo o no.

Ejemplo:

```
Por favor, introduzca un número entero positivo: 17
17 es primo
18 no es primo
19 es primo
20 no es primo
21 no es primo
```



Ejercicio 43

Escribe un programa que permita partir un número introducido por teclado en dos partes. Las posiciones se cuentan de izquierda a derecha empezando por el 1. Suponemos que el usuario introduce correctamente los datos, es decir, el número introducido tiene dos dígitos como mínimo y la posición en la que se parte el número está entre 2 y la longitud del número. No se permite en este ejercicio el uso de funciones de manejo de `String` (por ej. para extraer subcadenas dentro de una cadena).

Ejemplo:

```
Por favor, introduzca un número entero positivo: 406783
Introduzca la posición a partir de la cual quiere partir el número: 5
Los números partidos son el 4067 y el 83.
```



Ejercicio 44

Escribe un programa que sea capaz de insertar un dígito dentro de un número indicando la posición. El nuevo dígito se colocará en la posición indicada y el resto de dígitos se desplazará hacia la derecha. Las posiciones se cuentan de izquierda a derecha empezando por el 1. Suponemos que el usuario introduce correctamente los datos. Se recomienda usar `long` en lugar de `int` ya que el primero admite números más largos.

Ejemplo:

```
Por favor, introduzca un número entero positivo: 406783
Introduzca la posición donde quiere insertar: 3
Introduzca el dígito que quiere insertar: 5
El número resultante es 4056783.
```



Ejercicio 45

Escribe un programa que cambie un dígito dentro de un número dando la posición y el valor nuevo. Las posiciones se cuentan de izquierda a derecha empezando por el 1. Se recomienda usar `long` en lugar de `int` ya que el primero admite números más largos. Suponemos que el usuario introduce correctamente los datos.

Ejemplo:

```
Por favor, introduzca un número entero positivo: 406783
Introduzca la posición dentro del número: 3
Introduzca el nuevo dígito: 1
El número resultante es 401783
```



Ejercicio 46

Realiza un programa que pinte por pantalla un rectángulo hueco hecho con asteriscos. Se debe pedir al usuario la anchura y la altura. Hay que comprobar que tanto la anchura como la altura sean mayores o iguales que 2, en caso contrario se debe mostrar un mensaje de error.

Ejemplo 1:

```
Por favor, introduzca la anchura del rectángulo (como mínimo 2): 4
Ahora introduzca la altura (como mínimo 2): 1
Lo siento, los datos introducidos no son correctos, el valor mínimo para la anchura y la altura es 2.
```

Ejemplo 2:

```
Por favor, introduzca la anchura del rectángulo (como mínimo 2): 6
Ahora introduzca la altura (como mínimo 2): 4
```

```
* * * * *
*       *
*       *
* * * * *
```



Ejercicio 47

Con motivo de la celebración del día de la mujer, el 8 de marzo, nos han encargado realizar un programa que pinte un 8 por pantalla usando la letra M. Se pide al usuario la altura, que debe ser un número entero impar mayor o igual que 5. Si el número introducido no es correcto, el programa deberá mostrar un mensaje de error. A continuación se muestran algunos ejemplos. La anchura de la figura siempre será de 6 caracteres.

Ejemplo 1:

```
Por favor, introduzca la altura (número impar mayor o igual a 5): 8
La altura introducida no es correcta
```

Ejemplo 2:

```
Por favor, introduzca la altura (número impar mayor o igual a 5): 3
La altura introducida no es correcta
```

Ejemplo 3:

```
Por favor, introduzca la altura (número impar mayor o igual a 5): 5
MMMMMM
M      M
MMMMMM
M      M
MMMMMM
```

Ejemplo 4:

```
Por favor, introduzca la altura (número impar mayor o igual a 5): 9
MMMMMM
M      M
M      M
M      M
MMMMMM
M      M
M      M
M      M
MMMMMM
```



Ejercicio 48

Realiza un programa que diga los dígitos que aparecen y los que no aparecen en un número entero introducido por teclado. El orden es el que se muestra en los ejemplos. Utiliza el tipo long para que el usuario pueda introducir números largos.

