

# TALLER DE PROGRAMACIÓN

## JUAN CARLOS ZULUAGA

---

### Contenido

1	NUMERO ES PAR O NO .....	3
2	NÚMERO MAYOR.....	3
3	ORDENACIÓN DE TRES NÚMEROS.....	3
4	NUMERO ES MÚLTIPLO DE OTRO .....	3
5	AÑO BICIESTO .....	4
6	NOMINA SENCILLA .....	4
7	DESCUENTOS.....	5
8	VALOR MATRICULA .....	5
9	ENVÍO DE MERCANCÍAS .....	6
10	ALMACENES SUCESO S.A.....	7
11	EMPRESA DE TRANSPORTES .....	9
12	SERIE DE NÚMEROS, SUMATORIA Y PROMEDIO .....	12
13	CÁLCULO DEL FACTORIAL DE UN NÚMERO .....	12
14	SERIE DE TAYLOR.....	13
15	SERIE DE TAYLOR MODIFICADA .....	13
16	ECUACIONES CUADRÁTICAS .....	13
17	SERIE FIBONACCI .....	14
18	SERIE FIBONACCI MODIFICADA.....	14
19	¿ES PRIMO?.....	14
20	NÚMEROS PRIMOS .....	15
21	CALCULO DEL NÚMERO $e$ .....	15
22	CALCULO DEL NÚMERO $\pi$ .....	15
23	CALENDARIO DE UN AÑO.....	16
24	PARES E IMPARES.....	18
25	OPERACIONES EN UN VECTOR.....	19
26	ORDENACIÓN ESPECIAL DE VECTOR.....	19
27	FRASES PALÍNDROMAS .....	20
28	LA VIGA MÁS RESISTENTE .....	21

29	PUENTES DE MADISON .....	22
30	OPERACIONES EN UNA MATRIZ.....	23
31	TRIANGULAR INFERIOR DE UNA MATRIZ.....	23
32	ROMBO.....	24
33	MULTIPLICACIÓN DE MATRICES .....	25
34	RELOJ DE ARENA .....	25
35	DESCOMPOSICION EN FACTORES .....	26
36	COSECHANDO A CABALLO .....	27
37	CABALLOS EN CONFLICTO .....	28

## 1 NUMERO ES PAR O NO

Construir un programa que pida un número y luego diga si este número es par o no.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese número: 10
El número: 10, es par
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese número: 7
El número: 7, es impar
```

## 2 NÚMERO MAYOR

Construir un programa que pida por pantalla 3 números y luego diga cuál es el mayor de los números ingresados.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese primer número: 45
Ingrese segundo número: 23
Ingrese tercer número: 12
El número mayor es: 45
```

## 3 ORDENACIÓN DE TRES NÚMEROS

Construir un programa que pida por pantalla 3 números y luego diga cuál es el mayor, el del medio y el menor de los números ingresados.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese primer número: 45
Ingrese segundo número: 23
Ingrese tercer número: 78
El número mayor es: 78
El número del medio es: 45
El número menor es: 23
```

## 4 NUMERO ES MÚLTIPLO DE OTRO

Construir un programa que pida dos número y diga si el segundo es múltiplo del primero.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese primer número: 45
Ingrese segundo número: 3
El número: 3, es múltiplo de: 45
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese primer número: 123
Ingrese segundo número: 7
El número: 7, no es múltiplo de: 123
```

## 5 AÑO BISIESTO

Construir un programa que pida un año y luego determine si el año es bisiesto o no. Tenga en cuenta que los años bisiestos son los números múltiplos de 4, pero que no son múltiplos de 100 y si son múltiplos de 100 y múltiplos de 400 sí son bisiestos.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese año: 1974
El año: 1974, no es bisiesto
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese año: 2000
El año: 2000, si es bisiesto
```

**Prueba 3:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese año: 2012
El año: 2012, si es bisiesto
```

**Prueba 4:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese año: 2200
El año: 2200, no es bisiesto
```

## 6 NOMINA SENCILLA

Elaborar un programa que entre el nombre de un empleado, su salario básico por hora y el número de horas trabajadas en el mes; escriba su nombre y salario mensual si éste es mayor del salario mínimo, de lo contrario escriba sólo el nombre.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese nombre: Luis Martinez
Ingrese número de horas trabajadas: 45
Ingrese valor hora: 16000
Ingrese valor salario mínimo mensual: 585000
Nombre: Luis Martinez
Salario mensual: 720000
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese nombre: Pedro Suarez
Ingrese número de horas trabajadas: 38
Ingrese valor hora: 5200
Ingrese valor salario mínimo mensual: 585000
Nombre: Pedro Suarez
```

## 7 DESCUENTOS

Un almacén de escritorios hace los siguientes descuentos: si el cliente compra menos de 5 unidades se le da un descuento del 10% sobre la compra; si el número de unidades es mayor o igual a cinco pero menos de 10 se le otorga un 20% y, si son 10 o más se le da un 40%.

