

 <b>UCA</b> PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA	<b>Facultad de Ingeniería y Ciencias Agrarias</b>	<b>INFORMÁTICA GENERAL</b>
		<b>Practica N° 3</b>
	<i>Condicionales</i>	

- Ej. 1:** Desarrollar una función que reciba como parámetros dos números y un string con alguno de los cuatro caracteres (+,-,\*,/) y retorne el resultado de la operación. Desde el programa principal el usuario ingresará los datos que serán pasados como parámetros a la función y mostrará el resultado retornado por la misma. Ejemplo:

```
Ingrese el primer número: 8
Ingrese el segundo número: 2
Ingrese la operación (+, -, *, /): +

8 + 2 = 10
```

- Ej. 2:** Desarrollar una función que reciba tres números como parámetros, e imprima en pantalla los números pasados por parámetro de forma ordenada ascendente. La función debe ser invocada desde un programa que solicite el ingreso por teclado de los números. Ejemplo:

```
Ingrese el primer número: 8
Ingrese el segundo número: 2
Ingrese el tercer número: 7

Los números ordenados en forma ascendente son:
2
7
8
```

- Ej. 3:** Desarrollar una función que reciba un número real como parámetro y que retorne un mensaje si el número es “positivo”, “negativo” o “cero”. Adicionalmente deberá desarrollar otra función, que retorne otro mensaje, si el número recibido es “entero natural”, “entero” o “real”. El programa principal deberá efectuar el ingreso de un número real e imprimir por pantalla los mensajes retornados por las funciones. Ejemplos:

Ingrese un número: 13.4 El número es positivo y real.	Ingrese un número: 0 El número es cero y entero.
Ingrese un número: -4 El número es negativo y entero.	Ingrese un número: 7 El número es positivo y entero natural.

 <b>UCA</b> PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA	<b>Facultad de Ingeniería y Ciencias Agrarias</b>	<b>INFORMÁTICA GENERAL</b>
		<b>Practica N° 3</b>
	<i>Condicionales</i>	

**Ej. 4:** Desarrollar una función booleana que reciba como parámetro dos números enteros (*que no están en orden*) y valide si la resta entre el número mayor y el número menor es un valor que se encuentra entre ambos números (*es decir, la diferencia es mayor e igual que el más chico y menor e igual que el más grande de los valores recibidos*). Escribir un programa que ingrese por teclado los dos valores, invoque a la función y muestre por pantalla si cumplen o no con la condición. Ejemplos:

Ingrese un número A: 13 Ingrese un número B: 14  NO cumple condicion.	Ingrese un número A: -4 Ingrese un número B: 10  NO cumple condicion.
Ingrese un número A: 16 Ingrese un número B: 7  SI cumple condicion.	Ingrese un número A: 0 Ingrese un número B: 0  SI cumple condicion.

**Ej. 5:** Desarrollar una función booleana que reciba como parámetros tres números enteros positivos que representan el día, el mes y el año de una fecha. La función deberá retornar verdadero (True) si la fecha es válida caso contrario deberá retornar falso (False).

Desde el programa principal ingresar por teclado el día, mes y año por separado, invocar a la función y mostrar por pantalla un mensaje indicando si la fecha es correcta o no.

**Ayuda:** Un año es bisiesto si es múltiplo de 4 y no de 100, o es múltiplo de 400. Por ejemplo el año 2000 es bisiesto pero el 1800 no lo es.

Ejemplo:

Ingrese el día: 29 Ingrese el mes: 2 Ingrese el año: 2000  La fecha es correcta.
--

**Casos de prueba:**

- |                           |                           |
|---------------------------|---------------------------|
| - 29/2/2000 es correcta   | - 8/13/2007 es incorrecta |
| - 28/2/2000 es correcta   | - 29/2/1900 es incorrecta |
| - 29/2/1996 es correcta   | - 30/2/2000 es incorrecta |
| - 29/2/2012 es correcta   | - 0/10/2009 es incorrecta |
| - 5/12/1903 es correcta   | - 0/2/2000 es incorrecta  |
| - 31/4/2008 es incorrecta | - 30/0/2004 es incorrecta |
| - 32/7/2005 es incorrecta | - 30/2/2004 es incorrecta |

- Ej. 6:** Desarrollar una función booleana que reciba como parámetro un número entero positivo. Si el número recibido es par solicitará que se ingrese un número menor, y si es impar se solicitará que se ingrese mayor. La función deberá verificar si el ingreso es correcto retornando verdadero (True) o incorrecto retornando falso (False). Desde el programa principal ingresar por teclado el número entero positivo, invocar a la función e imprimir un mensaje indicando si es "Correcto!" o "Incorrecto!" según el resultado que retorna por la función. Ejemplos:

Ingrese un número entero positivo: 6 Ingrese un número menor que 6: 2  Correcto!	Ingrese un número entero positivo: 9 Ingrese un número mayor que 9: 1  Incorrecto!
---	---

- Ej. 7:** Desarrollar una función booleana que reciba como parámetros tres números enteros sin un orden en particular. Luego debe verificar si el promedio entre el mayor y el menor es exactamente igual al número restante, (*es decir: los extremos están igualmente distanciados del intermedio*), retornando verdadero (True) si cumple la condición o falso (False) en caso contrario. Desde el programa principal ingresar por teclado tres números enteros, invocar a la función e imprimir un mensaje indicando si los tres valores "Están igualmente distanciados!" o "NO Están igualmente distanciados!" según el resultado retornado por la función. Ejemplo:

Ingrese el primer número: 5 Ingrese el segundo número: 2 Ingrese el tercer número: 8  Están igualmente distanciados!	Ingrese el primer número: 5 Ingrese el segundo número: 5 Ingrese el tercer número: 3  NO Están igualmente distanciados!
--	---

**Casos de prueba:**

Primero	Segundo	Tercero	Mensaje
2	5	8	Están igualmente distanciados!
5	5	3	NO Están igualmente distanciados!
8	2	5	Están igualmente distanciados!
5	8	6	NO Están igualmente distanciados!
5	7	6	Están igualmente distanciados!

**Ej. 8:** Realizar un programa que solicite 3 notas de parciales obtenidas por un alumno. A continuación se mostrará por pantalla un mensaje que indique la situación del alumno:

- Si el alumno aprobó los 3 parciales (nota 4 o más) y su promedio es mayor a 7, promociona la materia con la nota promedio.
- Si el alumno aprobó los 3 parciales pero su promedio no supera los 7 puntos, debe rendir examen final.
- Si el alumno no aprobó uno o más parciales, se solicitará que se ingrese la nota de un recuperatorio. Si éste hubiera sido aprobado, se informará el promedio general (3 parciales + el recuperatorio) y su condición de aprobado (*aún cuando el promedio no supere los 4 puntos*). Si no hubiera aprobado el recuperatorio se informará que el alumno fue aplazado.

Desarrollar dos funciones: una que reciba como parámetro las tres notas de los parciales, y calcule y retorne el valor del promedio. Y otra que reciba las tres notas de parciales y la nota del recuperatorio, y retorne el promedio general (de las cuatro notas).  
Ejemplo:

```
Ingrese la nota del primer parcial: 2
Ingrese la nota del segundo parcial: 6
Ingrese la nota del tercer parcial: 2
Ingrese la nota del recuperatorio: 4
```

```
Promedio general = 3.5
El alumno deberá rendir final.
```

	<b>Facultad de Ingeniería y Ciencias Agrarias</b>	<b>INFORMÁTICA GENERAL</b>
		<b>Practica N° 3</b>
	<i>Condicionales</i>	

**Ej. 9:** Una empresa necesita calcular el sueldo total que dará a sus empleados a fin de este año. Para ello se sigue el siguiente criterio:

- Si el sueldo base supera los \$2000, el bono será del 15%. De lo contrario, el **bono** será del 20%.
- Además, recibirá un **plus**, de acuerdo al siguiente criterio:
  - Si el empleado tiene hijos se suma un plus del 5% del sueldo.
  - Si el empleado pertenece a la categoría 1, 2 ó 3, recibe un 10% del sueldo. Si pertenece a la categoría 4, 5 ó 6, recibe un 12% del sueldo. Si es de la categoría 7, 8 ó 9, recibe un 20% del sueldo pero no cobra el plus por tener hijos.

Realizar un programa que solicite al usuario la información necesaria (*sueldo base*, *hijos(s/n)* y *categoría*) para luego calcular el sueldo total. Para el cálculo del sueldo total se deberá resolver invocando a las siguientes funciones:

- La primer función: **bono** recibe por parámetro el sueldo base, luego calcula y retorna el valor del bono.
- La segunda función: **plus** recibe por parámetro el sueldo base y si tiene hijos o no (booleana) y la categoría. Esta función debe retornar la suma del plus por hijo más el plus por categoría. Para resolver al plus por hijo, deberá invocar a **plusH** y para resolver el plus por categoría deberá invocar a **plusC**.
- La tercera función: **plusH** recibe dos parámetros; el sueldo base y otra variable (booleana) que indica si tiene hijos(True) o no tiene hijos (False). Luego calcula y retorna el plus por hijos
- La cuarta función: **plusC** recibe por parámetro el sueldo base y el número de categoría, luego calcula y retorna el plus por categoría.

Desde el programa principal solicitar al usuario el ingreso de los datos, luego invocar a las funciones que correspondan, y por último mostrar por pantalla el total a pagar al empleado.