Ejemplo 1:

```
Introduzca un número entero: 67706
Dígitos que aparecen en el número: 0 6 7
Dígitos que no aparecen: 1 2 3 4 5 8 9
```

Ejemplo 2:

```
Introduzca un número entero: 555
Dígitos que aparecen en el número: 5
Dígitos que no aparecen: 1 2 3 4 6 7 8 9
```

Ejemplo 3:

```
Introduzca un número entero: 9876543210
Dígitos que aparecen en el número: 0 1 2 3 4 5 6 7 8 9
Dígitos que no aparecen:
```

Ejemplo 4:

```
Introduzca un número entero: 13247721
Dígitos que aparecen en el número: 1 2 3 4 7
Dígitos que no aparecen: 0 5 6 8 9
```



Ejercicio 49

Realiza un programa que calcule el máximo, el mínimo y la media de una serie de números enteros positivos introducidos por teclado. El programa terminará cuando el usuario introduzca un número primo. Este último número no se tendrá en cuenta en los cálculos. El programa debe indicar también cuántos números ha introducido el usuario (sin contar el primo que sirve para salir).

Ejemplo:

```
Por favor, vaya introduciendo números enteros positivos. Para terminar, introduzca un número p\
rimo:
6
8
15
12
23
Ha introducido 4 números no primos.
Máximo: 15
Mínimo: 6
Media: 10.25
```



Ejercicio 50

Una empresa de cartelería nos ha encargado un programa para realizar uno de sus diseños. Debido a los acontecimientos que han tenido lugar en Cataluña durante el 2018, han recibido muchos pedidos del cartel que muestra el número 155. Realiza un programa que pinte el número 155 mediante asteriscos. Al usuario se le pedirán dos datos, la altura del cartel y el número de espacios que habrá entre los números. La altura mínima es 5. La anchura de los números siempre es la misma. La parte superior de los cinco también es siempre igual. La parte inferior del 5 sí que varía en función de la altura.

Ejemplo 1:

Introduzca la altura (5 como mínimo): 5 Introduzca el número de espacios entre los números (1 como mínimo): 2

```
*  ****  ****
*  *      *
*  ****  ****
*      *      *
*  ****  ****
```

Ejemplo 2:

Introduzca la altura (5 como mínimo): 7 Introduzca el número de espacios entre los números (1 como mínimo): 3

```
*  ****  ****
*   *    *
*  ****  ****
*     *    *
*     *    *
*     *    *
*  ****  ****
```

Ejemplo 3:

Introduzca la altura (5 como mínimo): 6 Introduzca el número de espacios entre los números (1 como mínimo): 1

```
* **** ****
* *   *
* **** ****
*   *   *
*   *   *
* **** ****
```



Ejercicio 51

El gusano numérico se come los dígitos con forma de rosquilla, o sea, el 0 y el 8 (todos los que encuentre). Realiza un programa que muestre un número antes y después de haber sido comido por el gusano. Si el animalito no se ha comido ningún dígito, el programa debe indicarlo.

Ejemplo 1:

```
Introduzca un número entero (mayor que cero): 51803458
Después de haber sido comido por el gusano numérico se queda en 51345
```

Ejemplo 2:

```
Introduzca un número entero (mayor que cero): 29614
El gusano numérico no se ha comido ningún dígito.
```



Ejercicio 52

Realiza un programa que sea capaz de desplazar todos los dígitos de un número de derecha a izquierda una posición. El dígito de más a la izquierda, pasaría a dar la vuelta y se colocaría a la derecha. Si el número tiene un solo dígito, se queda igual.

Ejemplo 1:

Introduzca un número: 609831
El número resultado es 98316

Ejemplo 2:

Introduzca un número: 78201345
El número resultado es 82013457

Ejemplo 3:

Introduzca un número: 24
El número resultado es 42

Ejemplo 4:

Introduzca un número: 8
El número resultado es 8

**Ejercicio 53**

Realiza un programa que pinte un triángulo relleno tal como se muestra en los ejemplos. El usuario debe introducir la altura de la figura.