Hacer un programa que determine cuánto debe pagar un cliente si el valor de cada escritorio es de \$650.000.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Número de escritorios: 7
El valor a pagar es: 3640000
```

## 8 VALOR MATRICULA

Se desea obtener el valor de la matrícula de un estudiante cuyo valor se calcula de la siguiente manera en un subprograma:

- Si toma 20 o menos créditos, paga el crédito al valor normal.
- Si toma por encima de 20 créditos, se pagarán los créditos extras al doble de valor normal.
- Si el estudiante es de estrato 1, 2 o 3 recibe los siguientes descuentos:
  - ✓ Si el estrato es 1, el descuento es del 80%.
  - ✓ Si el estrato es 2, el descuento es del 50%.
  - ✓ Si el estrato es 3, el descuento es del 30%.

Además los estratos 1 y 2 reciben subsidio de alimentación y transporte de la siguiente manera (el cual se debe calcular en otro subprograma):

- Para el estrato 1, el subsidio de alimentación y transporte es igual a \$200.000.
- Para el estrato 2, el subsidio de alimentación y transporte es igual a \$100.000.

Se debe informar al usuario sobre el costo de la matrícula y el valor del subsidio.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Número de créditos: 25
Valor crédito: 85000
Estrato del estudiante: 2
Costo de la matricula: 1275000
Valor del subsidio: 100000
```

## 9 ENVÍO DE MERCANCÍAS

Una empresa de envío de mercancías, tiene un plan de tarifas y descuentos sobre el valor del envío de mercancía de sus clientes:

- Tarifas:
  - ✓ Si el peso de la mercancía es inferior a 100 kg, la tarifa para el envío de ésta es de 20.000 pesos.
  - ✓ Si el peso de la mercancía está entre 100 y 150 kg, la tarifa para el envío de ésta es de 25.000 pesos.
  - ✓ Si el peso de la mercancía es superior a 150 kg y menor o igual a 200 kg, la tarifa para el envío de ésta es de 30.000 pesos.
  - ✓ Si el peso de la mercancía es superior a 200 kg, la tarifa es de 35.000 pesos y además por cada 10 kg adicionales se paga 2.000 pesos.
- Descuentos:
  - ✓ Si el valor de la mercancía está entre 300.000 y 600.000 pesos se hace un descuento del 10% sobre el valor del envío.
  - ✓ Si el valor de la mercancía es superior a 600.000 pero menor o igual a 1.000.000 de pesos se hace un descuento del 20% sobre el valor del envío.
  - ✓ Si el valor de la mercancía es superior a 1.000.000 se hace un descuento del 30% sobre el valor del envío.
- Promociones (solo hay dos tipos de pago):
  - ✓ Si es día de promoción (lunes) y paga con tarjeta propia del almacén, sólo paga el 50% del costo de envío.
  - ✓ Si paga en efectivo y el valor de la mercancía es superior a 1.000.000, sólo paga el 60% del costo de envío.

Si el cliente aplica a una promoción, no puede aplicar a un descuento. Se debe obtener el valor total del envío.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Peso de la mercancía: 289
Valor de la mercancía: 780000
Es lunes [S]í [N]o: s
Tipo de pago [E]fectivo [T]arjeta: t
Tarifa: 51000
Promoción: 25500
Total a pagar: 25500
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Peso de la mercancía: 150
Valor de la mercancía: 1200000
Es lunes [S]í [N]o: n
Tipo de pago [E]fectivo [T]arjeta: t
Tarifa: 25000
Descuento: 7500
Total a pagar: 17500
```

## 10 ALMACENES SUCESO S.A.

Almacenes SUCESO S.A. desea saber el valor de venta de un producto **VR\_V**, a partir de la siguiente información que posee de éste:

Costo de compra **CC**

Tipo de producto **TP** (Perecedero o No perecedero).

Tipo de conservación **TC** (frio o Ambiente)

Periodo de Conservación en días. **PC**

Periodo de almacenamiento en días **PA**

Volumen en litros **VOL**

Medio de almacenamiento **MA** (Nevera, congelador, estantería y guacal)

El valor de venta **VR\_V** es igual a:

El 20% más del valor del producto **VR\_P**, cuando el tipo de producto **TP** corresponde a un artículo no perecedero, o El 40% más del valor del producto **VR\_P**, cuando el tipo de producto **TP** corresponde a un artículo perecedero.

El valor del producto **VR\_P**, se calcula de la suma del Costo de compra **CC**, costo de almacenamiento **CA** y costo de exhibición **CE**, multiplicados por el porcentaje de depreciación del producto **PDP**.

Los costos y el porcentaje de depreciación deben ser calculados en subprogramas así:

**Función Costo de Almacenamiento.** En este se debe calcular el costo de almacenamiento **CA** para ser devuelto al programa principal.

El costo de almacenamiento **CA**, se diferencia para artículos perecederos y no perecederos.

- Para los productos perecederos:

Costo de almacenamiento **CA** es igual al 5% del costo de compra **CC** cuando se trata de un producto de tipo de conservación **TC** en frio y con un período de conservación **PC** inferior a 10 días.

Costo de almacenamiento **CA** es igual al 10% del costo de compra **CC** cuando se trata de un producto de tipo de conservación **TC** en frio y con un período de conservación **PC** superior o igual 10 días.

Costo de almacenamiento **CA** es igual al 3% del costo de compra **CC** cuando se trata de un producto de tipo de conservación **TC** en ambiente y con un período de almacenamiento **PA** en inferior a 20 días.

Costo de almacenamiento **CA** es igual al 10% del costo de compra **CC** cuando se trata de un producto de tipo de conservación **TC** en ambiente y con un período de almacenamiento **PA** superior a 20 días.

Costo de almacenamiento **CA** es igual al 5% del costo de compra **CC** cuando se trata de un producto de tipo de conservación **TC** en ambiente y con un período de almacenamiento **PA** exacto a 20 días.

- Para los productos no perecederos

El costo de almacenamiento **CA** se calcula a través de su volumen en litros **VOL** así:

Costo de almacenamiento **CA** es igual al 10% del costo de compra **CC** cuando el volumen **VOL** es mayor o igual a los 50 litros,

Costo de almacenamiento **CA** es igual al 20% del costo de compra **CC** cuando el volumen **VOL** es menor a los 50 litros.

**Función Porcentaje Depreciación del Producto.** En este se debe calcular el porcentaje de depreciación del producto **PDP**, para ser devueltos al programa principal.

El porcentaje de depreciación del producto **PDP** es igual al 0.95 cuando el período de almacenamiento **PA** es inferior a 30 días, o

El porcentaje de depreciación del producto **PDP** es igual al 0.85 cuando el período de almacenamiento **PA** es superior o igual a 30 días

**Función Costo de e:** El costo de exhibición **CE** también se diferencia entre artículos perecederos y no perecederos, calculándose así:

- Para los productos perecederos:

Aquellos artículos perecederos cuyo tipo de conservación **TC** es frío, y el medio de almacenamiento **MA** es nevera, el costo de exhibición **CE** corresponde al doble del costo de almacenamiento **CA**.

Aquellos artículos perecederos cuyo tipo de conservación **TC** es frío, y el medio de almacenamiento **MA** es congelador, el costo de exhibición **CE** corresponde al del costo de almacenamiento **CA**.

- Para los productos no perecederos:

El costo de exhibición **CE** corresponde al 5% del costo de almacenamiento **CA**, para los productos cuyo medio de almacenamiento **MA** es en estantería

El costo de exhibición **CE** corresponde al 7% del costo de almacenamiento **CA**, para los productos cuyo medio de almacenamiento **MA** es en guacal



**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
*** DATOS DE ENTRADA ***
Costo de compra ($).....: 560000
Tipo de producto [P]erecedero, [N]o perecedero.....: n
Tipo de conservación [F]rio, [A]mbiente.....: a
Periodo de conservación (días).....: 150
Periodo de almacenamiento (días).....: 30
Volumen (litros).....: 600
Medio de almacenamiento [N]evera, [C]ongelador, [E]stanteria, [G]uacal: e
*** CALCULOS ***
Costos de almacenamiento.....: 56000
Porcentaje de depreciación.....: 0,85
Costo de exhibición.....: 2800
Valor producto.....: 525980
Valor venta.....: 631176
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
*** DATOS DE ENTRADA ***
Costo de compra ($).....: 1020000
Tipo de producto [P]erecedero, [N]o perecedero.....: p
Tipo de conservación [F]rio, [A]mbiente.....: f
Periodo de conservación (días).....: 60
Periodo de almacenamiento (días).....: 30
Volumen (litros).....: 600
Medio de almacenamiento [N]evera, [C]ongelador, [E]stanteria, [G]uacal: n
*** CALCULOS ***
Costos de almacenamiento.....: 102000
Porcentaje de depreciación.....: 0,85
Costo de exhibición.....: 204000
Valor producto.....: 1127100
Valor venta.....: 1577940
```