**Ejemplo:**

```
Ingrese el sueldo Base: 1000
Tiene hijos (s/n)? : s
Ingrese categoría (1 - 9): 2

El sueldo total será de $1350.00
```

 <b>UCA</b> <small>PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA</small>	<b>Facultad de Ingeniería y Ciencias Agrarias</b>	<b>INFORMÁTICA GENERAL</b>
		<b>Practica N° 3</b>
	<b>Condicionales</b>	

**Ej. 10:** Según las normas de tránsito, un vehículo no puede superar la velocidad máxima ni circular a menos que la velocidad mínima (*que es la mitad de la velocidad máxima*). Por una cuestión de errores de medición (*tanto en el automóvil como en los dispositivos de control*) hay un 15% de tolerancia. Sin embargo, se permite que los vehículos en emergencia (*ambulancias, patrulleros, etc.*) superen la velocidad máxima (*no así, circular a menos que la velocidad mínima*).

Hacer un programa que solicite al usuario el ingreso de la velocidad a la que circula un vehículo, la velocidad máxima permitida, y si se trata o no de un vehículo de emergencia (*contestando con "S" o "N" en mayúscula o minúscula*). Luego de invocar a la función **multa**, imprimir en pantalla el mensaje que la retorna función el cual indicará si recibe o no multa.

**Función multa:** Recibe tres parámetro (velocidad del vehículo, velocidad máxima permitida, y un booleano que indica si el vehículo es o no de emergencia). Luego la función retornará un mensaje de acuerdo al siguiente criterio:

- Si la velocidad del vehículo está entre la máxima y la mínima (ambas inclusive), el mensaje deberá ser: "No recibe multa".
- Si la velocidad está dentro de la tolerancia máxima (superando la máxima hasta el 15%, (*por ejemplo, para máxima de 60 km/h la tolerancia llega hasta 69 km/h*); el mensaje deberá ser: "Advertencia".  
*Aclaración:* No se aplica si es un vehículo en emergencia que, en cuyo caso el mensaje deberá ser: "No recibe multa"..
- Si la velocidad del vehículo está dentro de la tolerancia mínima (*hasta 15% por debajo de la velocidad mínima*), el mensaje deberá ser: "Advertencia".  
*Aclaración:* Este caso es independiente del tipo de vehículo.
- Si la velocidad del vehículo supera la velocidad máxima y el límite de tolerancia, el mensaje deberá ser: "Recibe multa por exceso de velocidad".  
*Aclaración:* No se aplica si es un vehículo en emergencia qué, en cuyo caso el mensaje deberá ser: "No recibe multa".
- Si la velocidad del vehículo no exceda la tolerancia de velocidad mínima, el mensaje deberá ser: "Recibe multa por entorpecer el tránsito".  
*Aclaración:* Este caso es independiente del tipo de vehículo.

Ejemplos:

Velocidad del vehículo: 60 velocidad máxima: 80 Emergencia (s/n): n  No recibe multa	Velocidad del vehículo: 100 velocidad máxima: 80 Emergencia (s/n): s  No recibe multa
--	---

	<b>Facultad de Ingeniería y Ciencias Agrarias</b>	<b>INFORMÁTICA GENERAL</b>
		<b>Practica N° 3</b>
	<i>Condicionales</i>	

**Ej. 11:** Una empresa se dedica a vender cañerías de gas y dispone únicamente de dos longitudes de caños que pueden empalmarse en línea recta pero no pueden cortarse. Las longitudes disponibles son 1 metro y 5 metros.

Desarrollar un programa que solicite el ingreso por teclado de la cantidad disponible de caños de 1 metro, la cantidad disponible de caños de 5 metros y la longitud total de tendido (*entero positivo expresado en metros*). El programa debe informar por pantalla si es posible cubrir exactamente la necesidad (*sin pasarse*) y sugerir una posible combinación de caños para alcanzar el total. Para determinar el mensaje, desde el programa principal se deberá invocar a la función mensaje.

Función mensaje: Recibe como parámetros los datos ingresados en el programa principal, y verifica dicho parámetros. Luego retorna un mensaje donde indicará si es posible o no cubrir el tendido. Y para el caso de ser posible agrega al mensaje sugerencias con la combinación posible de caños.

Ejemplos:

```
Cantidad de caños de 1 metro: 12
Cantidad de caños de 5 metros: 3
Metros totales a cubrir: 21
```

```
Es posible cubrir el tendido.
Sugerencia:
3 unidades de caño de 5 metros
6 unidades de caño de 1 metro
```

```
Cantidad de caños de 1 metro: 2
Cantidad de caños de 5 metros: 10
Metros totales a cubrir: 29
```

```
No es posible cubrir el tendido.
```