Ejemplo 1:

Introduzca la altura de la figura: 8

Ejemplo 2:

Introduzca la altura de la figura: 5



Ejercicio 54

Realiza un programa que pinte un triángulo hueco tal como se muestra en los ejemplos. El usuario debe introducir la altura de la figura.

Ejemplo 1:

Introduzca la altura de la figura: 8

* *

* *

* *

* *

* *

**

*

Ejemplo 2:

Introduzca la altura de la figura: 5

* *

* *

**

*



Ejercicio 55

Realiza un programa que sea capaz de desplazar todos los dígitos de un número de izquierda a derecha una posición. El dígito de más a la derecha, pasaría a dar la vuelta y se colocaría a la izquierda. Si el número tiene un solo dígito, se queda igual.

Ejemplo 1:

Introduzca un número: 609831

El número resultado es 160983

Ejemplo 2:

Introduzca un número: 78201345

El número resultado es 57820134

Ejemplo 3:

```
Introduzca un número: 24
El número resultado es 42
```

Ejemplo 4:

```
Introduzca un número: 8
El número resultado es 8
```



Ejercicio 56

Realiza un programa que pinte un triángulo relleno tal como se muestra en los ejemplos. El usuario debe introducir la altura de la figura.

Ejemplo 1:

```
Introduzca la altura de la figura: 8
*****
*****
*****
*****
*****
*****
*****
*****
```

Ejemplo 2:

```
Introduzca la altura de la figura: 5
*****
*****
*****
*****
*****
```



Ejercicio 57

Realiza un programa que pinte un triángulo hueco tal como se muestra en los ejemplos. El usuario debe introducir la altura de la figura.

Ejemplo 1:

Introduzca la altura de la figura: 8

```
*      *
*      *
*      *
*      *
*      *
*      *
*      *
*      *
```

Ejemplo 2:

Introduzca la altura de la figura: 5

```
*  *
*  *
*  *
*  *
*  *
```



Ejercicio 58

Realiza un programa que calcule la media de los dígitos que contiene un número entero introducido por teclado.

Ejemplo 1:

Introduzca un número: 609831

La media de sus dígitos es 4.5

Ejemplo 2:

Introduzca un número: 78201345

La media de sus dígitos es 3.75

Ejemplo 3:

Introduzca un número: 24

La media de sus dígitos es 3.0

Ejemplo 4:

Introduzca un número: 8

La media de sus dígitos es 8.0



Ejercicio 59

Escribe un programa que pinte por pantalla un árbol de navidad. El usuario debe introducir la altura. En esa altura va incluida la estrella y el tronco. Suponemos que el usuario introduce una altura mayor o igual a 4.

Ejemplo 1:

Por favor, introduzca la altura del árbol: 7

[illegible]

Ejemplo 2:

Por favor, introduzca la altura del árbol: 4

$$\begin{array}{c} * \\ \wedge \\ \wedge \wedge \wedge \\ Y \end{array}$$

Ejemplo 3:

Por favor, introduzca la altura del árbol: 10



Ejercicio 60

Escribe un programa que pinte por pantalla un par de calcetines, de los que se ponen al lado del árbol de Navidad para que Papá Noel deje sus regalos. El usuario debe introducir la altura. Suponemos que el usuario introduce una altura mayor o igual a 4. Observa que la talla de los calcetines y la distancia que hay entre ellos (dos espacios) no cambia, lo único que varía es la altura.

Ejemplo 1:

Introduzca la altura de los calcetines: 7

```
***      ***
***      ***
***      ***
***      ***
***      ***
*****   *****
*****   *****
```

Ejemplo 2:

Introduzca la altura de los calcetines: 4

```
***      ***
***      ***
*****   *****
*****   *****
```

Ejemplo 3:

Introduzca la altura de los calcetines: 9

```
***      ***
***      ***
***      ***
***      ***
***      ***
***      ***
***      ***
*****   *****
*****   *****
```



Ejercicio 61

Escribe un programa que pinte por pantalla la letra V. El ancho del palo de la V es siempre de 3 asteriscos. El usuario debe introducir la altura. La altura mínima es de 3 pisos. Si el usuario introduce una altura menor, el programa debe mostrar un mensaje de error.