## 11 EMPRESA DE TRANSPORTES

Una empresa de transportes necesita liquidar al final del día el valor que debe liquidar al conductor por concepto de honorarios.

El valor a pagar al conductor se calcula con número de viajes por valor de la ruta (solo un conductor hace una ruta, pero puede tener varios viajes). Adicional a eso se tiene un número mínimo de pasajeros por ruta y una tabla de comisiones que se le pagan al conductor dependiendo del número de pasajeros que viajan en el total de todos los viajes de una ruta.

Ruta	Valor	Comisiones Por Número de Pasajeros				
		Menos de 50	Entre 51 y 100	Entre 101 y 150	Entre 151 y 200	Más de 200
1	\$500,000	0%	5%	6%	7%	7% más \$50 por cada pasajero adicional a los 200
2	\$600,000	0%	7%	8%	9%	9% más \$60 por cada pasajero adicional a los 200
3	\$800,000	0%	10%	13%	15%	15% más \$100 por cada pasajero adicional a los 200
4	\$1,000,000	0%	12.5%	15%	17%	17% más \$150 por cada pasajero adicional a los 200

La empresa no solo transporta pasajeros, también presta el servicio de encomiendas y al conductor se le paga un porcentaje, dependiendo del número y el peso de las encomiendas enviadas.

# Paquetes - Peso	Valor x Paquete Ruta 1 y Ruta 2		Valor x Paquete Ruta 3 y Ruta 4		
	Menos de 10Kg	Más o igual a 10Kg	Menos 10Kg	Más o igual a 10Kg y Menos de 20Kg	Más o igual 20Kg
Menos de 50	\$100	\$120	\$130	\$140	\$170
Entre 50 y 100	\$120	\$140	\$160	\$180	\$210
Entre 101 y 130	\$150	\$160	\$175	\$200	\$250
Más de 130	\$160	\$180	\$200	\$250	\$300

El conductor debe pagar el ayudante y un seguro de viajes. Estos se calculan con un porcentaje del valor que recibe el conductor con la siguiente tabla:

Ingresos Conductor	% Pagar Ayudante	% Pagar Seguro
Menos de \$1,000,000	5%	3%
Entre \$1,000,000 y \$2,000,000	8%	4%
Entre \$2,000,001 y \$4,000,000	10%	6%
Más de \$4,000,000	13%	9%

El valor del galón de gasolina es: \$8,860 y estos buses rinden 39Km por galón. El conductor paga el combustible de su bolsillo, pero la empresa reconoce un subsidio del 25% del valor del combustible al conductor. Para calcular el consumo de gasolina aproximado se usa la siguiente tabla. **Nota:** cada persona pesa en promedio 60Kg.

Ruta	Kilómetros
1	150
2	167
3	184
4	203

Recargos de consumo de combustible por peso en el bus (se debe tomar en cuenta el peso de los pasajeros y el peso de las encomiendas)

Peso	% Incremento
Menor o igual a 5,000 Kg	0%
Mayor a 5,000 Kg y Menor o igual a 10,000 Kg	10%
Más de 10,000 Kg	25%

Se debe mostrar el valor a liquidar al conductor y al ayudante. Se debe realizar la prueba de escritorio con los siguientes valores:

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

*** DATOS DE ENTRADA ***	
Ruta [1][2][3][4].....	2
Número de viajes.....	3
Número de pasajeros total.....	145
Número de encomiendas de menos de 10Kg.....	34
Número de encomiendas entre 10Kg y menos de 20Kg:	23
Número de encomiendas de más de 20Kg.....	12
*** CALCULOS ***	
Ingresos por Pasajeros.....	1944000
Ingresos por Encomiendas.....	7600
	:
TOTAL INGRESOS.....	1951600
Pago Ayudante.....	156128
Pago Seguro.....	78064
Pago Combustible.....	79740
	:
TOTAL DEDUCCIONES.....	313932
	:
TOTAL A LIQUIDAR.....	1637668

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

*** DATOS DE ENTRADA ***	
Ruta [1][2][3][4].....	3
Número de viajes.....	3
Número de pasajeros total.....	134
Número de encomiendas de menos de 10Kg.....	36
Número de encomiendas entre 10Kg y menos de 20Kg:	67
Número de encomiendas de más de 20Kg.....	12
*** CALCULOS ***	
Ingresos por Pasajeros.....	2711999
Ingresos por Encomiendas.....	18780
	:
TOTAL INGRESOS.....	2730779
Pago Ayudante.....	273077
Pago Seguro.....	1638467

Pago Combustible.....:	93030
:	_____
TOTAL DEDUCCIONES.....:	2004574
:	_____
TOTAL A LIQUIDAR.....:	726205

## 12 SERIE DE NÚMEROS, SUMATORIA Y PROMEDIO

Construir un programa que imprima los N primeros, su sumatoria y su promedio.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

Cuantos números desea: 100														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	16	17	18	19	20	21	22	23	24	25	26	27	28	29
	30	31	32	33	34	35	36	37	38	39	40	41	42	43
	44	45	46	47	48	49	50	51	52	53	54	55	56	57
	58	59	60	61	62	63	64	65	66	67	68	69	70	71
	72	73	74	75	76	77	78	79	80	81	82	83	84	85
	86	87	88	89	90	91	92	93	94	95	96	97	98	99
	100													
La suma es: 5050														
El promedio es: 50														

## 13 CÁLCULO DEL FACTORIAL DE UN NÚMERO

Construir un programa que imprima el factorial de un número dado. La función factorial se representa como.

$$n! = 1 \times 2 \times 3 \times 4 \times \dots \times (n - 1) \times n$$

Ó también como:

$$n! = \prod_{k=1}^n k$$

Es decir, que factorial de 5 = 1 x 2 x 3 x 4 x 5 = 120.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese número: 5
El factorial es: 120

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese número: 20
El factorial es: 2432902008176640000

## 14 SERIE DE TAYLOR

Construir un programa que calcule los n primeros términos de la serie de Taylor que está representada por:

$$f(x) = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots, \quad -\infty < x < \infty$$

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos desea: 20
Digita el valor de x: 1
f(x) = 2,718282
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos desea: 500
Digita el valor de x: 2
f(x) = 7,389056
```

## 15 SERIE DE TAYLOR MODIFICADA

Construir un programa que calcule los n primeros términos de la serie representada por:

$$f(x) = 1 - \frac{x}{1!} + \frac{x^2}{2!} - \frac{x^3}{3!} + \dots, \quad -\infty < x < \infty$$

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos desea: 20
Digita el valor de x: 1
f(x) = 0,367879
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos desea: 300
Digita el valor de x: 8
f(x) = 0,000335
```

## 16 ECUACIONES CUADRÁTICAS

Construir un programa que resuelva los valores de X para una ecuación cuadrática la cual está dada por:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese valor de a: 1
Ingrese valor de b: 3
Ingrese valor de c: 2
x1: -1,00000
x2: -2,00000
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese valor de a: 2
Ingrese valor de b: 8
Ingrese valor de c: 3
x1: -0,41886
x2: -3,58114
```

## 17 SERIE FIBONACCI

Construir un programa que imprima los N primeros números de la serie Fibonacci y su sumatoria, la cual está representada por:

0, 1, 1, 2, 3, 5, 8, 13, 21,...

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos quiere: 20
0      1      1      2      3      5      8      13     21     34     55     89     144    233    377
      610    987    1597   2584   4181   6765
La suma es: 17710
```

## 18 SERIE FIBONACCI MODIFICADA

Construir un programa que imprima los N primeros números de la siguiente serie, la cual está representada por:

0, 1, 2, 3, 6, 11, 20, 37, 68,...

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos desea: 20
0      1      2      3      6      11     20     37     68     125    230    423    778    1431   2632
      4841   8904   16377  30122  55403
La sumatoria es: 121414
```

## 19 ¿ES PRIMO?

Construir un programa que pida un número por pantalla y luego diga si es primo o no.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese número: 997
El número: 997, es primo
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese número: 9
El número: 9, no es primo
```

## 20 NÚMEROS PRIMOS

Construir un programa que imprima los N primeros números primos, su sumatoria y promedio.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos primos desea: 10
1      2      3      5      7      11     13     17     19     23
La suma es: 101
El promedio es: 10
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos primos desea: 30
1      2      3      5      7      11     13     17     19     23     29     31     37     41     43
      47     53     59     61     67     71     73     79     83     89     97     101    103    107
      109
La suma es: 1481
El promedio es: 49
```

## 21 CALCULO DEL NÚMERO $e$

El número natural  $e$  se puede calcular con esta serie matemática:

$$e = \frac{1}{0!} + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \cdots + \frac{1}{n!}$$

Entre mayor sea el número  $n$ , mayor será la precisión del número natural  $e$ . Hacer un programa que calcule el número natural  $e$ , dado el número de términos requeridos por el usuario por el usuario.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantos términos de precisión desea: 10
e = 2.718281525573192243
```