Ejemplo 1:

Introduzca la altura de la V (un número mayor o igual a 3): 7

```
***          ***
***          ***
***          ***
***          ***
***          ***
***          ***
*****
```

Ejemplo 2:

Introduzca la altura de la V (un número mayor o igual a 3): 4

```
***          ***
***          ***
***          ***
*****
```

Ejemplo 3:

Introduzca la altura de la V (un número mayor o igual a 3): 9

```
***          ***
***          ***
***          ***
***          ***
***          ***
***          ***
***          ***
***          ***
*****
```

Ejemplo 4:

Introduzca la altura de la V (un número mayor o igual a 3): 2

La altura debe ser mayor o igual a 3.



Ejercicio 62

Según cierta cultura oriental, los números de la suerte son el 3, el 7, el 8 y el 9. Los números de la mala suerte son el resto: el 0, el 1, el 2, el 4, el 5 y el 6. Un número es **afortunado** si contiene más números de la suerte que de la mala suerte. Realiza un programa que diga si un número introducido por el usuario es afortunado o no.

Ejemplo 1:

```
Introduzca un número: 772
El 772 es un número afortunado.
```

Ejemplo 2:

```
Introduzca un número: 7720
El 7720 no es un número afortunado.
```

Ejemplo 3:

```
Introduzca un número: 43081
El 43081 no es un número afortunado.
```

Ejemplo 4:

```
Introduzca un número: 888
El 888 es un número afortunado.
```

Ejemplo 5:

```
Introduzca un número: 1234
El 1234 no es un número afortunado.
```

Ejemplo 6:

```
Introduzca un número: 6789
El 6789 es un número afortunado.
```

**Ejercicio 63**

Realiza un programa que pinte dos pirámides rellenas hechas con asteriscos, una al lado de la otra y separadas por un espacio en su base.

Ejemplo 1:

```
Introduzca la altura de la primera pirámide: 7
Introduzca la altura de la segunda pirámide: 3
```

```
      *
    ***
  *****
*****
*****      *
*****      ***
***** *****
```

Ejemplo 2:

Introduzca la altura de la primera pirámide: 4

Introduzca la altura de la segunda pirámide: 5

```

      *
    *   ***
  ***   *****
*****   **********
*****   **********

```



Ejercicio 64

Escribe un programa que pinte por pantalla un **rectángulo hueco** de 6 caracteres de ancho por 3 de alto y, a continuación, un menú que permita **agrandarlo**, **achicarlo** o **cambiar su orientación**. Cada vez que el rectángulo se agranda, se incrementa en 1 tanto su anchura como su altura. Cuando se achica, se decrementa en 1 su anchura y altura. Por último, cuando se cambia la orientación, los valores de anchura y altura se intercambian. El valor mínimo de la altura o la anchura es 2.

Ejemplo:

```

*****
*      *
*****

1. Agrandarlo
2. Achicarlo
3. Cambiar la orientación
4. Salir
Indique qué quiere hacer con el rectángulo: 2

```

```

****
****

1. Agrandarlo
2. Achicarlo
3. Cambiar la orientación
4. Salir
Indique qué quiere hacer con el rectángulo: 2

```

```

****
****

1. Agrandarlo
2. Achicarlo

```

```

3. Cambiar la orientación
4. Salir
Indique qué quiere hacer con el rectángulo: 1

```

```

*****
*   *
*****
1. Agrandarlo
2. Achicarlo
3. Cambiar la orientación
4. Salir
Indique qué quiere hacer con el rectángulo: 3

```

```

***
* *
* *
* *
***
1. Agrandarlo
2. Achicarlo
3. Cambiar la orientación
4. Salir
Indique qué quiere hacer con el rectángulo: 4

```



Ejercicio 65

Escribe un programa que pinte por pantalla la letra A. El usuario debe introducir la altura total y la fila en la que debe aparecer el palito horizontal (contando desde el vértice). La altura mínima es de 3 pisos. La fila donde va el palito horizontal debe ser mayor que 1 y menor que la altura total. Si el usuario introduce algún dato incorrecto, el programa debe mostrar un mensaje de error.