## 22 CALCULO DEL NÚMERO $\pi$

El número  $\pi$  (pi) se puede calcular con esta serie matemática:

$$\frac{\pi}{4} = \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{1}{n}$$

Entre mayor sea el número de términos, mayor será la precisión del número  $\pi$ . Hacer un programa que calcule el número  $\pi$ , dado el número de términos requeridos por el usuario.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

Cuantos términos de precisión desea: 800  
pi = 3.1403426540780757

## 23 CALENDARIO DE UN AÑO

Construir un programa que imprima el calendario de un año. El programa debe tener en cuenta si el año es bisiesto o no y el día en que empieza cada día de cada mes.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

Digite el año: 2012

Mes: 1

Dom	Lun	Mar	Mie	Jue	Vie	Sab
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Mes: 2

Dom	Lun	Mar	Mie	Jue	Vie	Sab
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29			

Mes: 3

Dom	Lun	Mar	Mie	Jue	Vie	Sab
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Mes: 4

Dom	Lun	Mar	Mie	Jue	Vie	Sab
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21



22	23	24	25	26	27	28
29	30					

Mes: 5

Dom	Lun	Mar	Mie	Jue	Vie	Sab
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Mes: 6

Dom	Lun	Mar	Mie	Jue	Vie	Sab
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Mes: 7

Dom	Lun	Mar	Mie	Jue	Vie	Sab
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Mes: 8

Dom	Lun	Mar	Mie	Jue	Vie	Sab
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Mes: 9

Dom	Lun	Mar	Mie	Jue	Vie	Sab
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

Mes: 10

Dom	Lun	Mar	Mie	Jue	Vie	Sab
	1	2	3	4	5	6

7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Mes: 11

Dom	Lun	Mar	Mie	Jue	Vie	Sab
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Mes: 12

Dom	Lun	Mar	Mie	Jue	Vie	Sab
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

**Nota:** entre día y día va un tabulador, cada que empieza una nueva semana va un salto de línea y cada que empieza un nuevo mes van dos saltos de línea.

**Ayuda:** con la congruencia de Zeller se puede determinar qué día inicia la semana de cualquier mes. Si te sirve, puedes usar esta función de Zeller:

```
// Devuelve
// 0 = Domingo, 1 = Lunes, 2 = Martes, 3 = Miercoles,
// 4 = Jueves, 5 = Viernes, 6 = Sábado
private static int zeller(int ano, int mes) {
    int a = (14 - mes) / 12;
    int y = ano - a;
    int m = mes + 12 * a - 2;
    int dia = 1, d;

    d = (dia + y + y / 4 - y / 100 + y / 400 + (31 * m) / 12) % 7;

    return (d);
}
```

## 24 PARES E IMPARES EN UN ARREGLO

Construir un programa que cree un vector de N posiciones. Si la posición es par, la celda del vector se debe llenar con la siguiente formula  $Celda[i] = i + 7$ . Si la celda es impar, la celda del vector se debe llenar con la siguiente formula  $Celda[i] = i - 1$ . Se debe mostrar como quedo el arreglo y luego debe mostrar la sumatoria de las celdas con valores pares y la productoria de las celdas con valores impares.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantas posiciones desea: 10
vec[0]=7
vec[1]=0
vec[2]=9
vec[3]=2
vec[4]=11
vec[5]=4
vec[6]=13
vec[7]=6
vec[8]=15
vec[9]=8
La suma es: 20
La productoria es: 135135
```

## 25 OPERACIONES EN UN VECTOR

Construir un programa que cree un vector N posiciones. El vector se debe llenar con la siguiente formula  $Celda[i] = (i + 1)$   
\* i. Luego debe mostrar la sumatoria y el promedio de los elementos del arreglo.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantas posiciones desea: 10
330
33
```

## 26 ORDENACIÓN ESPECIAL DE VECTOR

Construir un programa que cree un vector de N posiciones. Si la posición es par, la celda del vector se debe llenar con la siguiente formula  $Celda[i] = i * i + 1$ . Si la celda es impar, la celda del vector se debe llenar con la siguiente formula  $Celda[i] = 3 * (i + 1)$ . Luego los elementos que contengan un número par del vector se deben ordenar descendentemente en las primeras posiciones del vector y los elementos impares del vector se deben ordenar ascendentemente en las últimas posiciones del vector. Se debe mostrar el vector sin ordenar y luego el vector ordenado.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Cuantas posiciones desea: 10
SIN ORDENAR
vec[0]=1
vec[1]=6
vec[2]=5
vec[3]=12
vec[4]=17
vec[5]=18
vec[6]=37
vec[7]=24
vec[8]=65
```

```
vec[9]=30
ORDENADO
vec[0]=30
vec[1]=24
vec[2]=18
vec[3]=12
vec[4]=6
vec[5]=1
vec[6]=5
vec[7]=17
vec[8]=37
vec[9]=65
```

## 27 FRASES PALÍNDROMAS

Las palíndromas son palabras o frases que se leen igual de derecha a izquierda que en sentido habitual. Es conocido el Prueba "Dábale arroz a la zorra el Abad". La raíz etimológica palin significa "de nuevo" o "repetición".

Estas palabras o frases deben ser iguales (al derecho y al revés) para los palíndromos simples ("Ana", "Anilina") y deben ser distintas (al derecho y al revés) para los palíndromos bifrontes ("Anita lava la tina").

En la lectura al revés, la forma de agrupar las letras en palabras es la que se necesite. Dicho de otra manera, las frases no están formadas sólo por palabras que son palíndromos necesariamente. Al leer al revés, una palabra puede formarse usando letras de más de una de las palabras escritas al derecho.

Muchos escritores se han dejado fascinar por los palíndromos: Nabokov, Poe o Lewis Carroll se mostraron muy interesados por las frases capicúa. A Joyce, por Prueba, se le atribuye el tímido "Madam, I'm Adam". En castellano destacan, entre muchos otros, Augusto Monterroso, que tituló uno de sus cuentos "Onís es asesino"; o Julio Cortázar, que usando un palíndroma como "Atar a la rata", escribió un cuento. De ahí, surgió "Satarsa, la rata", plural de la frase anterior.

Recopilación de algunas frases palíndromas

- A cavar a Caravaca
- A mamá Roma le aviva el amor a papá y a papá Roma le aviva el amor a mamá
- Abusón, acá no suba
- ¿Acaso comeré mocos acá?
- Adán no calla con nada
- Allá, cada gorda drogada, calla
- Ánimo Romina
- Anita, la gorda lagartona, no traga la droga latina
- Anula la luz azul a la Luna
- Arde ya la yedra
- Arena mala me da de mala manera
- Dábale arroz a la zorra el Abad
- Ella te dará detalle
- Isaac no ronca así

- La ruta nos aportó otro paso natural
- Ni nicotina ni tocinín
- Odiosa, ¿has oído?
- Oye, sí. Versos revisé yo
- Roma le da té o pan a poeta del amor
- Roza las alas al azor
- Senil oí violines
- Sé verle del revés
- Sin anís o no, como taco. Coca tomo con o sin anís
- Sonreí, Bogart no cede contra gobiernos
- ¿Subo tu auto o tu autobús?
- Yo dono rosas, oro no doy
- Yo hallé ropa, yo voy a por ella hoy

Realiza un programa que se ingrese una frase completa, asuma que toda es ingresada en mayúsculas y que no tiene tildes. Y luego diga si la palabra o frase es palíndroma o no. Tenga presente que no se deben tener en cuenta los espacios en blanco o símbolos de puntuación como comas y puntos entre otros.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese la frase: LA RUTA NOS APORTO OTRO PASO NATURAL
La frase SI es palíndroma
```

## 28 LA VIGA MÁS RESISTENTE

Una viga horizontal (construida de izquierda a derecha) suspendida en el aire requiere de una base para conectarla a tierra; son tres tipos de bases con diferentes resistencias: %, &, #. El resto de la viga se construye de dos piezas: largueros (=) y conexiones (\*); los largueros se pueden conectar con otros largueros o con una conexión. Las conexiones solamente pueden conectarse con largueros, no hay dos o más conexiones conectadas sueltamente.

Las bases tiene las siguientes resistencias: % resiste 10 unidades de peso, & resiste 30 unidades de peso, # resiste 90 unidades de peso.

Para calcular el peso del resto de la viga se utiliza la siguiente fórmula: cada nueva secuencia de larguero, el primer larguero pesa una unidad de peso, el segundo peso 2 unidades y así hasta el final de la secuencia de largueros continuos. Cada conexión pesa el doble que su anterior secuencia de largueros. Cada nueva secuencia de largueros se calcula de forma similar iniciando en uno.

Hay que determinar si la viga está bien construida y si la base resiste el peso de todos los largueros y conexiones.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese la viga: %
La viga soporta el peso!
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese la viga: %=\*  
La viga soporta el peso!

**Prueba 3:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese la viga: %===\*  
La viga soporta el peso!

**Prueba 4:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese la viga: #====\*=====\*=  
La viga soporta el peso!

**Prueba 5:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese la viga: &====\*=====\*=\*=====\*=\*=====\*=\*=====  
La viga NO soporta el peso!

**Prueba 6:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese la viga: &=====\*\*=====\*\*=====!=  
La viga está mal construida!

## 29 PUENTES DE MADISON

Los ingenieros civiles requieren de un lenguaje que les permitan describir los componentes de los puentes. Se ha propuesto el siguiente lenguaje que permite describir los componentes del puente.

*	Base de los puentes
=	Plataforma del puente
+	Refuerzo del puente

Los puentes deben cumplir las siguientes reglas simples:

- Las bases (\*) solamente pueden estar en los extremos de los puentes.
- Cada dos plataformas se requiere de un refuerzo al lado de ellas.
- Se permiten tres plataformas seguidas únicamente en el centro del puente.
- Los puentes deben ser simétricos (igual la parte izquierda a la parte derecha).
- Si se permiten 2 o más refuerzos seguidos.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese el puente: \*==+==+==\*  
VALIDO

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el puente: *==+==+==*  
VALIDO
```

**Prueba 3:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el puente: *==+====+==*  
INVALIDO
```

**Prueba 4:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el puente: ==+==+==  
INVALIDO
```

**Prueba 5:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el puente: **  
VALIDO
```