Ejemplo 1:

```

Introduzca la altura de la A (un número mayor o igual a 3): 7
Introduzca la fila del palito horizontal (entre 2 y 6): 5

```

```

      *
    * *
  *   *
*     *
*****
*       *
*         *

```

Ejemplo 2:

Introduzca la altura de la A (un número mayor o igual a 3): 7

Introduzca la fila del palito horizontal (entre 2 y 6): 6

```
      *
    * *
  *   *
*     *
*     *
*****
*           *
```

Ejemplo 3:

Introduzca la altura de la A (un número mayor o igual a 3): 7

Introduzca la fila del palito horizontal (entre 2 y 6): 7

La fila introducida no es correcta.

Ejemplo 4:

Introduzca la altura de la A (un número mayor o igual a 3): 2

La altura introducida no es correcta.

Ejemplo 5:

Introduzca la altura de la A (un número mayor o igual a 3): 4

Introduzca la fila del palito horizontal (entre 2 y 3): 2

```
      *
    ***
  *   *
*     *
```

Ejemplo 1:

Introduzca la altura de la A (un número mayor o igual a 3): 5

Introduzca la fila del palito horizontal (entre 2 y 4): 4

```
      *
    * *
  *   *
*****
*           *
```



Ejercicio 66

La Guardia Civil de Tráfico nos ha encargado un programa que pinte una señal para desviar el tráfico hacia la derecha. La señal es una doble flecha con el vértice apuntando a la derecha. Se pide al usuario la altura de la figura, que debe ser un número impar mayor o igual que 3. La distancia entre cada flecha de asteriscos es siempre de 4 espacios. Si la altura introducida por el usuario no es un número impar mayor o igual que 3, el programa debe mostrar un mensaje de error.

Ejemplo 1:

Por favor, introduzca la altura de la figura: 7

```
*      *
*      *
*      *
*      *
*      *
*      *
*      *
```

Ejemplo 2:

Por favor, introduzca la altura de la figura: 3

```
*      *
*      *
*      *
```

Ejemplo 3:

Por favor, introduzca la altura de la figura: 4

La altura no es correcta, debe ser un número impar mayor o igual que 3.



Ejercicio 67

Realiza un programa que pinte una escalera que va descendiendo de izquierda a derecha. El programa pedirá el número de escalones y la altura de cada escalón. La anchura de los escalones siempre es la misma: 4 asteriscos.

Ejemplo 1:

```
Introduzca el número de escalones: 4
Introduzca la altura de cada escalón: 2
****
****
*****
*****
*****
*****
*****
*****
*****
```

Ejemplo 2:

```
Introduzca el número de escalones: 3
Introduzca la altura de cada escalón: 3
****
****
****
*****
*****
*****
*****
*****
*****
```



Ejercicio 68

Escribe un programa que pida un número por teclado y que luego lo **“disloque”** de tal forma que a cada dígito se le suma 1 si es par y se le resta 1 si es impar. Usa long en lugar de int donde sea necesario para admitir números largos.

Ejemplo 1:

```
Por favor, introduzca un número: 9402
Dislocando el 9402 sale el 8513.
```

Ejemplo 2:

```
Por favor, introduzca un número: 870958422
Dislocando el 870958422 sale el 961849533.
```

Ejemplo 3:

```
Por favor, introduzca un número: 137
Dislocando el 137 sale el 26
```



Ejercicio 69

Realiza un programa que pinte una **pirámide maya**. Por los lados, se trata de una pirámide normal y corriente. Por el centro se van pintando líneas de asteriscos de forma alterna (empezando por la superior): la primera se pinta, la segunda no, la tercera sí, la cuarta no, etc. La terraza de la pirámide siempre tiene 6 asteriscos, por tanto, las líneas centrales que se añaden a la pirámide normal tienen 4 asteriscos. El programa pedirá la altura. Se supone que el usuario introducirá un número entero mayor o igual a 3; no es necesario comprobar los datos de entrada.

Ejemplo 1:

Introduzca la altura de la pirámide maya: 5

```

*****
**      **
*****
****      ****
*****

```

Ejemplo 2:

Introduzca la altura de la pirámide maya: 8

```

*****
**      **
*****
****      ****
*****
*****      *****
*****
*****      *****

```