## 30 OPERACIONES EN UNA MATRIZ

Construir un programa que cree una matriz cuadrada de orden N. La matriz se debe llenar con la siguiente formula  $Celda[i][j] = (i + 1) - j$ . Luego debe mostrar la matriz como quedo llena y acto seguido, mostrar: la sumatoria de los elementos de la matriz, el mayor elemento de la matriz y el menor elemento de la matriz.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese orden de la matriz: 5  
1    0    -1    -2    -3  
2    1     0    -1    -2  
3    2     1     0    -1  
4    3     2     1     0  
5    4     3     2     1  
La sumatoria es: 25  
El valor máximo es: 5  
El valor mínimo es: -3
```

## 31 TRIANGULAR INFERIOR DE UNA MATRIZ

Construir un programa que cree una matriz cuadrada de orden N. La matriz se debe llenar con la siguiente formula  $Celda[i][j] = i + j$ . Luego debe mostrar la matriz como quedo llena y acto seguido, mostrar la triangular inferior de la matriz.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese orden de la matriz: 10
0   1   2   3   4   5   6   7   8   9
1   2   3   4   5   6   7   8   9  10
2   3   4   5   6   7   8   9  10  11
3   4   5   6   7   8   9  10  11  12
4   5   6   7   8   9  10  11  12  13
5   6   7   8   9  10  11  12  13  14
6   7   8   9  10  11  12  13  14  15
7   8   9  10  11  12  13  14  15  16
8   9  10  11  12  13  14  15  16  17
9  10  11  12  13  14  15  16  17  18
0
1   2
2   3   4
3   4   5   6
4   5   6   7   8
5   6   7   8   9  10
6   7   8   9  10  11  12
7   8   9  10  11  12  13  14
8   9  10  11  12  13  14  15  16
9  10  11  12  13  14  15  16  17  18
```

## 32 ROMBO

Hacer un programa que imprima un rombo con el símbolo # dependiendo del tamaño que el usuario solicite.

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el tamaño del rombo: 7
#
# #
#  #
#   #
#    #
#   #
#  #
# #
#
```

**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el tamaño del rombo: 5
#
# #
#  #
#   #
#  #
#
```



### 33 MULTIPLICACIÓN DE MATRICES

Dos matrices A y B son multiplicables si el número de columnas de A coincide con el número de filas de B.

$$A_{m \times n} \times B_{n \times p} = C_{m \times p}$$

El elemento  $C_{ij}$  de la matriz producto se obtiene multiplicando cada elemento de la fila  $i$  de la matriz A por cada elemento de la columna  $j$  de la matriz B y sumándolos.

$$A \cdot B = \begin{pmatrix} 2 & 0 & 1 \\ 3 & 0 & 0 \\ 5 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix} =$$
$$= \begin{pmatrix} 2 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 & 2 \cdot 0 + 0 \cdot 2 + 1 \cdot 1 & 2 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 \\ 3 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 & 3 \cdot 0 + 0 \cdot 2 + 0 \cdot 1 & 3 \cdot 1 + 0 \cdot 1 + 0 \cdot 0 \\ 5 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 & 5 \cdot 0 + 1 \cdot 2 + 1 \cdot 1 & 5 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 \end{pmatrix} =$$
$$= \begin{pmatrix} 3 & 1 & 2 \\ 3 & 0 & 3 \\ 7 & 3 & 6 \end{pmatrix}$$

Hacer un programa que pida por pantalla las variables m, n y p. Con estos datos crear la matriz A de dimensiones m x n y la matriz B con dimensiones n x p. Los elementos de la matriz A llenarla con la siguiente formula:  $A[i][j] = (i + 1) * j$ , Los elementos de la matriz B llenarla con la siguiente formula:  $B[i][j] = (j + 1) * i$ . Luego mostrar la matriz resultante de la multiplicación de A x B.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```
Ingrese el valor de m: 3
Ingrese el valor de n: 4
Ingrese el valor de p: 2
*** A ***
0 1 2 3
0 2 4 6
0 3 6 9
*** B ***
0 0
1 2
2 4
3 6
*** C ***
14 28
28 56
42 84
```

### 34 RELOJ DE ARENA

**2. Reloj de Arena (Valor 50%):** Haga un programa que llene una matriz cuadrada de orden N (asuma que N siempre es un número impar) con los siguientes números (asumiendo que N vale 11).

	0	1	2	3	4	5	6	7	8	9	10
0	0	2	4	6	8	10	12	14	16	18	20
1	1	3	5	7	9	11	13	15	17	19	21
2	2	4	6	8	10	12	14	16	18	20	22
3	3	5	7	9	11	13	15	17	19	21	23
4	4	6	8	10	12	14	16	18	20	22	24
5	5	7	9	11	13	15	17	19	21	23	25
6	6	8	10	12	14	16	18	20	22	24	26
7	7	9	11	13	15	17	19	21	23	25	27
8	8	10	12	14	16	18	20	22	24	26	28
9	9	11	13	15	17	19	21	23	25	27	29
10	10	12	14	16	18	20	22	24	26	28	30

Y luego muestre en pantalla los números que tienen fondo gris.

**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese orden de la matriz: 11										
MATRIZ COMPLETA										
0	1	2	3	4	5	6	7	8	9	10
2	3	4	5	6	7	8	9	10	11	12
4	5	6	7	8	9	10	11	12	13	14
6	7	8	9	10	11	12	13	14	15	16
8	9	10	11	12	13	14	15	16	17	18
10	11	12	13	14	15	16	17	18	19	20
12	13	14	15	16	17	18	19	20	21	22
14	15	16	17	18	19	20	21	22	23	24
16	17	18	19	20	21	22	23	24	25	26
18	19	20	21	22	23	24	25	26	27	28
20	21	22	23	24	25	26	27	28	29	30
RELOJ DE ARENA										
0	1	2	3	4	5	6	7	8	9	10
	3	4	5	6	7	8	9	10	11	
		6	7	8	9	10	11	12		
			9	10	11	12	13			
				12	13	14				
					15					
					16	17	18			
				17	18	19	20	21		
			18	19	20	21	22	23	24	
		19	20	21	22	23	24	25	26	27
20	21	22	23	24	25	26	27	28	29	30

## 35 DESCOMPOSICION EN FACTORES

Hacer un programa que descomponga un número en sus factores. Es decir:

$$\begin{array}{c|c} 72 & 2 \\ 36 & 2 \\ 18 & 2 \quad 50 & 2 \\ 9 & 3 \quad 25 & 5 \quad 6 & 2 \quad 33 & 3 \\ 3 & 3 \quad 5 & 5 \quad 3 & 3 \quad 11 & 11 \\ 1 & & 1 & & 1 & & 1 \end{array}$$

**Prueba 1:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese el número a descomponer: 72

$72 = 2 \times 2 \times 2 \times 3 \times 3$

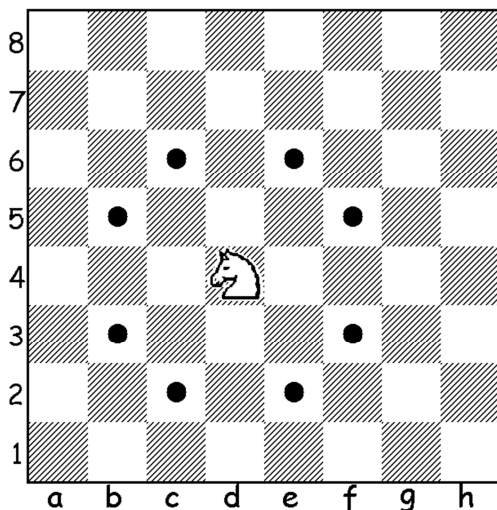
**Prueba 2:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese el número a descomponer: 117  
117 = 3 x 3 x 13

## 36 COSECHANDO A CABALLO

Somos dueños de un campo cuadrado (8x8) en el cual sembramos diferentes frutos algo bien raros: +, -, \*, =. Para hacer aun más rara nuestra cosecha, la recogemos a caballo. Este caballo solamente puede moverse como los caballos del ajedrez, siempre en L.

En el problema se da un campo regado con nuestros frutos y luego la ubicación del caballo dentro del campo donde va iniciar su cosecha. A continuación se da una secuencia de movimientos del caballo. Esta secuencia de movimientos sigue la notación dada en el ajedrez.



El caballo en la posición d4 se puede mover a las posiciones c2, e2, b3, f3, b5, f5, c6 y f7. Solamente a esas posiciones. Los movimientos del caballo los vamos a representar así:

- Arriba izquierda = UL
- Arriba derecha = UR

- Izquierda arriba = LU
- Izquierda abajo = LD
- Derecha arriba = RU
- Derecha abajo = RD
- Abajo izquierda = DL
- Abajo derecha = DR

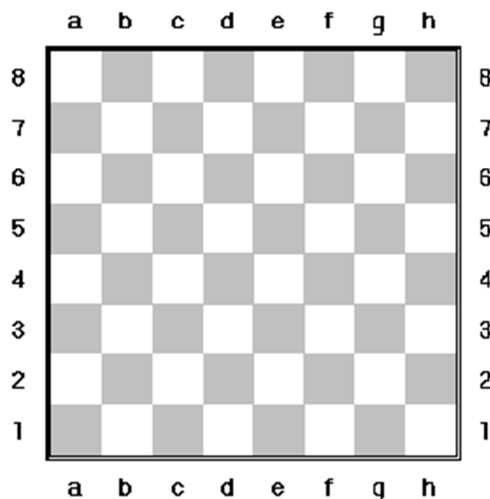
**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

Ingrese ubicación de los frutos: C4+,C7\*,E3-,E1=,H4\*  
 Ingrese posición inicial del caballo: B6  
 Ingrese los movimientos del caballo: DR,RD,RD,LD,RU,UR,UL  
 Los frutos recogidos son: +- = \*

### 37 CABALLOS EN CONFLICTO

En un tablero de ajedrez hay varios caballos, algunos de ellos pueden pastar tranquilamente otros pueden estar en peligro de muerte por la influencia de otros.

Son varias los esquemas para numerar un tablero de ajedrez. Nosotros vamos a utilizar el esquema algebraico:



Se debe realizar un programa que pida como entrada las ubicaciones de todos los caballos por Prueba: B7, C5, E2, H7, G5, F6. Y luego liste caballo por caballo, indicando con cual o cuales caballos tiene conflicto. Por Prueba:

B7 conflicto con: C5

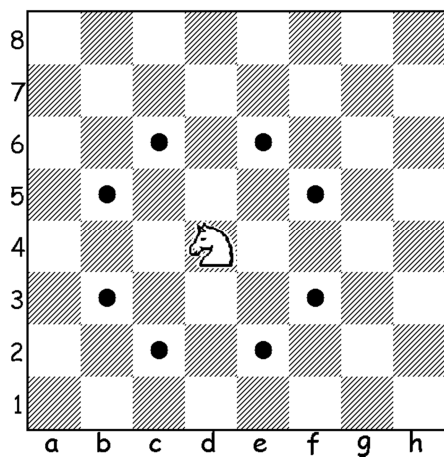
C5 conflicto con: B7

E2 conflicto con: ninguno

H7 conflicto con: G5, F6

G5 conflicto con: H7

F6 conflicto con: H7



**Prueba:** al ejecutar el programa debe mostrar lo siguiente:

```

Ingrese ubicación de los caballos: B7,C5,E2,H7,G5,F6
Analizando Caballo en 7B => Conflicto con 5C
Analizando Caballo en 5C => Conflicto con 7B
Analizando Caballo en 2E =>
Analizando Caballo en 7H => Conflicto con 6F      Conflicto con 5G
Analizando Caballo en 5G => Conflicto con 7H
Analizando Caballo en 6F => Conflicto con 7H

```